Representative Approach

for Big Data Dimension Reduction with Binary Responses

by

Xuelong Wang
B.S., Xi'an University of Technology, China, 2011
M.S., West Virginia University, West Virginia, 2013

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Mathematics
in the Graduate College of the
University of Illinois at Chicago, 2020

Chicago, Illinois

Defense Committee:
Dr. Jie Yang, Chair and Advisor
Dr. Yichao Wu
Dr. Jing Wang
Dr. Cheng Ouyang
Dr. Hua Yun Chen, Division of Epidemiology and Biostatistics

ACKNOWLEDGMENT

I want to take this opportunity to thank all the people who have been helping and supporting me through this most important journey in my life.

Firstly, I would like to thank my thesis advisor, Dr. Jie Yang, for the continuous encouragement and help of my Ph.D. study and research. His immense knowledge and insightful advice have helped me tremendously in all the time of research and writing of this thesis. I am especially grateful for his generous support, which allows me to overcome many difficulties. I could not have thought of a better choice of the mentor for my Ph.D.

I wish to express my sincere gratitude to my dissertation committee for their kindness and flexibility. Their critical comments and valuable suggestions have helped me widen my thesis from various perspectives. Particularly, I would like to thank Dr. Yichao Wu for his insightful feedback. His constructive comments and suggestions not only lead to a significant improvement to my thesis but also inspire me in the direction for the future work of this topic. I deeply appreciate Dr. Wu's tremendous help during this process. I also want to thank Dr. Jing Wang for sharing her profound insight of statistics with me during our independent study. I want to thank Dr. Cheng Ouyang for generously answering all my questions and patiently explaining abstract probability concepts to me. I also want to thank Dr. Hua Yun Chen for demonstrating a rigorous research attitude during our research project.

ACKNOWLEDGMENT (Continued)

Last but not least, I am very grateful to my wife and my parents for their unconditional love and patience. I would also like to thank my friend, Niuniu, who has provided me with moral and emotional support through this process.

I would also like to thank Dr. Seung Jun Shin for sharing their computer program. This work is supported in part by LAS Award for Faculty of Science at UIC and NSF grant DMS-1924859.

XW

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

CHAPTER **PAGE**

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Sufficient dimension reduction (SDR) reduces the data dimensionality without specifying a regression model. Since it was first introduced by Li, 1991, SDR has been popular and many SDR methods have been proposed and studied (Cook and Weisberg, 1991; Xia et al., 2009; Li and Wang, 2007; Lee et al., 2013). Among those methods, we focus on Sliced Inverse Regression (SIR) and Sliced Average Variance Estimation (SAVE), which are inverse-moment based methods (details in Section 1.1). Those methods work well with continuous responses, but not with binary cases due to the limited number of levels of the response, which is reviewed and studied in Sections 1.2 and 3.1. In order to solve the issue, Shin et al., 2014 have proposed a solution for SDR methods on binary data called Probability Enhanced SDR (PRE-SDR). The PRE-SDR works well under a binary dataset. But it becomes time-consuming when a dataset is large, e.g., $N > 10^4$, because of its computational intensity (details in Sections 1.3 and 5.3).

In this thesis, motivated by the existing solution and its limitation on large data, we investigate and improve the SIR and SAVE from different perspectives. Firstly, we incorporate an online algorithm, which helps to reduce the usage of computer memory when a dataset is large. The general idea of this method is to scan the data chunk by chunk, calculate intermediate statistics, and combine intermediate results to get the final result. We develop online algorithms for SIR and SAVE and show that the online method's result is the same as it calculated from using the full data at once. Besides, we enhance those algorithms with a parallel computation framework so that it could process multiple chunks at the same time. Simulation

results suggest that the online algorithm reduces the computational time at least by 3-5 times compared with the original methods.

Secondly, we propose a novel SDR approach, named as Mean Representative approach (MRDR), for binary responses. The main idea is to partition the data into blocks, calculate representatives for each block, and use the representatives as our new dataset for the following SDR analysis. By converting a block of data points into a representative data point, the corresponding binary responses become continuous, and the size of the data is reduced significantly because the number of the block is much smaller than the original observations. Therefore, the proposed representative approach provides an ideal solution for large data dimension reduction and can be incorporated with the classical SDR approaches naturally. The details of MRDR are introduced and discussed in Chapters 1 and 3. We study the asymptotic properties of MRDR in Chapter 4 and show that the proposed approach can recover the central subspace better than SIR and SAVE. Besides, we also discuss the optimal choice of the number of blocks in Section 4.3. The simulation studies in Chapter 5 verify the advantage of the proposed method over the original SIR and SAVE in estimating the central subspace and demonstrates the time efficiency compared to PRE-SIR. In the end, we apply the proposed method on the Electrical Grid Stability (EGS) data and simulated data based on the EGS data. The result shows the advantage of the proposed method over the several existing methods on sufficient dimension reduction with large data.

CHAPTER 1

INTRODUCTION

In this chapter, we introduce the background of sufficient dimension reduction (SDR) and two commonly used SDR methods, which are SIR and SAVE. In order to have a better understanding of how SDR methods work under the binary response, we also review two models for binary response data, which is latent variable model and link function model. Then we briefly discuss the issue of SDR methods under the binary response. Next, we review the existing solution for the issue and its limitations. In the end, we introduce our proposed solutions for big data.

1.1 Sufficient dimension reduction

Sufficient dimension reduction is based a model-free assumption (see details in Cook and others, 2007), which is

$$Y|\mathbf{X} \stackrel{d}{=} Y|\boldsymbol{\eta}^{\mathbf{T}}\mathbf{X}, \tag{1.1}$$

where $\mathbf{X} \in \mathbb{R}^p$, and $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_d) \in R^{p\times d}$. Under the Equation (1.1), the conditional distribution of Y given X is identical with the conditional distribution of Y given $\boldsymbol{\eta}^{\mathbf{T}}\mathbf{X}$. Moreover, if the joint distribution of $\mathbf{X}$ and Y exists, then condition Equation (1.1) is equivalent to

$$Y \perp\!\!\!\perp \mathbf{X}|\boldsymbol{\eta}^{\top}\mathbf{X}, \tag{1.2}$$

which means $\mathbf{X}$ and Y are conditional independent, after we fix d linear combinations of $(\eta_1^\top \mathbf{X}, \ldots, \eta_d^\top \mathbf{X})$. All of those conditions conveys the idea that $\boldsymbol{\eta}^\top \mathbf{X}$ carries all the information of Y. So if we want to fit a model between $\mathbf{X}$ and Y, we can focus on $\boldsymbol{\eta}^\top \mathbf{X}$ instead of the original $\mathbf{X}$, and by doing that we achieve the dimension reduction. Therefore, the goal of SDR method is to estimate the d linear combinations, $\boldsymbol{\eta}$. However, $\boldsymbol{\eta}$ itself is not identifiable, because any column transformation of $\boldsymbol{\eta}$ will still satisfy Equation (1.2). That is

$$Y|\mathbf{X} \overset{d}{=} Y|\boldsymbol{\eta}^{\mathbf{T}}\mathbf{X} \Rightarrow Y|\mathbf{X} \overset{d}{=} Y|(\boldsymbol{\eta}\mathbf{A})^\top \mathbf{X},$$

where $\mathbf{A}$ is a $d \times d$ non-singular matrix. Fortunately, the linear space spanned by $\boldsymbol{\eta}$: $\mathtt{Span}(\boldsymbol{\eta})$ is identifiable because it is invarinate of column transformation. Cook and others, 2007 further defined

$$Y \perp\!\!\!\perp \mathbf{X}|\mathbf{P}_{\mathcal{S}(\boldsymbol{\eta})}\mathbf{X}, \quad \mathcal{S}(\boldsymbol{\eta}) = \boldsymbol{\eta}(\boldsymbol{\eta}^\top \boldsymbol{\eta})^{-1}\boldsymbol{\eta}^\top, \tag{1.3}$$

where $\mathcal{S}$ is the column space of the matrix $\boldsymbol{\eta}$ and is named dimension-reduction space (drs). Now the goal of SDR methods becomes to estimate $\mathcal{S}$. However, $\mathcal{S}$ may also not be unique. Actually if $\mathcal{S} \subset \mathcal{S}_1$, then $\mathcal{S}_1$ is also a dimension-reduction space. In order to get the well defined target, Cook, 2009 has defined the smallest drs, which is

$$\mathcal{S}_{Y|\mathbf{X}} = \cap \mathcal{S}_{\mathtt{drs}},$$

where $\mathcal{S}_{Y|\mathbf{X}}$ is named Central Subspace. Under mild conditions (see details in Cook, 1994), $\mathcal{S}_{Y|\mathbf{X}}$ is unique and itself is a drs. Therefore, the final target of SDR methods is to recover partially or even fully the $\mathcal{S}_{Y|\mathbf{X}}$.

Linearity and constant variance conditions

To estimate $\mathcal{S}_{Y|\mathbf{X}}$, most SDR methods rely on two conditions of the distribution $\mathbf{X}$. One is the linearity condition, which assumes that

$$\mathsf{E}(\mathbf{X}|\boldsymbol{\eta}^{\mathsf{T}}\mathbf{X}) \text{ is a linear function of } \boldsymbol{\eta}^{\mathsf{T}}\mathbf{X}. \tag{1.4}$$

A sufficient and necessary condition for the condition Equation (1.4) is

$$\mathsf{E}\left(\mathbf{X}|\boldsymbol{\eta}^{\mathsf{T}}\mathbf{X}\right) - \mathsf{E}(\mathbf{X}) = \mathbf{P}^{\mathsf{T}}_{\mathcal{S}_{Y|\mathbf{X}}(\boldsymbol{\Sigma}_{\mathbf{X}})}(\mathbf{X} - \mathsf{E}(\mathbf{X})), \tag{1.5}$$

where $\mathsf{Var}(\mathbf{X}) = \Sigma_{\mathsf{X}}$ and $\mathcal{P}_{\mathcal{S}_{Y|\mathbf{X}}} = \boldsymbol{\eta}\left(\boldsymbol{\eta}^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathbf{X}}\boldsymbol{\eta}\right)^{-1}\boldsymbol{\eta}^{\mathsf{T}}\boldsymbol{\Sigma}_{\mathsf{X}}$ which is the projection matrix of $\mathcal{S}_{Y|\mathbf{X}}$ under the Σ_{X} inner product, or $\Sigma_{\mathsf{X}}\mathcal{S}_{Y|\mathbf{X}}$. Based on this condition, we may interpret the linearity condition as that the conditional expectation is actually a linear operation. For constant variance condition, it assumes

$$\mathsf{Var}(\mathbf{X}|\boldsymbol{\eta}^{\mathsf{T}}\mathbf{X}) = \mathbf{C}, \tag{1.6}$$

where $\mathbf{C}$ is a constant matrix. Note that a sufficient condition of linearity condition is that $\mathbf{X}$ has an elliptically contoured distribution and a sufficient condition for both linearity and constant variance condition is the multivariate normal distribution (see details in Li, 1991).

1.1.1 Methods for estimating the central subspace

Sliced inverse regression(SIR)

In (Li, 1991), the authors have found the connection between the inverse moment and the $\mathcal{S}_{Y|\mathbf{X}}$. Inverse moments refers to the moments of the conditional distribution of $\mathbf{X}$ given Y. The inverse version of Equation (1.1) is

$$\mathbf{X}|(Y, \boldsymbol{\eta}^{\mathsf{T}}\mathbf{X}) \stackrel{\mathrm{d}}{=} \mathbf{X}|\boldsymbol{\eta}^{\mathsf{T}}\mathbf{X}. \tag{1.7}$$

Under the conditional independence Equation (1.7) and linearity Equation (1.4), we have

$$\begin{aligned}
\mathsf{E}(\mathbf{X}|Y) - \mathsf{E}(\mathbf{X}) &= \mathsf{E}\left[\mathsf{E}\left(\mathbf{X}|Y, \mathbf{P}_{\mathcal{S}_{Y|\mathbf{X}}}\mathbf{X}\right)|Y\right] - \mathsf{E}(\mathbf{X}) \text{ b/c Equation (1.7)} \\
&= \mathsf{E}\left\{\mathbf{P}^{\mathsf{T}}_{\mathcal{S}_{Y|\mathbf{X}}(\boldsymbol{\Sigma}_{\mathbf{X}})}[\mathbf{X} - \mathsf{E}(\mathbf{X})]|Y\right\} \text{ b/c Equation (1.5)} \\
&= \mathbf{P}^{\mathsf{T}}_{\mathcal{S}_{Y|\mathbf{X}}(\boldsymbol{\Sigma}_{\mathbf{X}})}[\mathsf{E}(\mathbf{X}|Y) - \mathsf{E}(\mathbf{X})]
\end{aligned} \tag{1.8}$$

This fact shows that the centered conditional expectation of $\mathbf{X}$ is exact same after projecting it into the a subspace $\boldsymbol{\Sigma}_X \mathcal{S}_{Y|\mathbf{X}}$. That means $\mathsf{E}(\mathbf{X}|Y) - \mathsf{E}(\mathbf{X}) \in \boldsymbol{\Sigma}_{\mathbf{X}}\mathcal{S}_{Y|\mathbf{X}} \subseteq \mathbb{R}^p$. Let's define the candidate matrix of SIR, $\mathbf{M}_{\mathrm{SIR}}$, as conditional variance,

$$\mathbf{M}_{\mathrm{SIR}} = \mathrm{var}[\mathsf{E}(\mathbf{X}|Y)] = \mathsf{E}\left[[\mathsf{E}(\mathbf{X}|Y) - \mathsf{E}(\mathbf{X})][\mathsf{E}(\mathbf{X}|Y) - \mathsf{E}(\mathbf{X})]^{\mathsf{T}}\right].$$

Based on the Equation (1.8), Li has shown that $Span(\mathbf{M}_{SIR}) \subseteq \mathbf{\Sigma}_X S_{Y|\mathbf{X}} \Rightarrow \mathbf{\Sigma}_X^{-1} Span(\mathbf{M}_{SIR}) \subseteq S_{Y|\mathbf{X}}$. Therefore, we define

$$S_{SIR} := \mathbf{\Sigma}_X^{-1} Span(\mathbf{M}_{SIR}). \tag{1.9}$$

We can use eigenvalue decomposition to $\mathbf{\Sigma}_X^{-1} Span(\mathbf{M}_{SIR})$ to find a basis of the central subspace:

$$\mathbf{M}_{SIR}\eta_i = \lambda_i \mathbf{\Sigma}_{\mathbf{X}}\eta_i, \ i = 1, \ldots, d, \tag{1.10}$$

where λ_i is the ith largest eigenvalue of $\mathbf{M}_{SIR}$ and η_i is the corresponding eigenvector. Not that $d = dim(\mathcal{S}_{Y|\mathbf{X}})$ and also is the non-zero eigenvalues of $\mathbf{M}_{SIR}$.

$E(X|Y)$ estimation by slicing

The next questions is how to estimate the $\mathbf{M}_{SIR}$ and $\mathbf{\Sigma}_X$. Since we assume $n >> p$, $\mathbf{\Sigma}_X$ can be estimated by the sample covariance $\hat{\mathbf{\Sigma}}_X = \sum_i^N \frac{1}{N-1}(\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^T$, where $\bar{\mathbf{X}} = \frac{1}{N}\sum_i^N(\mathbf{X}_i)$, is the sample mean of $\mathbf{X}$. For estimating $\mathbf{M}_{SIR}$, we need to estimate the $E(\mathbf{X}|Y)$ first. Li proposed a slicing method to calculate the conditional expectations. The steps are following:

1. Split the data in to H slices based on their responses, $Y \in \mathbb{R}$. The H slices are non-overlap intervals, $I_h, h = 1, \ldots, H$, so that $I_h \cap I_{h'} = \emptyset, \ \forall j \neq h'$ and $\cup_h I_h = \mathbb{R}$. Let $\tilde{Y}_i = h$ if Y_i is in the hth slices, $h = 1, \ldots, H$.
2. Average all the $\mathbf{X}_i$'s within each slice, $\bar{\mathbf{X}}_h = \frac{\sum_i^{N_h} X_i}{N_h}$, where N_h is the total observations in the slice h.

3. Let $f_i = \frac{N_h}{N}$,

$$\widehat{\mathbf{M}}_{SIR} = \sum_{h=1}^{H} f_h \left(\overline{\mathbf{X}}_h - \overline{\mathbf{X}}\right)\left(\overline{\mathbf{X}}_h - \overline{\mathbf{X}}\right)^T$$

Remark 1.1.1. *Choice of* H

Let $\mathbb{1}_{I_h}(Y_i)$ *be the indicator function of* I_h. *Then we have* $E(\bar{\mathbf{X}}_h) = E(\mathbf{X}|\tilde{Y} = h) = E(\mathbf{X}|\mathbb{1}_{I_h}(Y_i) = 1)$. *Since* $\sigma(\mathbb{1}_{I_h}(Y)) \subseteq \sigma(Y),\ h = 1, \ldots, H$, $\sigma(E(\mathbf{X}|\tilde{Y})) \subseteq \sigma(E(\mathbf{X}|Y))$??. *Therefore, we have* $\mathcal{S}_{\tilde{Y}|\mathbf{X}} \subseteq \mathcal{S}_{Y|\mathbf{X}}$. *When h is large enough, we will have* $\mathcal{S}_{\tilde{Y}|\mathbf{X}} = \mathcal{S}_{Y|\mathbf{X}}$. *In (Li, 1991), the authors have shown that* $H = 10 - 20$ *should be large enough.*

After we get $\widehat{\mathbf{M}}_{SIR}$ based on the slicing procedure, we can estimate a basis of $\mathcal{S}_{Y|\mathbf{X}}$ via Equation (1.10), and define the sample version of Equation (1.9) as

$$\mathcal{S}_{SIR} = Span(\hat{\eta}_1, \ldots, \hat{\eta}_d) = Span(\hat{\boldsymbol{\eta}}_{SIR}), \tag{1.11}$$

where $\hat{\boldsymbol{\eta}}_{SIR} = \hat{\eta}_1, \ldots, \hat{\eta}_d$. Since SIR only uses the first inverse moment, it runs fast and efficient. But SIR has an issue when estimating a direction which is symmetric with origin, for example, $Y = X_1^2 + \epsilon$.

Sliced Average Variance Estimation (SAVE)

Motivated by SIR and its limitation, Cook and Weisberg, 1991 developed SAVE which uses the second inverse moment to estimate the central subspace. Similar with SIR, SAVE is based on the sliced response $\tilde{Y}$. It not only uses the first inverse moment but also the second inverse moment, which is $Var(\mathbf{X}|\tilde{Y})$. The candidate matrix of SAVE is $\mathbf{M}_{SAVE} = (\Sigma_X - Var(\mathbf{X}|\tilde{Y}))^2$. Based on the linearity and constant variance conditions, we have $Span(\mathbf{M}_{SAVE}) \subseteq \Sigma_X \mathcal{S}_{Y|\mathbf{X}}$,

which means that the conditional covariance is also related to the central subspace. Then the population version of central subspace estimated by SAVE is

$$S_{SAVE} := \mathbf{\Sigma}_X^{-1} Span(\mathbf{M}_{SAVE}). \tag{1.12}$$

Similar with SIR, we need to take the eigenvalue decomposition to find the directions,

$$\mathbf{M}_{SAVE}\eta_i = \lambda_i \mathbf{\Sigma}_{\mathbf{X}}\eta_i,\ i = 1, \ldots, d, \tag{1.13}$$

To estimate the candidate matrix $\mathbf{M}_{SAVE}$, we also need to split the data into slices on Y and let

$$\widehat{\mathbf{M}}_{SAVE} = \sum_{h=1}^{H} f_h \left(\widehat{\mathbf{\Sigma}}_{\mathbf{X}} - \widehat{\mathbf{\Sigma}}_{\mathbf{X}|h}\right)^2,$$

where $\widehat{\mathbf{\Sigma}}_{\mathbf{X}|h} = \frac{1}{N_h - 1}\sum_i^{N_h}(X_i - \bar{X}_h)(X_i - \bar{X}_h)^T$, for all the observations in the slice h. The sample version of Equation (1.12) is

$$\mathcal{S}_{SAVE} = Span(\hat{\eta}_1, \ldots, \hat{\eta}_d) = Span(\hat{\boldsymbol{\eta}}_{SAVE}), \tag{1.14}$$

where $\hat{\boldsymbol{\eta}}_{SAVE} = (\hat{\eta}_1, \ldots, \hat{\eta}_d)$.

Structural dimension

Another component of SDR methods is to estimate the structural dimension of $\mathcal{S}_{Y|\mathbf{X}}$, d. In general, there are four different ways to decide the d, which are sequential tests, bootstrap, BIC-type test, and sparse eigenvalue decomposition test. A review of those methods could be

found in (Ma and Zhu, 2013). Here, we only briefly introduce the sequential test, which is widely used for SDR methods based on inverse moments.

Sequential test of eigenvalues

For the SDR methods based on inverse regression, to decide the number of d is equivalent to detect the non-zero eigenvalues. Therefore, a sequential test is composed of a sequential hypothesis tests about eigenvalues of the candidate matrix. The null hypothesis is $H_0 : \lambda^{(i)} = 0$, where $\lambda^{(i)}$ is the ith largest eigenvalue of the candidate matrix. The alternative hypothesis of ith test is $H_1 : \lambda^{(i)} > 0$. Then, we estimate d as $\hat{d} = i$, when we reject ith test but fail to reject $(i+1)th$ test. Different SDR methods have different sequential tests. For instance, Li, 1991 have proposed a chi-square test for SIR and Cook and Ni, 2005 have proposed a similar weighted chi-square test. A comprehensive review about the sequential test can be found in (Bura and Yang, 2011).

1.2 Binary response

1.2.1 Models for binary response

There are two commonly used models for binary response. Both of them involve the conditional probability $G(\mathbf{X}) = \mathcal{P}(Y = 1|\mathbf{X})$. One is the link function model, which connects the $G(\mathbf{X})$ and $\mathbf{X}$ via a link function. The other one is the latent model which uses a latent continuous variable (see details in Gelman et al., 2013 P410). The reason we review those two models is that each model has its own advantage, so we use different models for different sections. For latent model, it is straightforward for interpretation and data simulation because of its continuous latent variable. For the link function model, since it models the $G(\mathbf{X})$ directly,

it is more convenient for asymptotic study. Moreover, those two models are equivalent to each other in certain conditions. Therefore, we use the latent variable model for simulation studies and the link function model for theory discussion. Note that most of SDR methods only have few assumptions about the model, so they are model-free methods. In binary context, the model-free property is equivalent to assume an arbitrary structure of the $G(\mathbf{x})$. In this thesis, although we need to add certain regular conditions for the function $G(\mathbf{x})$, they are reasonably mild conditions, so that we still keep the same model-free philosophy as the SDR does.

Latent variable model

In the latent model, we assume that there exists a latent variable Y^*, which is related to covariates and an error term by a function f. In order to have a binary response, we classify the latent variable into two groups by a cut off value. Let Y^* as the latent response and Y as the observed binary response,

$$Y = \begin{cases} 0 & Y^* - \theta \leq 0 \\ 1 & Y^* - \theta > 0 \end{cases} \quad \text{or } Y = sign(Y^* - \theta),$$

Where θ is the cutoff value and $sign(x) = \begin{cases} 0 & x \leq 0 \\ 1 & x > 0 \end{cases}$ is the indicator function for positive values.

$$Y^* = f(\mathbf{X}, \epsilon),$$

Where ϵ is a random variable. Thus, we have

$$G(\mathbf{X}) = \mathcal{P}(Y = 1|\mathbf{X}) = \mathcal{P}(Y^* > \theta|\mathbf{X}).$$

If we restrict the additive error structure of f, then we have

$$Y^* = H(\mathbf{X}) + \epsilon \Rightarrow Y = sign(H(\mathbf{X}) + \epsilon - \theta),$$

where $H : \mathbb{R}^p \to \mathbb{R}^1$. Let F_ϵ be the distribution function of ϵ, then we have,

$$\begin{aligned} G(\mathbf{X}) &= \mathcal{P}(Y^* > \theta|\mathbf{X}) \\ &= \mathcal{P}(H(\mathbf{X}) + \epsilon > \theta|\mathbf{X}) \\ &= \mathcal{P}(\epsilon > -H(\mathbf{X}) - \theta) \\ &= 1 - F_\epsilon(-H(\mathbf{X}) - \theta). \end{aligned}$$

Link function model

The link function model assumes that the conditional probability is related to the covariates by a link function. Similar with the generalized (linear) model, we have

$$g(E(Y|\mathbf{X})) = H(\mathbf{X}) \Rightarrow E(Y|\mathbf{X}) = G(\mathbf{X}) = g^{-1}(H(\mathbf{X})),$$

where $g : \mathbb{R} \to \mathbb{R}$ is invertible the link function and $H : \mathbb{R}^p \to \mathbb{R}^1$ is same as the one in latent variable model and $G = g^{-1} \circ H$.

Equivalence of those two models

Under certain conditions, those two models are equivalent.

- **Model I:** $Y = sign(Y^* - \theta)$, where $Y^* = H(\mathbf{X}) + \epsilon$, and ϵ follows some distribution function F_ϵ.
- **Model II:** $E(Y|\mathbf{X}) = g^{-1}(H(\mathbf{X}))$.

Note that in the sufficient dimension reduction setting, we also assume that H depends on $\mathbf{X}$ only through d linear combination, $\boldsymbol{\eta}^{\mathbf{T}}\mathbf{X}$. Therefore, the specific form of H will be

$$H(\mathbf{X}) = H(\boldsymbol{\eta}^{\mathbf{T}}\mathbf{X}) = H(\eta_1^T\mathbf{X}, \ldots, \eta_d^T\mathbf{X}).$$

Based on the latent variable model, we have

$$G_{latent}(\mathbf{X}) = 1 - F_\epsilon(-H(\mathbf{X}) - \theta),$$

Based on the link function model, we have

$$G_{link}(\mathbf{X}) = g^{-1}(H(\mathbf{X})).$$

If we assume that those two models are identical to each other,

$$G_{latent}(x) = G_{link}(x) \Rightarrow g^{-1}(x) = 1 - F_{\epsilon}(-x - \theta).$$

For example, if we assume $\theta = 0$ and ϵ is symmetric, then we could further simplify the relation,

$$\begin{aligned} g^{-1}(x) &= 1 - F_{\epsilon}(-x) \\ &= F_{\epsilon}(x) \end{aligned}$$

So we can have $F_{\epsilon}(\cdot) = g^{-1}(\cdot)$.

1.2.2 Issue of binary response

In general, the issue of binary data analysis comes from the limited information contained in the two-level response. For the SDR methods, the binary response reduces the information contained in the inverse moments, so it is challenging to recover the central subspace based on the inverse moments. The SDR methods based on the first inverse moment is affected the most by the binary response because they can only detect at most one basis of the central basis. The SDR method based on the second moment is not as bad as the first moment, but we find certain situations in which that method may not work well. We will review and discuss this issue with more details Section 3.1.

1.3 Existing solution and its limitations

1.3.1 Probability-enhanced SDR methods (PRE-SDR)

In (Shin et al., 2014), the authors proposed a probability-enhanced SDR method to overcome the issue caused by binary responses. The main idea is to slice the data based on $G(\mathbf{X}) = \mathcal{P}(Y = 1|\mathbf{X})$ instead of $Y \in \{0, 1\}$. Since $G(\mathbf{X})$ is continuous, the inverse moments calculated via $G(\mathbf{X})$ should contain more information than the moments calculated by binary responses. First of all, let's recall a lemma in (Shin et al., 2014)[1], which is an important property of $\mathcal{S}_{Y|\mathbf{X}}$ when Y is binary.

Lemma 1.3.1. *(Shin et al., 2014)[1]*

$$S_{Y|\mathbf{X}} = S_{G(\mathbf{X})}$$

Basically, the Lemma 1.3.1 indicates the central subspace of Y and $G(\mathbf{X})$ are identical, which verifies the advantage of using $G(\mathbf{X})$.

Slicing via WSVM

Although the $\mathcal{P}(Y = 1|\mathbf{X} = x)$ can be used for dimension reduction and it contains more information than a binary response, it is not available in most of the cases and needs to be estimated. However, based on the slicing procedure, we don't need to estimate the exact conditional probability for each observation. What we need is their relative order. Therefore, (Shin et al., 2014) have proposed three different slicing methods based on the Weight Support Vector Machine (WSVM). In this thesis, we only review one of them named PRE-SIR$_1$, which

is also recommended by the authors. Note that in the rest of the thesis, we write it as PRE-SIR. The slicing procedure of PRE-SIR is following: Based on a fixed grid $0 < \pi_1 < \cdots < \pi_h < \cdots < \pi_{H-1} < 1$, we fit the WSVM repeatedly. The hth slice is defined as

$$I_{PRE}^{(h)} := S_{PRE}^{(h)} \backslash S_{PRE}^{(h-1)}, h = 1, \ldots, H,$$

where $\pi_h, h = 1 \ldots, H$ are the weights for different slices and $\hat{l}_h(\cdot)$ is the WSVM solution of slice h and $S_{PRE}^{(l)} = \left\{ i : \hat{l}_{\pi_h}(x_i) < 0 \right\}$ for $h = 1, 2, \cdots, H-1$. After have the slices $I_{PRE}^{(h)}, h = 1 \ldots, H$, we could apply the SDR methods such as SIR and SAVE from Section 1.1.

1.3.2 Limitations of PRE-SDR

Time consuming

Shin's method can work well on moderate dataset, e.g $N < 1000$ and $p < 30$. However, it could be slow when $N > 10^4$. One reason is that PRE needs to run WSVM's algorithm repeatedly for determining the index for each slice, which could be around $10 - 20$ times. Moreover, it also needs to tune several parameters in order to have better performance (see details in Shin et al., 2014). Besides, WSVM itself is computationally intensive. For instance, the time complexity of WSVM could be about $\mathcal{O}(n^2)$ or even $\mathcal{O}(n^3)$ for certain kernels. Therefore, this method is not scalable with large data.

Another limitation of the method is how to determine the structural dimension. For the PRE-SDR method, it uses cumulative ratios of the eigenvalues for choosing the dimension d. It selects the first d largest eigenvalues based on a pre-specified cutoff value. It is a convenient

ad-hoc procedure to decide the d and works well in many cases. However, the cutoff ratio may be varied for different situations, so the choice of cutoff value may have a significant influence on the final result. Based on the simulation studies, the cutoff ratio is sensitive to N and p, so in order to detect the correct number of directions, we need to choose the cutoff value carefully for different data set.

1.4 Proposed big data solutions for SDR methods

Motivated by the existing solution and its limitation on big data, we develop two related solutions of SDR methods on big data. One solution aims to solve the computational issues caused by large data set, which is the online algorithms for SIR and SAVE. The online algorithm can work with large data that the original SDR methods cannot handle. More details can be found in Chapter 2. Another solution aims to take advantage of the huge amount of information provided by large data. More specifically, we establish a procedure to efficiently summarize information from the original data and, at the same time, reduce the sample size. It can be shown that the summarized information can improve the performance of SDR methods on binary response. See details in Chapters 3 and 4. Note that there are efforts that have been made in improving the performance of SDR methods on large data. For instance, Kevin, 2014 has uploaded an on-going paper on SDR methods in big data. However, we do not find any published or finished version of it. Therefore, we decide to work in this direction and try to make our contribution to it.

CHAPTER 2

AN ONLINE ALGORITHM FOR SIR AND SAVE

In this chapter, we introduce online algorithms for both SIR and SAVE to overcome computer memory shortage caused by big data. The main idea is to divide original data into chunks and then calculate the sufficient statistics for each chunk, in the end, aggregate all the statistics together to get the final result for SIR and SAVE. One advantage of the algorithm is that its result is as same as the result of using the full dataset at once. Another advantage is that it can run in a parallel computation framework, which reduces the running time of the SDR method for a large data set. The details of the algorithm could be found in Sections 2.2 and 2.3.

2.1 Online algorithm

For big data analysis, one of the issues is that the computer may not be able to load all the data into its memory. Therefore, it is impossible to apply any dimension reduction methods on the data. A solution to reduce the usage of memory is the online algorithm. The "online" algorithm refers to an algorithm that can process its input piece by piece. Zhang and Yang, 2016 has proposed an algorithm for principal component analysis (PCA) with big data. The general idea of the algorithm is to divide the data into smaller sub-datasets (chunks), which can be handled by our computer, and then calculate the intermediate results for each block, in the end, combine all the intermediate results to get the final result. The intermediate results are recorded as sufficient statistics for the method. Motivated by the idea of processing data

by chunks, we drive the sufficient statistics and develop online algorithms for SIR and SAVE, so that they can work with any big data.

Before we go to the details of the algorithms, let us introduce some notations. Assume that the $\mathbf{X}$ is a $\mathsf{N} \times \mathsf{p}$ matrix. The N observations have been split into L chunks randomly. We have

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_l \\ \vdots \\ \mathbf{X}_L \end{bmatrix},$$

Where X_l is $\mathsf{N}_l \times \mathsf{p}$ matrix and N_l is total number of observations in chuck l. Alternatively, recall in Section 1.1, we could also split the data into slices based on their response,

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_h \\ \vdots \\ \mathbf{X}_H \end{bmatrix},$$

X_h is $N_h \times p$ matrix and N_h is total number of observations in slice h. For each chuck of data it may contain observations from different slices, thus we have

$$\mathbf{X}_h = \begin{bmatrix} \mathbf{X}_{h1} \\ \mathbf{X}_{h2} \\ \vdots \\ \mathbf{X}_{hL} \end{bmatrix} \text{ or } \mathbf{X}_l = \begin{bmatrix} \mathbf{X}_{1l} \\ \mathbf{X}_{2l} \\ \vdots \\ \mathbf{X}_{Hl} \end{bmatrix},$$

where $\mathbf{X}_{hl}^T$ is a $N_{hl} \times p$ matrix and N_{hl} is the number of observations in the hth slice within chunk l. The goal of the online algorithm is to sequentially load X_l into a computer and calculate the result for SIR and SAVE. Note that we assume we have generated the index of slices based on Y, which is a vector.

2.2 An online algorithm for SIR

2.2.1 Calculate M_{SIR} by block

Recall that the candidate matrix of SIR (Equation (1.11))

$$\widehat{\mathbf{M}}_{SIR} = \sum_{h=1}^{H} \hat{f}_h \left(\bar{\mathbf{X}}_h - \bar{\mathbf{X}}\right)\left(\bar{\mathbf{X}}_h - \bar{\mathbf{X}}\right)^T \text{ and } S_{SIR} = \hat{\mathbf{\Sigma}}_X^{-1} \mathrm{Span}(\widehat{\mathbf{M}}_{SIR}).$$

Based on the equation above, we only need several statistics to calculate $\mathbf{M}_{SIR}$, which are

(i) $\bar{\mathbf{X}}_h$, the sample averages for slice $h, h = 1, \ldots, H$

(ii) $\bar{\mathbf{X}}$, the overall sample mean

(iii) $\hat{\Sigma}_X$ the sample covariance

(iv) $\hat{f}_h$ proportion for each slice.

Since those four statistics are straightforward, we can calculate by scan the data chunk by chunk.

Calculate $\bar{X}_h$ by chunks

Based on the layout of $\mathbf{X}$ mentioned in Section 2.1, the $\bar{\mathbf{X}}_h$ could be rewritten as following:

$$\begin{aligned}
\bar{X}_h &= X_h^T \frac{\mathbb{1}_{N_h}}{N_h} \\
&= \frac{1}{N_h} \begin{bmatrix} \mathbf{X}_{h1}^T & \dots & \mathbf{X}_{hL}^T \end{bmatrix} \begin{bmatrix} \mathbb{1}_{N_{h1}} \\ \vdots \\ \mathbb{1}_{N_{hL}} \end{bmatrix} \\
&= \frac{1}{N_h} \sum_{l=1}^{L} (\mathbf{X}_{hl}^T \mathbb{1}_{N_{hl}}),
\end{aligned}$$

where $\mathbb{1}_n$ is a vector of n $1's$, $\mathbf{X}_h$ is a $N_h \times p$ matrix, which is all the observations in slices h and N_{hl} is the number of observations in chunk l and slice h.

Calculate $\bar{X}$ by chunks

Similar with the $\bar{\mathbf{X}}_h$, we could get overall mean by scanning data by chunk

$$\begin{aligned}\bar{X} &= X^T \frac{\mathbb{1}_N}{N} \\ &= \frac{1}{N}\begin{bmatrix}\mathbf{X}_{11}^T & \dots & \mathbf{X}_{HL}^T\end{bmatrix}\begin{bmatrix}\mathbb{1}_{N_{11}} \\ \vdots \\ \mathbb{1}_{N_{HL}}\end{bmatrix} \\ &= \frac{1}{N}\sum_{h=1}^{H}\sum_{l=1}^{L}(\mathbf{X}_{hl}^T\mathbb{1}_{N_{hl}}),\end{aligned}$$

Calculate $\hat{\Sigma}_X$ by chunks

$$\begin{aligned}\hat{\mathbf{\Sigma}}_X &= \frac{1}{N-1}(\sum_{i=1}^{N}\mathbf{X}_i\mathbf{X}_i^T - N\bar{\mathbf{X}}\bar{\mathbf{X}}^T) \\ &= \frac{1}{N-1}(\sum_{h=1}^{H}\sum_{l=1}^{L}(\mathbf{X}_{hl}^T\mathbf{X}_{hl}) - N\bar{\mathbf{X}}\bar{\mathbf{X}}^T)\end{aligned}$$

Calculate $\hat{f}_h$ by chunks

$$\hat{f}_h = \frac{N_h}{N} = \frac{\sum_l N_{hl}}{\sum_{h=1}^{H}\sum_{l=1}^{L} N_{hl}}.$$

2.2.2 Sequential test of SIR by block

In (Li, 1991), the authors proposed a chi-square test to decide the structural dimension estimated by SIR. Based on the test, the only statistic we need is the eigenvalues of $\hat{\mathbf{M}}_{SIR}$, λ_i, and the slice number N_h. After we get the candidate matrix by the online algorithm, we have already got the eigenvalues and the observations for each slice. Therefore, the output of the online algorithm can be used for the sequential test of SIR directly.

2.2.3 Algorithm for SIR

Sufficient Statistics of SIR: $\mathcal{C}_{SIR}$

Based on the previous section, we do not keep all the data to get the candidate matrix. It is enough to store several statistics for each slice. Those statistics are

$$\mathcal{C}_{SIR} = \{C_{N_h} = \sum_l N_{hl},\ C_{S_h} = \sum_l (\mathbf{X}_{hl}^T \mathbb{1}_{N_{hl}}),\ C_{X_h} = \sum_l (\mathbf{X}_{hl}^T \mathbf{X}_{hl}),\ h = 1, \dots H.\} \tag{2.1}$$

Algorithm 1 An online Algorithm of SIR

procedure (Calculate $\hat{\mathbf{M}}_{\mathsf{SIR}}$ and $\hat{\boldsymbol{\eta}}_{\mathsf{SIR}}$)

  Input: $\mathbf{X}_{\mathtt{l}}$

  Output: $\hat{\boldsymbol{\eta}}_{\mathsf{SIR}}$

  Let $\mathsf{C}_{\mathsf{N}_{\mathtt{h}}}, \mathsf{C}_{\mathsf{S}_{\mathtt{h}}}, \mathsf{C}_{\mathsf{X}_{\mathtt{h}}}$ be scalars, vectors and matrices with all initial values as zero

  for for $\mathtt{l}\mathtt{t}\mathtt{h}$ chunk of data **do**

    for for $\mathtt{h}\mathtt{t}\mathtt{h}$ slice of the chunk data **do** Update

      $\mathsf{C}_{\mathsf{N}_{\mathtt{h}}} = \mathsf{C}_{\mathsf{N}_{\mathtt{h}}} + \mathsf{N}_{\mathtt{h}\mathtt{l}}$,

      $\mathsf{C}_{\mathsf{S}_{\mathtt{h}}} = \mathsf{C}_{\mathsf{S}_{\mathtt{h}}} + \mathbf{X}_{\mathtt{h}\mathtt{l}}^{\mathsf{T}} \mathbb{1}_{\mathsf{N}_{\mathtt{h}\mathtt{l}}}$,

      $\mathsf{C}_{\mathsf{X}_{\mathtt{h}}} = \mathsf{C}_{\mathsf{X}_{\mathtt{h}}} + \mathbf{X}_{\mathtt{h}\mathtt{l}}^{\mathsf{T}} \mathbf{X}_{\mathtt{h}\mathtt{l}}$

    end for

  end for

  Calculate $\hat{\mathsf{M}}_{\mathsf{SIR}}$ based Section 2.2.1

  Calculate $\hat{\boldsymbol{\eta}}_{\mathtt{i}}, \mathtt{i} = 1, \ldots, \mathtt{d}$ based on Equation (1.11)

  Select the $\mathtt{d}$ $\hat{\boldsymbol{\eta}}_{\mathtt{i}}$'s based on the sequential test

  Calculate $\hat{\boldsymbol{\eta}}_{\mathsf{SIR}}$

end procedure

2.3 An online algorithm for SAVE

Similar to SIR, we also develop an online algorithm for SAVE.

2.3.1 Calculate M_{SAVE} by block

Recall the candidate matrix of SAVE is

$$\widehat{\mathbf{M}}_{SAVE} = \sum_{h=1}^{H} \hat{f}_h \left(\widehat{\mathbf{\Sigma}}_{\mathbf{X}} - \widehat{\mathbf{\Sigma}}_{\mathbf{X}|h}\right)^2,$$

Based on the equation above, we only need several statistics to calculate $\mathbf{M}_{SIR}$, which are

(i) $\widehat{\mathbf{\Sigma}}_{\mathbf{X}|h}$ the sample covariance for slice $h, h = 1, \ldots, H$

(ii) $\hat{\Sigma}_X$ the sample covariance

(iii) $\hat{f}_h$ proportion for each slice.

Calculate $\widehat{\Sigma}_{X|h}$ by chunks

Since we have discussed how to calculate $\hat{\mathbf{\Sigma}}_X$ and $\hat{f}_h$ in Section 2.2.1, we only focus calculate the slice covariance matrix.

$$\begin{aligned}
\widehat{\mathbf{\Sigma}}_{\mathbf{X}|h} &= \frac{1}{N_h - 1} \sum_{i}^{N_h} (\mathbf{X}_i - \bar{\mathbf{X}}_h)(\mathbf{X}_i - \bar{\mathbf{X}}_h)^T \\
&= \frac{1}{N_h - 1} (\sum_{i}^{N_h} (\mathbf{X}_i \mathbf{X}_i^T) - N_h \bar{\mathbf{X}}_h \bar{\mathbf{X}}_h^T) \\
&= \frac{1}{N_h - 1} (\sum_{l=1}^{L} \mathbf{X}_{hl}^T \mathbf{X}_{hl} - N_h \bar{\mathbf{X}}_h \bar{\mathbf{X}}_h^T),
\end{aligned}$$

where $\bar{\mathbf{X}}_h^T = \frac{1}{N_h} \mathbf{X}_h^T \mathbb{1}_{N_h} = \frac{1}{N_h} \sum_{l=1}^{L} \mathbf{X}_{hl}^T \mathbb{1}_{N_{hl}}$.

2.3.2 Sequential test of SAVE by block

There are several large-sample tests available for SAVE. Compared to SIR's test, those sequential tests of SAVE are more complicated and not straightforward to be calculated by the piece-by-piece fashion. Therefore, we choose a marginal dimension test from (Cook and others, 2004) because the test is straightforward and its statistics can be calculated based on the online algorithm. However, the procedure is quite tedious, which involves calculate a 3 dimension array for each slice. More details could be found in (Cook and others, 2004).

2.3.3 Algorithm for SAVE

Sufficient Statistics of SIR: $\mathcal{C}_{\mathsf{SAVE}}$

The sufficient statistics for SAVE are actually same as SIR

$$\mathcal{C}_{\mathsf{SAVE}} = \{\mathsf{C}_{\mathsf{N}_h} = \sum_l \mathsf{N}_{hl},\ \mathsf{C}_{\mathsf{S}_h} = \sum_l (\mathbf{X}_{hl}^{\mathsf{T}} \mathbb{1}_{\mathsf{N}_{hl}}),\ \mathsf{C}_{\mathsf{X}_h} = \sum_l (\mathbf{X}_{hl}^{\mathsf{T}} \mathbf{X}_{hl}),\ h = 1, \dots \mathsf{H}.\} \tag{2.2}$$

Algorithm 2 An online Algorithm of SAVE

procedure (Calculate $\hat{\mathbf{M}}_{SAVE}$ and $\hat{\boldsymbol{\eta}}_{SAVE}$)

Input: $\mathbf{X}_l, l = 1 \ldots, L$

Output: $\hat{\boldsymbol{\eta}}_{SAVE}$

Let $C_{N_h}, C_{S_h}, C_{X_h}$ be scalars, vectors and matrices with all initial values as zero

for for lth chunk of data **do**

  for for hth slice of the chunk data **do** Update

    $C_{N_h} = C_{N_h} + N_{hl}$,

    $C_{S_h} = C_{S_h} + \mathbf{X}_{hl}^T \mathbb{1}_{N_{hl}}$,

    $C_{X_h} = C_{X_h} + \mathbf{X}_{hl}^T \mathbf{X}_{hl}$

  end for

end for

Calculate $\hat{\mathbf{M}}_{SAVE}$ based Section 2.3.1

Calculate $\boldsymbol{\eta}_i, i = 1, \ldots, d$ based on Equation (1.14)

Select the d $\boldsymbol{\eta}_i$'s based on the marginal dimension test

Calculate $\hat{\boldsymbol{\eta}}_{SAVE}$

end procedure

2.4 Simulation result

In this section, we compare the empirical computational times between the SIR and SAVE's original algorithm and the online algorithm. We record the running times under different combination of N and p, where $(n, p) \in \{10^4, 10^5, 10^6, 10^7\} \times \{6, 10, 20\}$. As for the algorithms, we compare the original SDR algorithms, online algorithms that load the data chunks sequentially (donated as SIR_online_seq and SAVE_online_seq) and online algorithms which load five chunks simultaneously (donated as SIR_online_5 and SAVE_online_5). Table I reports the running time for different methods. In general, the sequential online algorithm shows little advantage over the original algorithm because of its additional steps for scanning data and aggregating results. The difference between the online and original algorithms becomes obvious when N and p is large. To further speed up the computation, we adapt the parallel computation framework to the online algorithm. Based on the result, it can reduce the running time significantly when N is large. For instance, the parallel online algorithm of SAVE runs seven times faster than the original algorithm when $N = 10^7$ and $p = 20$.

TABLE I: EMPIRICAL COMPUTATIONAL TIME FOR ONLINE ALGORITHM

p	logn	SIR	SAVE	SIR_online_seq	SIR_online_5	SAVE_online_seq	SAVE_online_5
6	4	0.00	0.00	0.00	0.00	0.01	0.00
	5	0.01	0.02	0.02	0.00	0.01	0.00
	6	0.22	0.26	0.18	0.05	0.15	0.05
	7	2.43	3.62	1.61	0.53	1.75	0.53
10	4	0.00	0.00	0.00	0.00	0.00	0.00
	5	0.03	0.03	0.03	0.01	0.03	0.01
	6	0.44	0.57	0.27	0.07	0.30	0.07
	7	3.77	5.40	3.74	0.94	3.64	0.80
20	4	0.01	0.01	0.01	0.00	0.01	0.00
	5	0.07	0.08	0.05	0.02	0.05	0.02
	6	0.99	1.23	0.68	0.18	0.64	0.19
	7	8.93	14.02	6.00	1.83	6.57	1.89

Empirical computational time (in minutes) calculated from 100 independent iterations.
The machine equips Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz and 32GB memory

CHAPTER 3

MEAN REPRESENTATIVE APPROACH FOR SDR (MRDR)

After solving the computational obstacle caused by large data, another interesting question to answer is how we could take advantage of the vast amount of information provided by the massive data. Motivated by Shin's method and its also limitations, we propose a representative approach for estimating the central space for binary response. The proposed method is also based on conditional probability ($\mathsf{G}(\mathbf{X})$) to enrich the information of the binary response. Nevertheless, it estimates $\mathsf{G}(\mathbf{X})$ thought the Mean Representative (MR), which is an efficient summary of the original data. The main idea of the MR approach is to partition the data into several blocks based on its covariates. Then it calculates the sample mean of $\mathbf{X}$ and $\mathbf{Y}$ for each block, which we call the representatives. After summarizing those representatives, we could use them to estimate the central subspace via SDR methods. One advantage of our proposed approach is that it can naturally work with large data (i.e., $\mathsf{N} > 10^4$) because of the partition steps largely reduce the data size. In order to have a better understanding of the proposed method, we first discuss the issue of SDR under binary in Section 3.1. Then we describe the proposed method in Sections 3.2.2 and 3.2.

3.1 Inverse regression on binary response

Before introducing the proposed method, we would like to discuss how a binary response fails inverse-regression SDR methods with more details. We first review the key results of the inverse regression on binary responses from (Cook and Lee, 1999).

In (Cook and Lee, 1999), the properties of SDR methods when the response is binary have been studied. The authors have shown that the inverse moments of a binary response tends to contain less information of the central subspace than a continuous response. Therefore, the SDR methods' results based on inverse regression are likely to be smaller than the central subspace. The followings are discussion about the first and second inverse moments for binary responses.

Let $\mathbf{X}$ be the standardized covariates, so that $\mathsf{E}(\mathbf{X}) = 0$ and $\mathsf{Var}(\mathbf{X}) = \mathrm{I}_p$. Let $\boldsymbol{\nu} = \mathsf{E}(\mathbf{X}|Y = 1) - \mathsf{E}(\mathbf{X}|Y = 0)$ and $\boldsymbol{\Delta} = \mathsf{Var}(\mathbf{X}|Y = 1) - \mathsf{Var}(\mathbf{X}|Y = 0)$. It can be shown that the first inverse moments $(\mathsf{E}(\mathbf{X}|Y = 1), \mathsf{E}(\mathbf{X}|Y = 0))$ are linear functions of $\boldsymbol{\nu}$ and second moments $(\mathsf{Var}(\mathbf{X}|Y = 1), \mathsf{Var}(\mathbf{X}|Y = 0))$ are linear functions of $(\boldsymbol{\nu}, \boldsymbol{\Delta})$. That means all the information that the first and second moments can provide for the $\mathcal{S}_{Y|\mathbf{X}}$ is contained in $\mathcal{S}(\boldsymbol{\nu})$ and $\mathcal{S}(\boldsymbol{\Delta})$.

3.1.1 Limitation of SIR under binary response

For SDR methods which relies on the first moment, the performance of those methods is highly affected by the binary response. Because all the information of central subspace carried by the first inverse moment is equivalent to $\mathcal{S}(\boldsymbol{\nu})$. Therefore, we end up with only one

independent basis for the candidate matrix of the first-inverse moment based SDR methods, M_{1st}. That is

$$\mathcal{S}_{M_{1st}} = \mathcal{S}(\hat{E}(\mathbf{X}|Y=1) - \hat{E}(\mathbf{X}|Y=0)) = \mathcal{S}(\boldsymbol{\nu}).$$

Based on the result of (Cook and Lee, 1999), it can be shown that

$$\mathcal{S}_{SIR} = \mathcal{S}_{M_{1st}} = \mathcal{S}(\boldsymbol{\nu}),$$

where $\boldsymbol{\nu}$ is just a vector. Therefore, it's obvious that SIR can only find one direction at most.

3.1.2 Limitation of SAVE under a binary response

For methods use the second inverse moment, we have

$$\mathcal{S}_{M_{2nd}} = \mathcal{S}(\boldsymbol{\Delta}),$$

where $\mathcal{S}_{M_{2nd}}$ is the central subspace (in population level) of SDR methods using the second inverse moment, which has more information about the central subspace compared to the first inverse moment. In certain cases ((Cook and Lee, 1999)), we could have $\mathcal{S}(\boldsymbol{\nu}) = \mathcal{S}(\boldsymbol{\Delta}) = \mathcal{S}_{Y|\mathbf{X}}$, which means we could recover the central subspace via $\boldsymbol{\Delta}$ itself. However, in other situations, the binary response still affects the $\mathcal{S}(\boldsymbol{\Delta})$. For example, it is possible that $\boldsymbol{\Delta} = 0$ or $\mathtt{rank}(\mathcal{S}(\boldsymbol{\Delta})) < \mathtt{rank}(\mathcal{S}_{Y|\mathbf{X}})$. In both situations, we cannot recover the full central subspace based on the second moment. That is, $\mathcal{S}(\boldsymbol{\Delta}) \subset \mathcal{S}_{Y|\mathbf{X}}$. Note that the influence of a binary response is smaller for the higher-order inverse moments than the lower-order moments. Therefore, the SDR methods

based on higher moments should work better for the binary response. However, the methods using higher-order moment are more complicated and time-consuming than the lower-moments based methods. In practice, considering the computational efficiency under large datasets, we focus on the methods that use up to the second moments, like SAVE. We will discuss the details in the following sections and chapters.

In (Cook and Lee, 1999), we have

$$\mathcal{S}_{\mathsf{SAVE}} = \mathcal{S}(\boldsymbol{\nu}, \boldsymbol{\Delta}).$$

Moreover, if the conditional distribution of Y given $\mathbf{X}$ is normal, then we can have

$$\mathcal{S}_{\mathsf{SAVE}} = \mathcal{S}_{Y|\mathbf{X}}.$$

With linearity and constant variance conditions (Cook and Lee, 1999), Cook has proved that SAVE can recover the central subspace better than several other SDR methods such as SIR, principal Hessian direction(PHD,Li, 1992), and Difference of Covariance (DOC). Therefore, SAVE seems to be a good candidate for dimension reduction under binary response. However, even for SAVE, it may be suffered from limited information provided by the binary responses in certain situations. We discuss our findings of those situations in the following sections.

Joint distribution of X and Y

We assume the joint distribution of Y and $\mathbf{X}$ exists and the joint distribution has a density $f_{\mathbf{X},Y}(\mathbf{x}, y)$, which can be written as marginal multiply conditional density as the following:

$$f_{\mathbf{X},Y}(\mathbf{x}, y) = f_X(\mathbf{x}) f_{Y|\mathbf{X}}(y|\mathbf{x}) = f_Y(y) f_{\mathbf{X}|Y}(\mathbf{x}|y)$$

Since Y is a binary response, we have

$$f_{Y|\mathbf{X}}(y = 1|\mathbf{X} = \mathbf{x}) = G(\mathbf{x}), \quad f_{Y|\mathbf{X}}(y = 0|\mathbf{X} = \mathbf{x}) = 1 - G(\mathbf{x}),$$

Therefore, we have

$$f_{\mathbf{X},Y}(\mathbf{x}, y = 1) = f_X(\mathbf{x})G(\mathbf{x}), \quad f_{\mathbf{X},Y}(\mathbf{x}, y = 0) = f_X(\mathbf{x})(1 - G(\mathbf{x}))$$

The conditional distribution on Y is,

$$f_{\mathbf{X}|Y}(\mathbf{x}|y) = \frac{f_{\mathbf{X},Y}(\mathbf{x}, y)}{f_Y(y)} = \begin{cases} \frac{f_X(\mathbf{x})G(\mathbf{x})}{\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}, & y = 1 \\ \frac{f_X(\mathbf{x})(1-G(\mathbf{x}))}{\int f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x}}, & y = 0 \end{cases}$$

When $\Delta = 0$

$$\Delta = Var(\mathbf{X}|Y = 1) - Var(\mathbf{X}|Y = 0) = 0 \Rightarrow Var(\mathbf{X}|Y = 1) = Var(\mathbf{X}|Y = 0)$$

So we have,

$$S(\boldsymbol{\Delta}, \boldsymbol{\nu}) = S(\boldsymbol{\nu}),$$

which means that SAVE can only, at most, find one direction which is same as the $\boldsymbol{\nu}$. Based on the decomposition of the variance term,

$$Var(\mathbf{X}|Y = 1) - Var(\mathbf{X}|Y = 0) = E(\mathbf{X}^2|Y = 1) - E(\mathbf{X}|Y = 1)^2 - E(\mathbf{X}^2|Y = 0) + E(\mathbf{X}|Y = 0)^2,$$

a sufficient condition for $\boldsymbol{\Delta} = 0$ is

$$E(\mathbf{X}^2|Y = 1) = E(\mathbf{X}^2|Y = 0) \text{ and } E(\mathbf{X}|Y = 1) = E(\mathbf{X}|Y = 0),$$

Next, we apply the joint distribution

$$\begin{aligned} E(\mathbf{X}|Y = 1) = E(\mathbf{X}|Y = 0) &\Rightarrow \int x f_{\mathbf{X}|Y}(\mathbf{x}|y = 1)d\mathbf{x} = \int x f_{\mathbf{X}|Y}(\mathbf{x}|y = 0)d\mathbf{x} \\ &\Rightarrow \frac{\int x f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}{\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}} = \frac{\int x f_X(\mathbf{x})(1 - G(\mathbf{x}))d\mathbf{x}}{\int f_X(\mathbf{x})(1 - G(\mathbf{x}))d\mathbf{x}} \end{aligned}$$

$$\begin{aligned} E(\mathbf{X}^2|Y = 1) = E(\mathbf{X}^2|Y = 0) &\Rightarrow \int x^2 f_{\mathbf{X}|Y}(\mathbf{x}|y = 1)d\mathbf{x} = \int x^2 f_{\mathbf{X}|Y}(\mathbf{x}|y = 0)d\mathbf{x} \\ &\Rightarrow \frac{\int x^2 f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}{\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}} = \frac{\int x^2 f_X(\mathbf{x})(1 - G(\mathbf{x}))d\mathbf{x}}{\int f_X(\mathbf{x})(1 - G(\mathbf{x}))d\mathbf{x}} \end{aligned}$$

After some algebra, we have

$$\frac{\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}{\int f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x}} = \frac{\int \mathbf{x} f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}{\int \mathbf{x} f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x}} = \frac{\int \mathbf{x}^2 f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}{\int \mathbf{x}^2 f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x}} \tag{3.1}$$

One sufficient condition of Equation (3.1) is

$$f_X(-\mathbf{x}) = f_X(\mathbf{x}) \text{ and } G(-\mathbf{x}) = 1 - G(\mathbf{x}). \tag{3.2}$$

Plug in the Equation (3.2) into Equation (3.1), we have

$$\begin{aligned}
\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x} &= \int f_X(-x)(G(-x))d\mathbf{x} = \int f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x} \\
\int \mathbf{x} f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x} &= \int -\mathbf{x} f_X(-x)(G(-x))d\mathbf{x} = \int \mathbf{x} f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x} \\
\int \mathbf{x}^2 f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x} &= \int \mathbf{x}^2 f_X(-x)(1-G(-x))d\mathbf{x} = \int \mathbf{x}^2 f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x}.
\end{aligned}$$

The condition $f_X(\mathbf{x}) = f_X(-\mathbf{x})$ is equivalent to $\mathbf{X} \overset{d}{\sim} -\mathbf{X}$. Under the latent variable model introduced in Section 1.2, $G(\mathbf{x}) = 1 - F_\epsilon(-H(\mathbf{x})) = F_\epsilon(H(\mathbf{x}))$ with a symmetric distributed ϵ. If we further assume that $H(-\mathbf{x}) = -H(\mathbf{x})$, then we have $G(\mathbf{x}) = 1 - F_\epsilon(-H(\mathbf{x})) = 1 - F_\epsilon(H(-\mathbf{x})) = 1 - G(-\mathbf{x})$. Therefore, a more specific sufficient condition for $\boldsymbol{\Delta} = 0$ is $\mathbf{X}$ follows a symmetric distribution and $H(x)$ is an odd function.

When $\Delta \neq 0$, but Δ is not full rank

Let assume $\mathbf{X}$ has following properties,

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_p \end{bmatrix} \qquad \mathbf{x}_i \perp\!\!\!\perp \mathbf{x}_j, \forall i \neq j \qquad E(\mathbf{X}) = 0.$$

So the conditional covariance between $\mathbf{X}_i$ and $\mathbf{X}_j$ will be,

$$\begin{aligned} Cov(\mathbf{x}_i, \mathbf{x}_j|Y=1) &= E(\mathbf{x}_i\mathbf{x}_j|Y=1) - E(\mathbf{x}_i|Y=1)E(\mathbf{x}_j|Y=1) \\ &= \int \mathbf{x}_i\mathbf{x}_j f_{\mathbf{X}|Y}(\mathbf{x}, y=1)d\mathbf{x} - \int \mathbf{x}_i f_{\mathbf{X}|Y}(\mathbf{x}, y=1)d\mathbf{x} \int \mathbf{x}_j f_{\mathbf{X}|Y}(\mathbf{x}, y=1)d\mathbf{x} \\ &= \int \mathbf{x}_i\mathbf{x}_j \frac{f_X(\mathbf{x})G(\mathbf{x})}{\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}d\mathbf{x} - \int \mathbf{x}_i \frac{f_X(\mathbf{x})G(\mathbf{x})}{\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}d\mathbf{x} \int \mathbf{x}_j \frac{f_X(\mathbf{x})G(\mathbf{x})}{\int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}}d\mathbf{x} \\ &= \frac{1}{p}\left(\int \mathbf{x}_i\mathbf{x}_j f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x} - \int \mathbf{x}_i f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x} \int \mathbf{x}_j f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}\right), \end{aligned}$$

where $p = \int f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}$. We have similar result for $Y = 0$

$$\begin{aligned} Cov(\mathbf{x}_i, \mathbf{x}_j|Y=0) &= E(\mathbf{x}_i\mathbf{x}_j|Y=0) - E(\mathbf{x}_i|Y=1)E(\mathbf{x}_j|Y=0) \\ &= \frac{1}{(1-p)}\left(\int \mathbf{x}_i\mathbf{x}_j f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x} - \int \mathbf{x}_i f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x} \int \mathbf{x}_j f_X(\mathbf{x})(1-G(\mathbf{x}))d\mathbf{x}\right) \\ &= \frac{1}{(1-p)}\left(-\int \mathbf{x}_i\mathbf{x}_j f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x} - \int \mathbf{x}_i f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x} \int \mathbf{x}_j f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}\right). \end{aligned}$$

If we further assume that $p = 1 - p = 0.5$, we have

$$\Delta_{ij} = Cov(x_i, x_j|Y = 1) - Cov(x_i, x_j|Y = 0) = \frac{2}{p}\int x_i x_j f_X(\mathbf{x})G(\mathbf{x})d\mathbf{x}.$$

Next, Let's take some baby steps to find situations where the conditional covariates are same for some covariates but different for others, so that we could have a Δ which is not full rank. Note that those conditions are some sufficient conditions under which the performance of SAVE will be affected by the binary response.

For $H(\mathbf{x})$, we assume that only $\eta^T\mathbf{x}$ matters

$$H(\mathbf{x}) = H(\eta^T\mathbf{x}).$$

To simply the situation, we let $\eta = (e_1, \ldots, e_d)$, where e_i is an element vector with its ith element as 1 and other elements as 0. Therefore, we have

$$\eta^T\mathbf{x} = (x_1, \ldots, x_d), \text{ where } d < p,$$

where $x_i, i \in \{1, \ldots, d\}$ are first d element of $\mathbf{x}$ If we also assume a symmetric distribution for F_ϵ, then we have

$$G(\mathbf{x}) = F_\epsilon(H(\eta^T\mathbf{x})).$$

If $\mathbf{x}_i$ is not in $\{\mathbf{x}_1, \ldots, \mathbf{x}_d\}$, then the $\boldsymbol{\Delta}_{ij} = 0 \; \forall \; j \neq i$. That is

$$\begin{aligned}
\boldsymbol{\Delta}_{ij} &= \frac{2}{p} \int \mathbf{x}_i \mathbf{x}_j f_X(\mathbf{x}) G(\mathbf{x}) d\mathbf{x} \\
&= \frac{2}{p} \int \mathbf{x}_i f_{X_i}(\mathbf{x}_i) d\mathbf{x} \int \mathbf{x}_j \prod_{j \neq i} f_{X_i}(\mathbf{x}_i) G(\boldsymbol{\eta}^T x) d\mathbf{x} \\
&= 0
\end{aligned}$$

Therefore, we could show that the $rank(\boldsymbol{\Delta}) \leq d$. If $rank(\boldsymbol{\Delta}) = d$, we still have potential to recover the whole central subspace. But if $0 < rank(\boldsymbol{\Delta}) < d$, then we can only recover part of the central subspace by using the $\boldsymbol{\Delta}$. A sufficient situation for $0 < rank(\boldsymbol{\Delta}) < d$ is

$$\exists \; i < d \; \forall \; j, \text{ such that } \boldsymbol{\Delta}_{ij} = 0.$$

For example, let $G(\mathbf{x}) = (\mathbf{x}_1)^2 \cdot \sin(\mathbf{x}_2) \cdot \exp(\mathbf{x}_3)$, then the true direction of the central subspace is $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$, so $d = 3$. However, it can be shown that $rank(S(\boldsymbol{\Delta}, \mathbf{v})) = 2 < 3$ because the $\sin(\mathbf{x}_2)$ is an odd function. Therefore, SAVE can only find 2 directions at the most. Besides, since $(\mathbf{x}_1)^2$ is symmetric with 0, SIR also can only find 2 directions. The details of this case can be found in Section 5.3.

3.2 Proposed approach

3.2.1 Motivation

All of this issues we mentioned before is caused by the limited information of the inverse regression on binary response. Shin et al., 2014 has proved that if we use the conditional

probability, $G(\mathbf{X})$, to replace the binary response, then the issue are solved. The justification of using $G(\mathbf{X})$ is the identity between the central subspace of $Y|\mathbf{X}$ and $G(\mathbf{X})|\mathbf{X}$, which is mentioned in Section 1.3. The identity of the central subspaces can be explained in the following way. Based on the assumption of SDR, we have

$$Y = f(\mathbf{X}, \epsilon) = f(\eta_1^T\mathbf{X}, \ldots, \eta_d^T\mathbf{X}, \epsilon),$$

where f is an arbitrary function connecting Y and $\mathbf{X}$. We assume that f depends $\mathbf{X}$ via the d linear combinations, $\boldsymbol{\eta}$. After we transform the binary response into its conditional probability, we have

$$\mathcal{P}(Y = 1|\mathbf{X}) = G(\mathbf{X}) = G(\eta_1^T\mathbf{X}, \ldots, \eta_d^T\mathbf{X}).$$

The identity of central subspaces implies that the $G(\mathbf{X})$ depends on $\mathbf{X}$ through same d linear combinations as Y does. Therefore, $G(\mathbf{X})$ contains the same information of central subspace as Y does.

An ideal situation to recover the central subspace is to observe the pair $(G(\mathbf{X}), \mathbf{X})$ instead of $(Y, \mathbf{X})$. However, in most of the cases, $G(\mathbf{X})$ is not available and has to be estimated, $\hat{G}(\mathbf{X})$. Intuitively, we want $\hat{G}(\mathbf{X})$ and $G(\mathbf{X})$ to be as close as possible so that SDR methods can have a good estimation of the central subspace. There are many methods to estimate the $\hat{G}(\mathbf{X})$, but we have two requirements on those methods.

1. The method has to be a non-parametric method because SDR methods have few assumptions on the model structure,

2. It can work with massive data sets because of the needs of the large data application.

Those two requirements seem to be contradicted because most of the non-parametric methods are time-consuming when the sample size is large. However, based on the philosophy of SDR, we can estimate the $\boldsymbol{\eta}$ efficiently well as long as the observations can reflect the structure of $\mathsf{G}(\cdot)$ correctly. Therefore, we may only need a few pairs of $(\hat{\mathsf{G}}(\mathbf{X}), \mathbf{X})$ with high quality so that we could use them to recover the central subspace. Moreover, the estimated pairs need not even be the observed points. Motivated by this fact, we develop our proposed method, which will be discussed in detail in the next section.

3.2.2 Representative approach

The Representative approach has two steps, which are the partition step and the summary step. The goal of this method is to extract useful information from the original data, meanwhile reduces the sample size. The advantage of our approach is that it can naturally work with massive data. For the partition step, it reduces the total number of observations from N to K, where K is the number of total blocks and $\mathsf{K} \ll \mathsf{N}$. For the summary step, we select one or more statistics that summarize the information based on our interests, which is $(\hat{\mathsf{G}}(\mathbf{X}), \mathbf{X})$ in our case.

Partition step

The goal of this step is to divide data into blocks based on the similarity of $\mathbf{X}$. There are different definitions of similarity which serves for different research interests. In this paper, we

are interested in estimating $G(\mathbf{x})$ for a given $\mathbf{x}$. So intuitively, we define the similarity of two points as the euclidean distance.

$$d(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{\sum_{j=1}^{p} (x_{1j} - x_{2j})^2},$$

where $\mathbf{x}_i = (x_{i1}, \ldots x_{ip})^T$. Ideally, all the observations inside each block are close to each other.

Partition methods

Many methods are available for partitioning the data. One straightforward method is the binning method. The general idea of this method is to split the original data into small and equal size multi-dimensional intervals (rectangular, cube) so that all the points within each interval will be close to each other. This method is also named as equal-wide binning method. This method is simple and easy to apply, but it may have a disadvantage when data points are not evenly distributed. That is, it could end up with intervals with few or even no points inside, which affects the performances of the following summary step. Besides, it may not work well when p is large. For example, if we split the data into two intervals for each dimension, then we will end up with 2^p blocks. An alternative method is the equal-frequency binning method, which guarantees that each block contains the same number of observations. However, the shape of the multi-dimensional intervals could vary a lot for different data set, which is not preferred.

K-means is a commonly used method of cluster analysis. The goal of this method is to find K groups so that all the points within each block will have the smallest average Euclidean

distance, where the number of K is pre-specified. Compared to the previous methods, K-means does not guarantee equal-width or equal-frequency blocks. However, its blocks are in-between of those two situations. That is, most of its blocks have similar numbers of observations and volume sizes. Besides, K-means algorithm is easy to use and runs relatively fast. Since we only need to estimate the boundaries of the blocks and do not require the algorithm to converge, its time complexity can be $\mathcal{O}(NKp)$, which is linear with N and p.

Notations of partition

Let the feature space partition: B_k, $k = 1, \ldots, K$ such that $\cup_k B_k = \mathcal{X} \subset \mathbb{R}^p$. Let $\nu_k = \int_{B_k} d\mathbf{x}$ be the volume of block B_k. For the corresponding sample partition, we may assume that $\mathcal{X}$ is compact (finite and bounded) such that its complement in $\mathbb{R}^p$ is a negligible set with probability less than $\epsilon > 0$. For the samples we have the index partition: I_k, $k = 1, \ldots, K$. $\cup_k I_k = I = \{1, \ldots, N\}$. Let $N_k = |I_k|$. When a feature space partition $\{B_k, k = 1, \ldots, K\}$ exists, the index partition can be defined as $I_k = \{i \mid \mathbf{x}_i \in B_k\}$, $k = 1, \ldots, K$.

Summary step

In this step, we calculate the representatives for each block. A representative is a summary statistic of all the observations inside a block. For instance, let $\mathbf{x}_i, y_i$'s are observations such that $i \in I_{B_k}$, the representatives are

$$\mathbf{x}_R = s(\mathbf{x}_i), \text{ and } y_R = s(\mathbf{y}_i),$$

where s is a summary function. Based on different goals, we could use different summary functions. In (Li and Yang, 2018), the authors have evaluated and compared several different summary functions, such as mean, median and middle point function. They have suggested to use the mean function as the summary function for regression problem. Let $s(x) = \frac{\sum_i^N \mathbf{x}_i}{N}$, then

$$\mathbf{x}_R = s(\mathbf{x}_i) = \bar{\mathbf{x}}_k, \text{ and } \mathbf{y}_R = S(\mathbf{y}_i) = \bar{\mathbf{y}}_k, , \quad \forall i \in I_{B_k},$$

where $\bar{\mathbf{x}}_k$ and $\bar{\mathbf{y}}_k$ are the averages of observations in the block k and named as the Mean Representatives (MR).

3.2.3 Mean Representative for $G(X)$ estimation

Assumed we have spited the data into K blocks $B_k, k = 1, \ldots, K$. Let the volume of a block be $v_k = \int_{B_k} d\mathbf{x}$. Given Y is a binary random variable, the MR of Y is the estimator of a conditional probability, which can be written as

$$E(\bar{Y}_k) = \mathcal{P}(Y_i = 1 | \mathbf{X}_i \in B_k).$$

On the other hand, the MR of $\mathbf{X}$ will be the estimator of the conditional mean of $\mathbf{X}$ that are restricted in B_k,

$$E(\bar{\mathbf{X}}_k) = E(\mathbf{X}_i | \mathbf{X}_i \in B_k).$$

Besides, those MRs are actually consistent estimator of the conditional expectations. That is given a partition $B_k, k = 1, \ldots, K$, we have

$$\bar{Y}_k \xrightarrow{P} \mathcal{P}(Y_i = 1 | \mathbf{X}_i \in B_k) \text{ and } \bar{\mathbf{X}}_k \xrightarrow{P} E(\mathbf{X}_i | \mathbf{X}_i \in B_k), k = 1, \ldots, K.$$

The choice of the number of blocks

Recall in Section 3.2.1, we want to have pairs of observations like $(G(\mathbf{X}), \mathbf{X})$, so that we could use them to estimate the linear directions via SDR methods. After the partition and summary steps, we transform the binary response into continuous variable, so we have the pairs of MRs, $(\bar{Y}_k = \hat{G}(\bar{\mathbf{X}}_k), \bar{\mathbf{X}}_k)$. However, we still need to be careful before applying the SDR methods on the MRs. Because the $G(\bar{\mathbf{X}}_k)$ are not necessarily close to $\bar{Y}_k$ in most of cases. Actually, the consistency property of MRs indicates

$$\bar{Y}_k - G(\bar{\mathbf{X}}_k) \xrightarrow{P} c, \text{ as } N \to \infty,$$

where c is a non-zero constant.

In order to achieve a better approximation between $\bar{Y}_k$ and $G(\bar{\mathbf{X}}_k)$, we define the number of blocks K as a function of N and p, which is notated as K_N and requires $K_N \to \infty, \frac{K_N}{N} \to 0$, as $N \to \infty$. For the simplicity, we assume a power structure between K_N and N. Under some regular conditions, we have shown that the best choice of K_N is

$$K_N = C_K N^{p/(p+4)}, \tag{3.3}$$

where c_K is a constant. Note that K_N could be large when N is large, which may increase the computational time of the partition step. Therefore, in practice, we may set

$$K_N = \max(C_K N^{p/(p+4)}, K_r),$$

where K_r is the largest number of clusters that can be handled by a computer. More details of clustering method are in Section 7.2.1. Moreover, under the Equation (3.3), we have

$$\bar{Y}_k - G(\bar{\mathbf{X}}_k) \xrightarrow{P} 0, \text{ as } N \to \infty.$$

which we will discuss with details in Chapter 4.

3.2.4 Mean representative approach for SDR methods (MRDR)

The application of MR on the SDR methods is straightforward. The procedure of MRDR-SDR are the following:

1. Apply the K-means method on predictors to partition the data into K_N blocks, $I_{B_1}, \ldots, I_{B_{K_N}}$, where $K_N = C_K N^{p/(p+4)}$.
2. Calculate the MRs for all non-empty blocks. That is $\bar{x}_k, \bar{y}_k, k = \{1, \ldots, K_N\}$, where K_N is the number of blocks when sample size is N. In this step, we transform the binary responses into continuous values between 0 and 1.
3. Apply SDR methods on K_N MRs to estimate a basis of $\mathcal{S}_{Y|\mathbf{X}}$.

Structural dimension estimation based on MRs

For the MRDR method, we directly adopt the tests or procedures to detect the structural dimension from the original SDR methods. Since the MRs are the sample averages, we expect the test statistics based on MRs should maintain similar properties as the test statistics calculated based on original data. Moreover, simulation results show that the large sample tests work well with MRs, especially for SIR. Note that SAVE's test is sensitive to the choice of slices number H, so we need to be careful to choose the numbers of slices and clusters for SAVE.

Tuning parameters of partitioning and slicing

One of the parameters is the constant C_k from Equation (3.3). The large C_K, the more blocks we will end up with, but the less point inside each block. Based on our simulation study, we suggest just set the $C_K = 1$. Since the K-means method does not control the number of observations for each block, some of the blocks may contain only few points in them. That means their mean representatives tend to have large variability. Therefore, we recommend removing some of the small blocks. For example, we could remove the first smallest 5% blocks. More details of choosing C_k are discussed in Section 4.3.

For the slicing procedure, we need to select the number of slice H. The number of H is affected by the number of MRs. In order to have H slices with size N_H, we need the number of unique values of MRs for Y is more than H. Besides, the number of K_N is large enough, so that $K_N/H \approx N_H$. We suggest to control the H or K_N and modify the other value to satisfy a pre-specified restriction. Different SDR methods have different requirements on H. Based on our simulations, we find out that MRDR-SIR will not affect a lot by the number of H or K_N,

which is consistent with the property of SIR because of its simplicity. However, MRDR-SAVE is sensitive to the choice of H and N_K. In (Li et al., 2007), the authors have shown that SAVE is very sensitive to the number of slices. They have suggested $N_H = 20$ based on their simulation studies. Based on our simulation studies on a large dataset, we suggest that $N_H = 100$.

CHAPTER 4

ASYMPTOTIC PROPERTIES OF MEAN REPRESENTATIVE

In this chapter, we study the asymptotic distributions of $\bar{\mathbf{X}}_k$, $\bar{Y}_k$ and $\bar{Y}_k - G(\bar{\mathbf{X}}_k)$. In Section 4.1, we study the asymptotic properties of the mean representatives assuming that the partition is not related to the sample size N, which we call fixed partition. Then, in Section 4.2, we assume the partition is shrinking. When N increases, the number of blocks (N_k) increases, and the volume of each block decreases. Under the shrinkage partition, we prove that the mean representative becomes a better estimation of the conditional probability than when the partition is fixed. Besides, we also discuss a possible optimal relation between N_k and N in Section 4.3.

4.1 Fixed partition

In this section, we consider the situation when a feature partition is given and fixed. The covariates $\mathbf{X}_1, \ldots, \mathbf{X}_N$ are iid from a distribution function F on $\mathbb{R}^p$.

In this case, given the kth block B_k, we denote the kth block sample size $N_k = \sum_{i=1}^{N} \mathbb{1}_{\mathbf{X}_i \in B_k}$. In order to avoid trivial cases, we assume the kth block probability $p_k = \int_{B_k} F(d\mathbf{x}) > 0$.

Denote $Z_i = \mathbb{1}_{\mathbf{X}_i \in B_k}$, $i = 1, \ldots, N$. Then $Z_1, \ldots, Z_N$ are iid from $\mathrm{Bernoulli}(p_k)$, $N_k = \sum_{i=1}^{N} Z_i$, and the kth representative

$$\bar{\mathbf{X}}_k = \frac{\sum_{i=1}^{N} \mathbf{X}_i Z_i}{\sum_{i=1}^{N} Z_i}, \quad \bar{Y}_k = \frac{\sum_{i=1}^{N} Y_i Z_i}{\sum_{i=1}^{N} Z_i}$$

In order to find the asymptotic distribution of the kth representative, we denote $\mathbf{V}_i = (Z_i, Z_i\mathbf{X}_i^T)^T = Z_i(1, \mathbf{X}_i^T)^T \in \mathbb{R}^{p+1}$. Assuming $\mathbf{X}_1, \ldots, \mathbf{X}_N$ are iid from a multivariate distribution with mean $\boldsymbol{\mu} \in \mathbb{R}^p$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{p\times d}$, then $\mathbf{V}_1, \ldots, \mathbf{V}_N$ are iid with mean $\boldsymbol{\mu}_v = p_k(1, \boldsymbol{\mu}_k^T)^T$ and covariance

$$\boldsymbol{\Sigma}_v = \begin{pmatrix} p_k(1-p_k) & p_k(1-p_k)\boldsymbol{\mu}_k^T \\ p_k(1-p_k)\boldsymbol{\mu}_k & p_k(1-p_k)\boldsymbol{\mu}_k\boldsymbol{\mu}_k^T + p_k\boldsymbol{\Sigma}_k \end{pmatrix} = \frac{1-p_k}{p_k}\boldsymbol{\mu}_v\boldsymbol{\mu}_v^T + p_k \begin{pmatrix} 0 & 0 \\ 0 & \boldsymbol{\Sigma}_k \end{pmatrix}$$

where $\boldsymbol{\mu}_k = p_k^{-1}\int_{B_k} \mathbf{x}F(d\mathbf{x})$ and $\boldsymbol{\Sigma}_k = p_k^{-1}\int_{B_k} \mathbf{x}\mathbf{x}^T F(d\mathbf{x}) - \boldsymbol{\mu}_k\boldsymbol{\mu}_k^T$ are the mean and variance of $\mathbf{X}_i$ restricted to block B_k (i.e., with probability measure $p_k^{-1}F$ on B_k).

Besides, to make sure that the random vector $\mathbf{V}$ has a non-degenerate distribution, we need to check if the covariance matrix is positive-definite.

Lemma 4.1.1. *Suppose $\boldsymbol{\Sigma}$ is positive definite. Then $\boldsymbol{\Sigma}_v$ is positive definite if and only if $0 < p_k < 1$ and $\boldsymbol{\Sigma}_k$ is positive definite.*

Note that $\boldsymbol{\Sigma}_k$ is not positive definite only if there exist a nonzero vector $\mathbf{a} \in \mathbb{R}^p$ and a constant $b \in \mathbb{R}$, such that, if $\mathbf{X}_i \in B_k$, then $\mathbf{a}^T\mathbf{X}_i = b$ almost surely. A special case is $X_{ij} \equiv c$ for some j given $\mathbf{X}_i = (X_{i1}, \ldots, X_{id})^T \in B_k$.

Proof of Lemma 4.1.1: For any $b \in \mathbb{R}$ and $\mathbf{a} \in \mathbb{R}^p$,

$$(-b, \mathbf{a}^T)\boldsymbol{\Sigma}_v \begin{pmatrix} -b \\ \mathbf{a} \end{pmatrix} = p_k(1-p_k)\cdot(\mathbf{a}^T\boldsymbol{\mu}_k - b)^2 + p_k\cdot\mathbf{a}^T\boldsymbol{\Sigma}_k\mathbf{a} \tag{4.1}$$

The conclusions can be obtained using the fact that $\mathbf{\Sigma}_\nu$ is positive definite if and only if (4.1)=0 always implies $\mathbf{a} = 0$ and $b = 0$. □

4.1.1 Asymptotic distribution of $\bar{X}_k$

In this section, we consider the cases when F has a density f. Then $p_k = \int_{B_k} f(\mathbf{x})d\mathbf{x}$. In this case, $\mathbf{\Sigma}_\nu$ must be positive definite.

According to the multivariate central limit theorem (see, for example, Theorem 5 in (Ferguson, 1996)),

$$\sqrt{N}(\bar{\mathbf{V}} - \boldsymbol{\mu}_\nu) \xrightarrow{\mathcal{D}} N_{p+1}(\mathbf{0}, \mathbf{\Sigma}_\nu)$$

as $N \to \infty$, where $\bar{\mathbf{V}} = N^{-1}\sum_{i=1}^{N} \mathbf{V}_i$.

Denote the map $M : \mathbb{R}^{p+1} \to \mathbb{R}^p$ such that $M((z, \mathbf{u}^T)^T) = \mathbf{u}/z$, where $z \in \mathbb{R}$ and $\mathbf{u} \in \mathbb{R}^p$. Then $M(\bar{\mathbf{V}}) = \sum_{i=1}^{N} Z_i\mathbf{X}_i / \sum_{i=1}^{N} Z_i = \bar{\mathbf{X}}_k$ and $M(\boldsymbol{\mu}_\nu) = \boldsymbol{\mu}_k$. It can be verified that the gradient of M at $\boldsymbol{\mu}_\nu$ is $\nabla M(\boldsymbol{\mu}_\nu) = p_k^{-1}(-\boldsymbol{\mu}_k, \mathbf{I}_p)^T$, where $\mathbf{I}_p$ is the identity matrix of order d.

According to the multivariate Delta method (see, for example, Theorem 7 in (Ferguson, 1996)),

$$\sqrt{N}(M(\bar{\mathbf{V}}) - M(\boldsymbol{\mu}_\nu)) \xrightarrow{\mathcal{D}} N_p\left(\mathbf{0}, \nabla M(\boldsymbol{\mu}_\nu)^T \cdot \mathbf{\Sigma}_\nu \cdot \nabla M(\boldsymbol{\mu}_\nu)\right) = N_p(\mathbf{0}, p_k^{-1}\mathbf{\Sigma}_k)$$

as $N \to \infty$.

Theorem 4.1.1. *Suppose* $\mathbf{\Sigma}$ *is positive definite,* $p_k \in (0, 1)$*, and* F *has a density. Then*

$$\sqrt{N}(\bar{\mathbf{X}}_k - \boldsymbol{\mu}_k) \xrightarrow{\mathcal{D}} N_p(\mathbf{0}, p_k^{-1}\mathbf{\Sigma}_k)$$

as $N \to \infty$.

4.1.2 Asymptotic distribution of $\bar{Y}_k$

In this section, we consider the asymptotic distribution of $\bar{Y}_k = \sum_{i=1}^{N} Y_i Z_i / \sum_{i=1}^{N} Z_i$ as N goes to infinity. Similar to the procedure for $\bar{\mathbf{X}}_k$, we define $\mathbf{W}_i = Z_i(1, Y_i)^T$, $i = 1, \ldots, N$. Then $\mathbf{W}_1, \ldots, \mathbf{W}_N$ are iid with mean $\boldsymbol{\mu}_w = p_k(1, \mu_g)^T$ and covariance

$$\boldsymbol{\Sigma}_w = \begin{pmatrix} p_k(1-p_k) & p_k(1-p_k)\mu_g \\ p_k(1-p_k)\mu_g & p_k\mu_g(1-p_k\mu_g) \end{pmatrix} = \frac{1-p_k}{p_k}\boldsymbol{\mu}_w\boldsymbol{\mu}_w^T + \begin{pmatrix} 0 & 0 \\ 0 & p_k\mu_g(1-\mu_g) \end{pmatrix}$$

where $\mu_g = p_k^{-1}\int_{B_k} G(\mathbf{x})F(d\mathbf{x}) \in [0,1]$, since $G(\mathbf{x}) = P(Y_i = 1 \mid \mathbf{X}_i = \mathbf{x}) \in [0,1]$.

Since $|\boldsymbol{\Sigma}_w| = p_k^2(1-p_k)\mu_g(1-\mu_g)$, then $\boldsymbol{\Sigma}_w$ is positive definite if and only if $0 < \mu_g < 1$. Note that $\mu_g = 0$ indicates $G(\mathbf{x}) = 0$ almost surely given $\mathbf{x} \in B_k$, while $\mu_g = 1$ indicates $G(\mathbf{x}) = 1$ almost surely within B_k. Both are trivial cases.

When $\boldsymbol{\Sigma}_w$ is positive definite, the multivariate central limit theorem implies

$$\sqrt{N}(\bar{\mathbf{W}} - \boldsymbol{\mu}_w) \xrightarrow{\mathcal{D}} N_2(\mathbf{0}, \boldsymbol{\Sigma}_w)$$

where $\bar{\mathbf{W}} = N^{-1}\sum_{i=1}^{N} \mathbf{W}_i$.

Similar to Section 4.1.1, we denote the map $M : \mathbb{R}^2 \to \mathbb{R}$ such that $M((z,u)^T) = u/z$. Then $M(\bar{\mathbf{W}}) = \sum_{i=1}^{N} Y_i Z_i / \sum_{i=1}^{N} Z_i = \bar{Y}_k$ and $M(\boldsymbol{\mu}_w) = \mu_g$. The gradient of M at $\boldsymbol{\mu}_w$ is $\nabla M(\boldsymbol{\mu}_w) = p_k^{-1}(-\mu_g, 1)^T$, and $\nabla M(\boldsymbol{\mu}_w)^T \cdot \boldsymbol{\Sigma}_w \cdot \nabla M(\boldsymbol{\mu}_w) = p_k^{-1}\mu_g(1-\mu_g)$.

According to the multivariate Delta method, we obtain the theorem of $\bar{Y}_k$ as follows:

Theorem 4.1.2. *Suppose* $0 < p_k < 1$ *and* $0 < \mu_g < 1$. *Then* $\boldsymbol{\Sigma}_w$ *is positive definite and*

$$\sqrt{N}(\bar{Y}_k - \mu_g) \xrightarrow{\mathcal{D}} N\left(0, p_k^{-1}\mu_g(1-\mu_g)\right)$$

as $N \to \infty$.

4.1.3 Asymptotic distribution of $\bar{Y}_k - G(\bar{X}_k)$

In this section, we consider the asymptotic distribution of $\bar{Y}_k - G(\bar{\mathbf{X}}_k)$ as N goes to infinity. Similar to the procedures for $\bar{\mathbf{X}}_k$ and $\bar{Y}_k$, we define $\mathbf{U}_i = Z_i(1, Y_i, \mathbf{X}_i^T)^T \in \mathbb{R}^{p+2}$, $i = 1, \ldots, N$. Then $\mathbf{U}_1, \ldots, \mathbf{U}_N$ are iid with mean $\boldsymbol{\mu}_u = p_k(1, \mu_g, \boldsymbol{\mu}_k^T)^T$ and covariance

$$\boldsymbol{\Sigma}_u = \frac{1-p_k}{p_k}\boldsymbol{\mu}_u\boldsymbol{\mu}_u^T + p_k \begin{pmatrix} 0 & 0 & 0 \\ 0 & \mu_g(1-\mu_g) & \boldsymbol{\Sigma}_{xg}^T \\ 0 & \boldsymbol{\Sigma}_{xg} & \boldsymbol{\Sigma}_k \end{pmatrix}$$

where $\boldsymbol{\Sigma}_{xg} = p_k^{-1}\int_{B_k} \mathbf{x}G(\mathbf{x})F(d\mathbf{x}) - \mu_g\boldsymbol{\mu}_k \in \mathbb{R}^p$.

In order to investigate when $\boldsymbol{\Sigma}_u$ is positive definite, we consider an arbitrary $\mathbf{u} = (-b, -c, \mathbf{a}^T)^T \in \mathbb{R}^{p+2}$ with $b, c \in \mathbb{R}$ and $\mathbf{a} \in \mathbb{R}^p$. It can be verified that

$$\begin{aligned} \mathbf{u}^T\boldsymbol{\Sigma}_u\mathbf{u} &= p_k(1-p_k)\left(\mathbf{a}^T\boldsymbol{\mu}_k - c\mu_g - b\right)^2 \\ &+ \int_{B_k}\left\{\left[\mathbf{a}^T\mathbf{x} - cG(\mathbf{x})\right] - \left(\mathbf{a}^T\boldsymbol{\mu}_k - c\mu_g\right)\right\}^2 F(d\mathbf{x}) \\ &+ c^2\int_{B_k} G(\mathbf{x})[1-G(\mathbf{x})]F(d\mathbf{x}) \end{aligned}$$

Note that $\mathbf{\Sigma}_u$ is positive definite always implies $\mathbf{\Sigma}_w$ and $\mathbf{\Sigma}_v$ are positive definite. Then the positive definiteness of $\mathbf{\Sigma}_u$ implies $0 < p_k < 1$, $0 < \mu_g < 1$ and the positive definiteness of $\mathbf{\Sigma}_k$. However, those conditions are not sufficient for the positive definiteness of $\mathbf{\Sigma}_u$. Below is a counterexample.

Example 4.1.1. *Suppose* $\mathbf{X}_i = (X_{i1}, \ldots, X_{id})^T$ *has independent components. Each of* $X_{i2}, \ldots, X_{id}$ *follows* $N(0,1)$*, while* X_{i1} *is discrete within* B_k *such that* $P(X_{i1} = c_0) = p_0 > 0$ *and* $P(X_{i1} = c_1) = p_k - p_0 > 0$*,* $c_0 \neq c_1$*. Suppose further* $G(\mathbf{x}) \equiv 0$ *if* $\mathbf{x} = (c_0, x_{i2}, \ldots, x_{id})^T \in B_k$ *and* $G(\mathbf{x}) \equiv 1$ *if* $\mathbf{x} = (c_1, x_{i2}, \ldots, x_{id})^T \in B_k$*. It can be verified that if* $0 < p_k < 1$*, then* $\mu_g = (p_k - p_0)/p_k \in (0,1)$ *and* $\mathbf{\Sigma}_k$ *is positive definite. However,* $\mathbf{u}^T\mathbf{\Sigma}_u\mathbf{u} = 0$ *if* $\mathbf{u} = (-c_0, c_0 - c_1, 1, 0, \ldots, 0)^T \in \mathbb{R}^{p+2}$*. That is,* $\mathbf{\Sigma}_u$ *is not positive definite.*

Lemma 4.1.2. *Suppose* $P(\mathbf{X}_i \in \{\mathbf{x} \in B_k \mid 0 < G(\mathbf{x}) < 1\}) > 0$*. Then* $\mathbf{\Sigma}_u$ *is positive definite if and only if* $0 < p_k < 1$ *and* $\mathbf{\Sigma}_k$ *is positive definite.*

Proof of Lemma 4.1.2: We only need to prove the "if" part. Note that the assumption $P(\mathbf{X}_i \in \{\mathbf{x} \in B_k \mid 0 < G(\mathbf{x}) < 1\}) > 0$ implies $\int_{B_k} G(\mathbf{x})[1 - G(\mathbf{x})]F(d\mathbf{x}) > 0$. Since $0 < p_k < 1$, then $\mathbf{u}^T\mathbf{\Sigma}_u\mathbf{u} = 0$ implies $b = \mathbf{a}^T\mathbf{\mu}_k - c\mu_g$, $c = 0$, and $\mathbf{a}^T\mathbf{X}_i = \mathbf{a}^T\mathbf{\mu}_k$ almost surely in B_k. Since $\mathbf{\Sigma}_k$ is positive definite, we must have $\mathbf{a} = 0$ and then $b = 0$. Thus, $\mathbf{\Sigma}_u$ is positive definite. □

Lemma 4.1.3. *Suppose* F *has a density. Then* $\mathbf{\Sigma}_u$ *is positive definite if and only if* $0 < p_k < 1$ *and* $0 < \mu_g < 1$*.*

Proof of Lemma 4.1.3: We only need to show the "if" part. Since $0 < p_k < 1$, then $\mathbf{u}^T\boldsymbol{\Sigma}_u\mathbf{u} = 0$ implies (i) $b = \mathbf{a}^T\boldsymbol{\mu}_k - c\mu_g$; (ii) $c^2\int_{B_k} G(\mathbf{x})[1 - G(\mathbf{x})]F(d\mathbf{x}) = 0$; and (iii) $\mathbf{a}^T\mathbf{X}_i = \mathbf{a}^T\boldsymbol{\mu}_k - c\mu_g + cG(\mathbf{X}_i)$ almost surely in B_k.

If $c \neq 0$, then (ii) implies $G(\mathbf{X}_i) \in \{0, 1\}$ almost surely in B_k. Due to $0 < \mu_g < 1$, if we denote $B_{k0} = \{\mathbf{x} \in B_k \mid G(\mathbf{x}) = 0\}$ and $B_{k1} = \{\mathbf{x} \in B_k \mid G(\mathbf{x}) = 1\}$, then $P(\mathbf{X}_i \in B_{k0}) > 0$, $P(\mathbf{X}_i \in B_{k1}) > 0$, and $P(\mathbf{X}_i \in B_k \setminus (B_{k0} \cup B_{k1})) = 0$. Combining with (iii), we get (iv) $\mathbf{a}^T\mathbf{x} = \mathbf{a}^T\boldsymbol{\mu}_k - c\mu_g$ if $\mathbf{x} \in B_{k0}$ and $\mathbf{a}^T\mathbf{x} = \mathbf{a}^T\boldsymbol{\mu}_k - c\mu_g + c$ if $\mathbf{x} \in B_{k1}$. Since $\mathbf{a}$ can not be zero here, (iv) implies $B_{k0} \cup B_{k1}$ has Lebesgue measure zero and then $p_k = \int_{B_k} f(\mathbf{x})d\mathbf{x} = \int_{B_{k0}\cup B_{k1}} f(\mathbf{x})d\mathbf{x} = 0$, where f is the density of F. The contradiction implies $c = 0$.

Combining with (iii), $c = 0$ implies $\mathbf{a}^T\mathbf{X}_i = \mathbf{a}^T\boldsymbol{\mu}_k$ almost surely in B_k, which violates the existence of the density f unless $\mathbf{a} = 0$. After all, we must have $c = 0$, $\mathbf{a} = 0$ and then $b = 0$. Thus, $\boldsymbol{\Sigma}_u$ is positive definite. □

When $\boldsymbol{\Sigma}_u$ is positive definite, the multivariate central limit theorem implies

$$\sqrt{N}(\bar{\mathbf{U}} - \boldsymbol{\mu}_u) \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0}, \boldsymbol{\Sigma}_u)$$

where $\bar{\mathbf{U}} = N^{-1}\sum_{i=1}^{N}\mathbf{U}_i$.

We denote the map $M : \mathbb{R}^{p+2} \to \mathbb{R}$ such that $M((z, v, \mathbf{u}^T)^T) = v/z - G(\mathbf{u}/z)$ with $z, v \in \mathbb{R}$ and $\mathbf{u} \in \mathbb{R}^p$. Then $M(\bar{\mathbf{U}}) = \bar{Y}_k - G(\bar{\mathbf{X}}_k)$ and $M(\boldsymbol{\mu}_u) = \mu_g - G(\boldsymbol{\mu}_k)$. The gradient of M at $\boldsymbol{\mu}_u$ is $\nabla M(\boldsymbol{\mu}_u) = p_k^{-1} \left(\boldsymbol{\mu}_k^T \cdot \nabla G(\boldsymbol{\mu}_k) - \mu_g, 1, -\nabla G(\boldsymbol{\mu}_k)^T\right)^T$, and

$$\nabla M(\boldsymbol{\mu}_u)^T \cdot \boldsymbol{\Sigma}_u \cdot \nabla M(\boldsymbol{\mu}_u) = p_k^{-1} \left(\mu_g(1 - \mu_g) - 2\nabla G(\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_{xg} + \nabla G(\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k \nabla G(\boldsymbol{\mu}_k)\right)$$

denoted by σ^2_{ygx} .

According to Lemma 4.1.3 and the multivariate Delta method, we obtain the theorem as follows:

Theorem 4.1.3. *Suppose* $0 < p_k < 1$, $0 < \mu_g < 1$, *and* F *has a density. Then*

$$\sqrt{N}\left[\left(\bar{Y}_k - G(\bar{\mathbf{X}}_k)\right) - (\mu_g - G(\boldsymbol{\mu}_k))\right] \xrightarrow{\mathcal{D}} N\left(0, \sigma^2_{ygx}\right)$$

as $N \to \infty$.

As a direct conclusion by Theorem 4.1.3, as N goes to infinity,

$$\bar{Y}_k - G(\bar{\mathbf{X}}_k) \xrightarrow{P} \mu_g - G(\boldsymbol{\mu}_k) = p_k^{-1} \int_{B_k} G(\mathbf{x}) F(d\mathbf{x}) - G\left(p_k^{-1} \int_{B_k} \mathbf{x} F(d\mathbf{x})\right) \tag{4.2}$$

which is typically nonzero unless G is linear or $\mathbf{X}_i$ is a constant almost surely within B_k.

4.2 Shrinking partition

When the block B_k is fixed as $N \to \infty$, according to Theorem 4.1.1, $\bar{Y}_k - G(\bar{\mathbf{X}}_k)$ converges to a nonzero constant (4.2) unless G is linear or $\mathbf{X}_i$ is a constant almost surely within B_k.

In this section, we consider the asymptotic distribution of $\bar{Y}_{k_N} - G(\bar{\mathbf{X}}_{k_N})$ on a sequence of shrinking blocks B_{k_N} as N goes to infinity. More specifically, (1) the number of blocks $K_N \to \infty$ as $N \to \infty$; (2) $1 \le k_N \le K_N$ and B_{k_N}'s are nested, that is, $B_{k_N} \supseteq B_{k_{N+1}}$ for all N; (3) the blocks are shrinking, that is, the sizes of blocks $\delta_{k_N} = \max_{\mathbf{x}_1,\mathbf{x}_2 \in B_{k_N}} \|\mathbf{x}_1 - \mathbf{x}_2\| \to 0$ as $N \to \infty$, where $\|\cdot\|$ stands for Euclidean distance; (4) the blocks are not trivial, that is, each block B_{k_N} contains an open ball $B_\epsilon(\mathbf{x}) = \{\mathbf{x}' \in \mathbb{R}^p \mid \|\mathbf{x}' - \mathbf{x}\| < \epsilon\}$ for some $\mathbf{x} \in \mathbb{R}^p$ and $\epsilon > 0$.

As a matter of mathematical facts, if a sequence of blocks satisfies the above conditions, then there exists a single point $\mathbf{x}_0 \in \mathbb{R}^p$ such that $\lim_{N\to\infty} \bar{B}_{k_N} = \{\mathbf{x}_0\}$ as a limit of sets, where $\bar{B}_{k_N}$ is the closure of B_{k_N}. Actually, $\{\mathbf{x}_0\} = \cap_N \bar{B}_{k_N}$. Note that it is possible that $\mathbf{x}_0 \notin B_{k_N}$ for each N. From the last two conditions we also know that the volumes of the blocks $v_{k_N} = \int_{B_{k_N}} d\mathbf{x} > 0$ for each N and $\lim_{N\to\infty} v_{k_N} = 0$.

Let $\mathbf{x}_0 = (x_{01}, \ldots, x_{0p})^T$. We first consider a simplified block type $B_{k_N} = \prod_{j=1}^p [x_{0j}, x_{0j} + h_N)$, where $h_N = c_h \cdot N^{-1/(pr)}$ for some $c_h > 0$ and $r > 1$. Then the volume $v_{k_N} = h_N^p = c_h^p \cdot N^{-1/r}$ and the number of blocks $K_N \approx c_K \cdot N^{1/r}$ for some $c_K > 0$. On average, the number of points n_{k_N} in B_{k_N} is about $c_n \cdot N^{1-1/r}$ for some $c_n > 0$. Note that both K_N and n_{k_N} go to infinity as N goes to infinity.

Given N and $B_{k_N} \subseteq \mathbb{R}^p$, we define $\mathbf{U}_i = Z_i(1, Y_i, \mathbf{X}_i^T)^T \in \mathbb{R}^{p+2}$, $i = 1, \ldots, N$, where $Z_i = \mathbf{1}_{\mathbf{X}_i \in B_{k_N}}$. Then $\mathbf{U}_1, \ldots, \mathbf{U}_N$ are iid with mean $\boldsymbol{\mu}_u = p_k(1, \mu_g, \boldsymbol{\mu}_k^T)^T$ and covariance

$$\boldsymbol{\Sigma}_u = \frac{1 - p_k}{p_k} \boldsymbol{\mu}_u \boldsymbol{\mu}_u^T + p_k \begin{pmatrix} 0 & 0 & 0 \\ 0 & \mu_g(1 - \mu_g) & \boldsymbol{\Sigma}_{xg}^T \\ 0 & \boldsymbol{\Sigma}_{xg} & \boldsymbol{\Sigma}_k \end{pmatrix}$$

where $p_k = \int_{B_{k_N}} F(d\mathbf{x})$, $\mu_g = p_k^{-1} \int_{B_{k_N}} G(\mathbf{x}) F(d\mathbf{x})$, $\boldsymbol{\mu}_k = p_k^{-1} \int_{B_{k_N}} \mathbf{x} F(d\mathbf{x})$, $\boldsymbol{\Sigma}_{xg} = p_k^{-1} \int_{B_{k_N}} \mathbf{x} G(\mathbf{x}) F(d\mathbf{x}) - \mu_g \boldsymbol{\mu}_k \in \mathbb{R}^p$, and $\boldsymbol{\Sigma}_k = p_k^{-1} \int_{B_{k_N}} \mathbf{x}\mathbf{x}^T F(d\mathbf{x}) - \boldsymbol{\mu}_k \boldsymbol{\mu}_k^T$.

4.2.1 Asymptotic distribution of $\bar{Y}_k - G(\bar{X}_k)$

In this section, we consider the cases when F has a density f. According to Lemma 4.1.3, $\boldsymbol{\Sigma}_u$ is positive definite if and only if $0 < p_k < 1$ and $0 < \mu_g < 1$.

We assume $0 < p_k < 1$, $0 < \mu_g < 1$ and $f \in C^2$, that is, the first two derivatives of f exist and are continuous. We denote $\nabla f(\mathbf{x}) \in \mathbb{R}^p$ be the gradient of f at $\mathbf{x} \in \mathbb{R}^p$. Then in a neighborhood of $\mathbf{x}_0$ we have a multivariate Taylor series expansion

$$f(\mathbf{x}) = f(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \nabla f(\mathbf{x}_0) + O(\|\mathbf{x} - \mathbf{x}_0\|^2)$$

For the simplified block type $B_{k_N} = \prod_{j=1}^{p}[x_{0j}, x_{0j} + h_N)$ with $h_N = c_h \cdot N^{-1/(pr)}$ for some $c_h > 0$ and $r > 1$, if $f(\mathbf{x}_0) > 0$, it can be verified that

$$\begin{aligned}
p_k &= c_h^p \cdot f(\mathbf{x}_0) \cdot N^{-\frac{1}{r}} + \frac{1}{2} c_h^{p+1} \cdot \mathbf{1}^T \nabla f(\mathbf{x}_0) \cdot N^{-\frac{1}{r}(1+\frac{1}{p})} + O(N^{-\frac{1}{r}(1+\frac{2}{p})}) \\
p_k(1-p_k) &= c_h^p \cdot f(\mathbf{x}_0) \cdot N^{-\frac{1}{r}} + \left[\frac{1}{2} c_h^{p+1} \cdot \mathbf{1}^T \nabla f(\mathbf{x}_0) - c_h^{2d} \cdot f(\mathbf{x}_0)^2 \cdot \mathbb{1}_{p=1}\right] N^{-\frac{1}{r}(1+\frac{1}{p})} \\
&\quad + O(N^{-\frac{1}{r}(1+\frac{2}{p})}) \\
\mu_g &= G(\mathbf{x}_0) + \frac{1}{2} c_h \cdot \mathbf{1}^T \nabla G(\mathbf{x}_0) \cdot N^{-\frac{1}{rp}} + O(N^{-\frac{2}{rp}}) \\
\mu_k &= \mathbf{x}_0 + \frac{1}{2} c_h \cdot \mathbf{1} \cdot N^{-\frac{1}{rp}} + O(N^{-\frac{2}{rp}}) \\
\mu_g(1-\mu_g) &= G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)] + \frac{1}{2} c_h [1 - 2G(\mathbf{x}_0)] \cdot \mathbf{1}^T \nabla G(\mathbf{x}_0) \cdot N^{-\frac{1}{rp}} + O(N^{-\frac{2}{rp}}) \\
\boldsymbol{\Sigma}_{xg} &= O(N^{-\frac{2}{rp}}) \\
\boldsymbol{\Sigma}_k &= O(N^{-\frac{2}{rp}})
\end{aligned}$$

as $N \to \infty$. Therefore,

$$\begin{aligned}
\boldsymbol{\mu}_u &= c_h^p f(\mathbf{x}_0) \cdot \boldsymbol{\mu}_{u1} \cdot N^{-\frac{1}{r}} + \frac{1}{2} c_h^{p+1} \cdot \boldsymbol{\mu}_{u2} \cdot N^{-\frac{1}{r}(1+\frac{1}{p})} + O(N^{-\frac{1}{r}(1+\frac{2}{p})}) \\
\boldsymbol{\Sigma}_u &= c_h^p f(\mathbf{x}_0) \cdot \boldsymbol{\Sigma}_{u1} \cdot N^{-\frac{1}{r}} + \frac{1}{2} c_h^{p+1} \cdot \boldsymbol{\Sigma}_{u2} \cdot N^{-\frac{1}{r}(1+\frac{1}{p})} + O(N^{-\frac{1}{r}(1+\frac{2}{p})})
\end{aligned}$$

where

$$\boldsymbol{\mu}_{u1} = \begin{pmatrix} 1 \\ G(\mathbf{x}_0) \\ \mathbf{x}_0 \end{pmatrix}, \quad \boldsymbol{\mu}_{u2} = \begin{pmatrix} \mathbf{1}^T \nabla f(\mathbf{x}_0) \\ \mathbf{1}^T [f(\mathbf{x}_0) \nabla G(\mathbf{x}_0) + G(\mathbf{x}_0) \nabla f(\mathbf{x}_0)] \\ f(\mathbf{x}_0)\mathbf{1} + \mathbf{1}^T \nabla f(\mathbf{x}_0) \mathbf{x}_0 \end{pmatrix}$$

$$\boldsymbol{\Sigma}_{u1} = \begin{pmatrix} 1 & G(\mathbf{x}_0) & \mathbf{x}_0^T \\ G(\mathbf{x}_0) & G(\mathbf{x}_0) & G(\mathbf{x}_0)\mathbf{x}_0^T \\ \mathbf{x}_0 & G(\mathbf{x}_0)\mathbf{x}_0 & \mathbf{x}_0\mathbf{x}_0^T \end{pmatrix}, \quad \boldsymbol{\Sigma}_{u2} = \begin{pmatrix} \sigma_{11}^{(u2)} & \sigma_{21}^{(u2)} & (\sigma_{31}^{(u2)})^T \\ \sigma_{21}^{(u2)} & \sigma_{22}^{(u2)} & (\sigma_{32}^{(u2)})^T \\ \sigma_{31}^{(u2)} & \sigma_{32}^{(u2)} & \sigma_{33}^{(u2)} \end{pmatrix}$$

$$\begin{aligned}
\sigma_{11}^{(u2)} &= \mathbf{1}^T \nabla f(\mathbf{x}_0) - 2c_h^{p-1} f(\mathbf{x}_0)^2 \cdot \mathbb{1}_{p=1} \\
\sigma_{21}^{(u2)} &= \mathbf{1}^T [f(\mathbf{x}_0)\nabla G(\mathbf{x}_0) + G(\mathbf{x}_0)\nabla f(\mathbf{x}_0)] - 2c_h^{p-1} f(\mathbf{x}_0)^2 G(\mathbf{x}_0) \cdot \mathbb{1}_{p=1} \\
\sigma_{31}^{(u2)} &= f(\mathbf{x}_0)\mathbf{1} + \mathbf{1}^T \nabla f(\mathbf{x}_0)\mathbf{x}_0 - 2c_h^{p-1} f(\mathbf{x}_0)^2 \mathbf{x}_0 \cdot \mathbb{1}_{p=1} \\
\sigma_{22}^{(u2)} &= \mathbf{1}^T [f(\mathbf{x}_0)\nabla G(\mathbf{x}_0) + G(\mathbf{x}_0)\nabla f(\mathbf{x}_0)] - 2c_h^{p-1} f(\mathbf{x}_0)^2 G(\mathbf{x}_0)^2 \cdot \mathbb{1}_{p=1} \\
\sigma_{32}^{(u2)} &= \mathbf{1}^T [f(\mathbf{x}_0)\nabla G(\mathbf{x}_0) + G(\mathbf{x}_0)\nabla f(\mathbf{x}_0)]\mathbf{x}_0 + f(\mathbf{x}_0)G(\mathbf{x}_0)\mathbf{1} \\
&\quad -2c_h^{p-1} f(\mathbf{x}_0)^2 G(\mathbf{x}_0)\mathbf{x}_0 \cdot \mathbb{1}_{p=1} \\
\sigma_{33}^{(u2)} &= \mathbf{1}^T \nabla f(\mathbf{x}_0)\mathbf{x}_0\mathbf{x}_0^T + f(\mathbf{x}_0)(\mathbf{x}_0\mathbf{1}^T + \mathbf{1}\mathbf{x}_0^T) - 2c_h^{p-1} f(\mathbf{x}_0)^2 \mathbf{x}_0\mathbf{x}_0^T \cdot \mathbb{1}_{p=1}
\end{aligned}$$

Fixing N, recall that $\mathbf{U}_1, \ldots, \mathbf{U}_N$ are iid with mean $\boldsymbol{\mu}_u \in \mathbb{R}^{p+2}$ and covariance $\boldsymbol{\Sigma}_u \in \mathbb{R}^{(p+2)\times(p+2)}$. When $\boldsymbol{\Sigma}_u$ is positive definite, we denote the map $M : \mathbb{R}^{p+2} \to \mathbb{R}$ such that $M((z, v, \mathbf{u}^T)^T) = v/z - G(\mathbf{u}/z)$ with $z, v \in \mathbb{R}$ and $\mathbf{u} \in \mathbb{R}^p$. Then $M(\bar{\mathbf{U}}) = \bar{Y}_k - G(\bar{\mathbf{X}}_k)$ and

$$\begin{aligned}
M(\boldsymbol{\mu}_u) &= \mu_g - G(\boldsymbol{\mu}_k) = O(N^{-2/(rp)}) \\
\nabla M(\boldsymbol{\mu}_u) &= p_k^{-1}[\boldsymbol{\mu}_k^T \nabla G(\boldsymbol{\mu}_k) - \mu_g,\ 1,\ -\nabla G(\boldsymbol{\mu}_k)^T]^T \\
\sigma^2_{ygx} &= \nabla M(\boldsymbol{\mu}_u)^T \cdot \boldsymbol{\Sigma}_u \cdot \nabla M(\boldsymbol{\mu}_u) \\
&= p_k^{-1}\left(\mu_g(1-\mu_g) - 2\nabla G(\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_{xg} + \nabla G(\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k \nabla G(\boldsymbol{\mu}_k)\right) \\
&= \frac{G(\mathbf{x}_0)[1-G(\mathbf{x}_0)]}{c_h^p f(\mathbf{x}_0)} \cdot N^{\frac{1}{r}} \\
&\quad + \frac{f(\mathbf{x}_0)[1-2G(\mathbf{x}_0)]\mathbf{1}^T \nabla G(\mathbf{x}_0) - G(\mathbf{x}_0)[1-G(\mathbf{x}_0)]\mathbf{1}^T \nabla f(\mathbf{x}_0)}{2c_h^{p-1} f(\mathbf{x}_0)^2} \cdot N^{\frac{1}{r}(1-\frac{1}{p})} \\
&\quad + O(N^{\frac{1}{r}(1-\frac{2}{p})})
\end{aligned}$$

If we further denote $\nabla^2 f(\mathbf{x}) \in \mathbb{R}^{p\times d}$ be the Hessian matrix of f at $\mathbf{x} \in \mathbb{R}^p$ and assume that $f \in C^3$, then in a neighborhood of $\mathbf{x}_0$

$$f(\mathbf{x}) = f(\mathbf{x}_0) + (\mathbf{x}-\mathbf{x}_0)^T \nabla f(\mathbf{x}_0) + \frac{1}{2}(\mathbf{x}-\mathbf{x}_0)^T \cdot \nabla^2 f(\mathbf{x}_0) \cdot (\mathbf{x}-\mathbf{x}_0) + O(\|\mathbf{x}-\mathbf{x}_0\|^3)$$

If $f(\mathbf{x}_0) > 0$, it can be verified that

$$\begin{aligned}
p_k &= c_h^p \cdot f(\mathbf{x}_0) \cdot N^{-\frac{1}{r}} + \frac{1}{2} c_h^{p+1} \cdot \mathbf{1}^T \nabla f(\mathbf{x}_0) \cdot N^{-\frac{1}{r}(1+\frac{1}{p})} \\
&\quad + \frac{1}{8} c_h^{p+2} [\mathbf{1}^T \nabla^2 f(\mathbf{x}_0) \mathbf{1} + \frac{1}{3} tr(\nabla^2 f(\mathbf{x}_0))] \cdot N^{-\frac{1}{r}(1+\frac{2}{p})} + O(N^{-\frac{1}{r}(1+\frac{3}{p})}) \\
\mu_g &= G(\mathbf{x}_0) + \frac{1}{2} c_h \cdot \mathbf{1}^T \nabla G(\mathbf{x}_0) \cdot N^{-\frac{1}{rp}} \\
&\quad + \frac{1}{24} c_h^2 \left[3 \cdot \mathbf{1}^T \nabla^2 G(\mathbf{x}_0) \mathbf{1} + tr(\nabla^2 G(\mathbf{x}_0)) + \frac{2}{f(\mathbf{x}_0)} \nabla f(\mathbf{x}_0)^T \nabla G(\mathbf{x}_0) \right] \cdot N^{-\frac{2}{rp}} \\
&\quad + O(N^{-\frac{3}{rp}}) \\
\mu_k &= \mathbf{x}_0 + \frac{1}{2} c_h \cdot \mathbf{1} \cdot N^{-\frac{1}{rp}} + \frac{1}{12} c_h^2 \frac{1}{f(\mathbf{x}_0)} \nabla f(\mathbf{x}_0) \cdot N^{-\frac{2}{rp}} + O(N^{-\frac{3}{rp}}) \\
M(\mu_u) &= \mu_g - G(\mu_k) = \frac{1}{24} c_h^2 \cdot tr(\nabla^2 G(\mathbf{x}_0)) \cdot N^{-\frac{2}{rp}} + O(N^{-\frac{3}{rp}}) \\
\mu_u &= c_h^p f(\mathbf{x}_0) \cdot \mu_{u1} \cdot N^{-\frac{1}{r}} + \frac{1}{2} c_h^{p+1} \cdot \mu_{u2} \cdot N^{-\frac{1}{r}(1+\frac{1}{p})} \\
&\quad + \frac{1}{24} c_h^{p+2} \cdot \mu_{u3} \cdot N^{-\frac{1}{r}(1+\frac{2}{p})} + O(N^{-\frac{1}{r}(1+\frac{3}{p})})
\end{aligned}$$

as $N \to \infty$, where

$$\begin{aligned}
\mu_{u3} &= \begin{pmatrix} 3 \cdot \mathbf{1}^T \nabla^2 f(\mathbf{x}_0) \mathbf{1} + tr(\nabla^2 f(\mathbf{x}_0)) \\ \mu_{u3b} \\ tr(\nabla^2 f(\mathbf{x}_0)) \mathbf{x}_0 + 2\nabla f(\mathbf{x}_0) + 3 \cdot \mathbf{1}^T \nabla^2 f(\mathbf{x}_0) \mathbf{1} \cdot \mathbf{x}_0 + 6 \cdot \mathbf{1}^T \nabla f(\mathbf{x}_0) \mathbf{1} \end{pmatrix} \\
\mu_{u3b} &= f(\mathbf{x}_0) tr(\nabla^2 G(\mathbf{x}_0)) + G(\mathbf{x}_0) tr(\nabla^2 f(\mathbf{x}_0)) + 2\nabla f(\mathbf{x}_0)^T \nabla G(\mathbf{x}_0) \\
&\quad + 3 f(\mathbf{x}_0) \mathbf{1}^T \nabla^2 G(\mathbf{x}_0) \mathbf{1} + 3 G(\mathbf{x}_0) \mathbf{1}^T \nabla^2 f(\mathbf{x}_0) \mathbf{1} + 6 \cdot \mathbf{1}^T \nabla f(\mathbf{x}_0) \cdot \mathbf{1}^T \nabla G(\mathbf{x}_0)
\end{aligned}$$

Theorem 4.2.1. *Let* $\mathbf{U}_i = Z_i(1, Y_i, \mathbf{X}_i^T)^T$, $i = 1, \ldots, N$, *where* $Z_i = \mathbf{1}_{\mathbf{X}_i \in B_{k_N}}$. *Suppose* $\mathbf{X}_1, \ldots, \mathbf{X}_N$ *are iid with density* $f \in C^3$, $f(\mathbf{x}_0) > 0$, *and* $G \in C^3$. *If* $r < 1 + 6/p$, *then*

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_N) \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0},\, c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

where $\bar{\mathbf{U}} = N^{-1}\sum_{i=1}^N \mathbf{U}_i$ *and* $\boldsymbol{\mu}_N = c_h^p f(\mathbf{x}_0)\boldsymbol{\mu}_{u1} \cdot N^{-\frac{1}{r}} + \frac{1}{2}c_h^{p+1}\boldsymbol{\mu}_{u2} \cdot N^{-\frac{1}{r}(1+\frac{1}{p})} + \frac{1}{24}c_h^{p+2}\boldsymbol{\mu}_{u3} \cdot N^{-\frac{1}{r}(1+\frac{2}{p})}$.

It should be noted that $\boldsymbol{\Sigma}_{u1}$ is degenerated with rank 2 if $G(\mathbf{x}_0) \in (0, 1)$ or rank 1 if $G(\mathbf{x}_0) \in \{0, 1\}$ (see Lemma 4.2.1).

Proof of Theorem 4.2.1: Let $\mathbf{Z}_N = N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_N) = N^{-\frac{1}{2}(1-\frac{1}{r})}\sum_{i=1}^N(\mathbf{U}_i - \boldsymbol{\mu}_N)$. Let $\varphi_{\mathbf{Z}_N}$ and φ_N be the characteristic functions of $\mathbf{Z}_N$ and $\mathbf{U}_i - \boldsymbol{\mu}_N$, respectively. Then

$$\varphi_{\mathbf{Z}_N}(\mathbf{t}) = \varphi_{\sum_{i=1}^N(\mathbf{U}_i - \boldsymbol{\mu}_N)}(\mathbf{t} \cdot N^{-\frac{1}{2}(1-\frac{1}{r})}) = \varphi_N(\mathbf{t} \cdot N^{-\frac{1}{2}(1-\frac{1}{r})})^N$$

On the other hand, for $\mathbf{t} = (t_1, t_2, \mathbf{t}_3^T)^T \in \mathbb{R}^{p+2}$,

$$\begin{aligned}
\varphi_N(\mathbf{t}) &= E e^{i\mathbf{t}^T(\mathbf{U}_i - \boldsymbol{\mu}_N)} \\
&= e^{-i\mathbf{t}^T\boldsymbol{\mu}_N} \cdot E e^{i\mathbf{t}^T\mathbf{U}_i} \\
&= e^{-i\mathbf{t}^T\boldsymbol{\mu}_N} \cdot E\left(E\left(e^{i\mathbf{t}^T\mathbf{U}_i} \mid \mathbf{X}_i\right)\right) \\
&= e^{-i\mathbf{t}^T\boldsymbol{\mu}_N} \cdot E\left(e^{iZ_i(t_1 + \mathbf{t}_3^T\mathbf{X}_i)}E\left(e^{it_2Z_iY_i} \mid \mathbf{X}_i\right)\right) \\
&= e^{-i\mathbf{t}^T\boldsymbol{\mu}_N} \cdot E\left(e^{iZ_i(t_1 + \mathbf{t}_3^T\mathbf{X}_i)}\left[1 - G(\mathbf{X}_i) + G(\mathbf{X}_i)e^{it_2Z_i}\right]\right) \\
&= e^{-i\mathbf{t}^T\boldsymbol{\mu}_N}\left[1 - p_k + e^{it_1}\int_{B_{k_N}} e^{i\mathbf{t}_3^T\mathbf{x}}\left[1 - G(\mathbf{x}) + G(\mathbf{x})e^{it_2}\right] f(\mathbf{x})d\mathbf{x}\right]
\end{aligned}$$

Since $f \in C^3$ and $G \in C^3$, with the Taylor expansions of f and G in a neighborhood of $\mathbf{x}_0$, we can verify that

$$\begin{aligned}
& \varphi_N(\mathbf{t} \cdot N^{-\frac{1}{2}(1-\frac{1}{r})}) \\
= & \left(1 - it^T \mu_N \cdot N^{-\frac{1}{2}(1-\frac{1}{r})} + O(N^{-1-\frac{1}{r}})\right) \cdot \left(1 + it^T \mu_N \cdot N^{-\frac{1}{2}(1-\frac{1}{r})}\right. \\
& - \left.\frac{1}{2} c_h^p f(\mathbf{x}_0) \mathbf{t}^T \Sigma_{u1} \mathbf{t} \cdot N^{-1} + O(N^{-\frac{1}{2}(1+\frac{1}{r})-\frac{3}{rp}}) + O(N^{-1-\frac{1}{rp}}) + O(N^{-1-\frac{1}{2}(1-\frac{1}{r})})\right) \\
= & 1 - \frac{1}{2} c_h^p f(\mathbf{x}_0) \cdot \mathbf{t}^T \Sigma_{u1} \mathbf{t} \cdot N^{-1} + O(N^{-\frac{1}{2}(1+\frac{1}{r})-\frac{3}{rp}}) + O(N^{-1-\frac{1}{rp}}) + O(N^{-1-\frac{1}{2}(1-\frac{1}{r})})
\end{aligned}$$

as N goes to infinity. If $r < 1 + 6/p$, then $-\frac{1}{2}(1 + \frac{1}{r}) - \frac{3}{rp} < -1$, which implies

$$\lim_{N\to\infty} \varphi_{\mathbf{Z}_N}(\mathbf{t}) = \lim_{N\to\infty} \varphi_N(\mathbf{t} \cdot N^{-\frac{1}{2}(1-\frac{1}{r})})^N = \exp\left\{-\frac{1}{2}\mathbf{t}^T \cdot c_h^p f(\mathbf{x}_0) \Sigma_{u1} \cdot \mathbf{t}\right\}$$

That is, $\mathbf{Z}_N \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0}, c_h^p f(\mathbf{x}_0)\Sigma_{u1})$, as N goes to infinity. □

The condition $r < 1 + 6/p$ in Theorem 4.2.1 can be further extended. Actually, if, for example, $r < 1 + 8/p$, from the proof of Theorem 4.2.1, the Taylor expansion of $1 - p_k + e^{it_1} \int_{B_{k_N}} e^{it_3^T \mathbf{x}} \left[1 - G(\mathbf{x}) + G(\mathbf{x}) e^{it_2}\right] f(\mathbf{x}) d\mathbf{x}$ can be extended to items at order $N^{-\frac{1}{2}(1+\frac{1}{r})-\frac{3}{rp}}$ with the leftover $O(N^{-\frac{1}{2}(1+\frac{1}{r})-\frac{4}{rp}})$. In this case, by updating μ_N with an additional item at order $N^{-\frac{1}{r}(1+\frac{3}{p})}$, the same asymptotic normal distribution still holds.

Lemma 4.2.1. *The rank of* Σ_{u1} *is* 2 *if* $G(\mathbf{x}_0) \in (0, 1)$ *or* 1 *if* $G(\mathbf{x}_0) \in \{0, 1\}$. *More specifically,*

(1) The eigenvalues of $\mathbf{\Sigma}_{u1}$ *are*

$$\lambda_1, \lambda_2 = \frac{1}{2}\left\{1 + \mathbf{x}_0^T\mathbf{x}_0 + G(\mathbf{x}_0) \pm \left[\left(1 + \mathbf{x}_0^T\mathbf{x}_0 - G(\mathbf{x}_0)\right)^2 + 4(1 + \mathbf{x}_0^T\mathbf{x}_0)G(\mathbf{x}_0)^2\right]^{1/2}\right\}$$

$\lambda_3 = \cdots = \lambda_{p+2} = 0$, *where* $\lambda_1 > \lambda_2 \geq 0$, *and* $\lambda_2 = 0$ *if and only if* $G(\mathbf{x}_0) \in \{0, 1\}$.

(2) If $\lambda \neq 0$, *an eigenvector corresponding to it is*

$$(1, \frac{\lambda G(\mathbf{x}_0)}{\lambda - G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}, \mathbf{x}_0^T)^T$$

(3) The collection of eigenvectors of 0, *or the null space of* $\mathbf{\Sigma}_{u1}$, *is* $\{(-\mathbf{x}_0^T\mathbf{u},\ 0,\ \mathbf{u}^T)^T \in \mathbb{R}^{p+2} \mid \mathbf{u} \in \mathbb{R}^p\}$ *if* $G(\mathbf{x}_0) \in (0, 1)$*; or* $\{(-G(\mathbf{x}_0)\nu - \mathbf{x}_0^T\mathbf{u},\ \nu,\ \mathbf{u}^T)^T \in \mathbb{R}^{p+2} \mid \mathbf{u} \in \mathbb{R}^p, \nu \in \mathbb{R}\}$ *if* $G(\mathbf{x}_0) \in \{0, 1\}$.

Proof of Lemma 4.2.1: In order to find the eigenvalues of $\mathbf{\Sigma}_{u1}$, we consider the determinant $f(\lambda) = |\lambda\mathbf{I}_{p+2} - \mathbf{\Sigma}_{u1}|$. By row operations, $f(\lambda)$ is equal to the determinant of

$$\begin{pmatrix} \lambda - 1 & -G(\mathbf{x}_0) & -\mathbf{x}_0^T \\ -\lambda G(\mathbf{x}_0) & \lambda - G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)] & \mathbf{0}^T \\ -\lambda\mathbf{x}_0 & 0 & \lambda\mathbf{I}_p \end{pmatrix} \triangleq \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}$$

where $\mathbf{A} \in \mathbb{R}^{2\times 2}$, $\mathbf{B} \in \mathbb{R}^{2\times d}$, $\mathbf{C} \in \mathbb{R}^{p\times 2}$, and $\mathbf{D} \in \mathbb{R}^{p\times d}$. If $\lambda \neq 0$, by Schur's determinant identity

$$\begin{aligned} f(\lambda) &= |\mathbf{D}| \cdot |\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}| \\ &= \lambda^p \cdot \left\{\lambda^2 - [1 + \mathbf{x}_0^T\mathbf{x}_0 + G(\mathbf{x}_0)]\lambda + (1 + \mathbf{x}_0^T\mathbf{x}_0)G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]\right\} \end{aligned} \tag{4.3}$$

Apparently, (4.3) is true as well if $\lambda = 0$. Then the eigenvalues listed in (1) can be obtained by solving $f(\lambda) = 0$. The eigenvectors listed in (2) and (3) can be verified as well. □

Corollary 4.2.1. *Under the conditions of Theorem 4.2.1, if* $r \in (1, 1 + 2/p)$,

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_{N1}) \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0},\ c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

where $\boldsymbol{\mu}_{N1} = c_h^p f(\mathbf{x}_0)\boldsymbol{\mu}_{u1} \cdot N^{-\frac{1}{r}}$. *If* $r = 1 + 2/p$, *then*

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_{N1}) \xrightarrow{\mathcal{D}} N_{p+2}(\frac{1}{2}c_h^{p+1}\boldsymbol{\mu}_{u2},\ c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

If $r \in (1 + 2/p, 1 + 4/p)$, *then*

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_{N2}) \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0},\ c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

where $\boldsymbol{\mu}_{N2} = c_h^p f(\mathbf{x}_0)\boldsymbol{\mu}_{u1} \cdot N^{-\frac{1}{r}} + \frac{1}{2}c_h^{p+1}\boldsymbol{\mu}_{u2} \cdot N^{-\frac{1}{r}(1+\frac{1}{p})}$. *If* $r = 1 + 4/p$, *then*

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_{N2}) \xrightarrow{\mathcal{D}} N_{p+2}(\frac{1}{24}c_h^{p+2}\boldsymbol{\mu}_{u3},\ c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

Proof of Corollary 4.2.1: From Theorem 4.2.1, we have

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_N) \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0},\ c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

Note that

$$\begin{aligned} N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - \boldsymbol{\mu}_N) &= N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - c_h^p f(\mathbf{x}_0)\boldsymbol{\mu}_{u1} \cdot N^{-\frac{1}{r}}) \\ &\quad - \frac{1}{2}c_h^{p+1}\boldsymbol{\mu}_{u2} \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{1}{rp}} - \frac{1}{24}c_h^{p+2}\boldsymbol{\mu}_{u3} \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{2}{rp}} \end{aligned}$$

If $r < 1 + 2/p$, then $\frac{1}{2}(1 - \frac{1}{r}) - \frac{1}{rp} < 0$ and thus $\frac{1}{2}(1 - \frac{1}{r}) - \frac{2}{rp} < 0$. By Slutsky's Theorem (see, for example, Theorem 1.5 in (DasGupta, 2008)), we get

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - c_h^p f(\mathbf{x}_0)\boldsymbol{\mu}_{u1} \cdot N^{-\frac{1}{r}}) \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0},\ c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

If $r < 1 + 4/p$, then $\frac{1}{2}(1 - \frac{1}{r}) - \frac{2}{rp} < 0$. By Slutsky's Theorem, we get

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{\mathbf{U}} - c_h^p f(\mathbf{x}_0)\boldsymbol{\mu}_{u1} \cdot N^{-\frac{1}{r}} - \frac{1}{2}c_h^{p+1}\boldsymbol{\mu}_{u2} \cdot N^{-\frac{1}{r}(1+\frac{1}{p})}) \xrightarrow{\mathcal{D}} N_{p+2}(\mathbf{0},\ c_h^p f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1})$$

Actually, if $r = 1 + 2/p$, then $\frac{1}{2}(1 - \frac{1}{r}) - \frac{1}{rp} = 0$ and $\frac{1}{2}(1 - \frac{1}{r}) - \frac{2}{rp} < 0$. By Slutsky's Theorem, we get

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{U} - c_h^p f(x_0)\mu_{u1} \cdot N^{-\frac{1}{r}}) \xrightarrow{\mathcal{D}} N_{p+2}(\frac{1}{2}c_h^{p+1}\mu_{u2},\ c_h^p f(x_0)\Sigma_{u1})$$

If $r = 1 + 4/p$, then $\frac{1}{2}(1 - \frac{1}{r}) - \frac{2}{rp} = 0$. By Slutsky's Theorem, we get

$$N^{\frac{1}{2}(1+\frac{1}{r})}(\bar{U} - c_h^p f(x_0)\mu_{u1} \cdot N^{-\frac{1}{r}} - \frac{1}{2}c_h^{p+1}\mu_{u2} \cdot N^{-\frac{1}{r}(1+\frac{1}{p})}) \xrightarrow{\mathcal{D}} N_{p+2}(\frac{1}{24}c_h^{p+2}\mu_{u3},\ c_h^p f(x_0)\Sigma_{u1})$$

□

Theorem 4.2.2. *Suppose* $X_1, \ldots, X_N$ *are iid with density* $f \in C^3$, $f(x_0) > 0$, *and* $G \in C^3$. *If* $1 < r < 1 + 6/p$, *then*

$$N^{\frac{1}{2}(1-\frac{1}{r})}\left[\bar{Y}_k - G(\bar{X}_k) - \frac{1}{24}c_h^2 \cdot \mathrm{tr}(\nabla^2 G(x_0)) \cdot N^{-\frac{2}{rp}}\right] \xrightarrow{\mathcal{D}} N\left(0,\ \frac{G(x_0)[1 - G(x_0)]}{c_h^d f(x_0)}\right) \tag{4.4}$$

Proof of Theorem 4.2.2: Under the conditions of Theorem 4.2.1, let $\bar{U}' = N^{1/r}\bar{U}$, $\mu'_{N1} = c_h^d f(x_0)\mu_{u1}$, and $\Delta_N = \frac{1}{2}c_h^{d+1}\mu_{u2} \cdot N^{-\frac{1}{rp}} + \frac{1}{24}c_h^{d+2}\mu_{u3} \cdot N^{-\frac{2}{rp}}$. Then $N^{1/r}\mu_N = (\mu'_N + \Delta_N)$ and for $r < 1 + 6/p$,

$$N^{\frac{1}{2}(1-\frac{1}{r})}(\bar{U}' - \mu'_N - \Delta_N) \xrightarrow{\mathcal{D}} N_{d+2}(\mathbf{0},\ c_h^d f(x_0)\Sigma_{u1}) \tag{4.5}$$

Recall that $M(\mathbf{w}) = \nu/z - G(\mathbf{u}/z)$ for $\mathbf{w} = (z, \nu, \mathbf{u}^T)^T \in \mathbb{R}^{d+2}$, $z > 0$. Let $\delta = \frac{1}{2}c_h^d f(\mathbf{x}_0) > 0$. According to the mean-value theorem (see, for example, Chapter 4 of (Ferguson, 1996)), for $|\mathbf{w} - \boldsymbol{\mu}'_{N1}| < \delta$, we have $z > \frac{1}{2}c_h^d f(\mathbf{x}_0) > 0$ and

$$M(\mathbf{w}) = M(\boldsymbol{\mu}'_{N1}) + \int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\mathbf{w} - \boldsymbol{\mu}'_{N1}))^T ds \cdot (\mathbf{w} - \boldsymbol{\mu}'_{N1})$$

$$\nabla M(\boldsymbol{\mu}'_{N1} + s(\mathbf{w} - \boldsymbol{\mu}'_{N1})) = \nabla M(\boldsymbol{\mu}'_{N1}) + \int_0^1 \nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\mathbf{w} - \boldsymbol{\mu}'_{N1}))du \cdot s(\mathbf{w} - \boldsymbol{\mu}'_{N1})$$

Then for $|\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}| < \delta$,

$$\begin{aligned}
M(\bar{\mathbf{U}}') &= M(\boldsymbol{\mu}'_{N1}) + \int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))^T ds \cdot (\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}) \\
&= M(\boldsymbol{\mu}'_{N1}) + \int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))^T ds \cdot (\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1} - \boldsymbol{\Delta}_N) \\
&\quad + \int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))^T ds \cdot \boldsymbol{\Delta}_N \\
&= M(\boldsymbol{\mu}'_{N1}) + \int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))^T ds \cdot (\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1} - \boldsymbol{\Delta}_N) \\
&\quad + \boldsymbol{\Delta}_N^T \int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))^T du ds \cdot (\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1} - \boldsymbol{\Delta}_N) \\
&\quad + \nabla M(\boldsymbol{\mu}'_{N1})^T \boldsymbol{\Delta}_N + \boldsymbol{\Delta}_N^T \int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))^T du ds \cdot \boldsymbol{\Delta}_N
\end{aligned}$$

Since $M(\bar{\mathbf{U}}') = M(\bar{\mathbf{U}}) = \bar{Y}_k - G(\bar{\mathbf{X}}_k)$, $M(\boldsymbol{\mu}'_{N1}) = 0$, $\nabla M(\boldsymbol{\mu}'_{N1}) = [c_h^d f(\mathbf{x}_0)]^{-1}(-G(\mathbf{x}_0) + \mathbf{x}_0^T \nabla G(\mathbf{x}_0),\ 1,\ -\nabla G(\mathbf{x}_0)^T)^T$ and $\nabla M(\boldsymbol{\mu}'_{N1})^T \boldsymbol{\Delta}_N = \frac{1}{24}c_h^2[tr(\nabla^2 G(\mathbf{x}_0)) + 3 \cdot \mathbf{1}^T \nabla^2 G(\mathbf{x}_0)\mathbf{1}] \cdot N^{-\frac{2}{rp}}$, it can be verified that

$$N^{\frac{1}{2}(1-\frac{1}{r})}\left[\bar{Y}_k - G(\bar{\mathbf{X}}_k) - \frac{1}{24}c_h^2 \cdot tr(\nabla^2 G(\mathbf{x}_0)) \cdot N^{-\frac{2}{rp}}\right] = A_1 + A_2 + A_3 + A_4 + A_5 \qquad (4.6)$$

where

$$
\begin{aligned}
A_1 &= \int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))^T ds \cdot N^{\frac{1}{2}(1-\frac{1}{r})}(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1} - \boldsymbol{\Delta}_N) \\
A_2 &= \boldsymbol{\Delta}_N^T \int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))duds \cdot N^{\frac{1}{2}(1-\frac{1}{r})}(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1} - \boldsymbol{\Delta}_N) \\
A_3 &= \left[\frac{1}{4}c_h^{2d+2}\boldsymbol{\mu}_{u2}^T \int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))duds \cdot \boldsymbol{\mu}_{u2} + \frac{1}{8}c_h^2 \mathbf{1}^T \nabla^2 G(\mathbf{x}_0)\mathbf{1}\right] \\
&\quad \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{2}{rp}} \\
A_4 &= \frac{1}{48}c_h^{2d+3}\boldsymbol{\mu}_{u2}^T \int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))duds \cdot \boldsymbol{\mu}_{u3} \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{3}{rp}} \\
A_5 &= \frac{1}{576}c_h^{2d+4}\boldsymbol{\mu}_{u3}^T \int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))duds \cdot \boldsymbol{\mu}_{u4} \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{4}{rp}}
\end{aligned}
$$

Due to (4.5), we have $\bar{\mathbf{U}}' - \boldsymbol{\mu}'_N - \boldsymbol{\Delta}_N \xrightarrow{\mathcal{D}} 0$ or equivalently $\bar{\mathbf{U}}' - \boldsymbol{\Delta}_N \xrightarrow{\mathcal{D}} \boldsymbol{\mu}'_N$. Since $\boldsymbol{\Delta}_N \xrightarrow{P} 0$, we obtain $\bar{\mathbf{U}}' \xrightarrow{\mathcal{D}} \boldsymbol{\mu}'_N$, which implies $P(|\bar{\mathbf{U}}' - \boldsymbol{\mu}'_N| < \delta) \to 1$. As a continuous function of $\bar{\mathbf{U}}'$, we have

$$
\int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))ds \xrightarrow{\mathcal{D}} \int_0^1 \nabla M(\boldsymbol{\mu}'_{N1} + s(\boldsymbol{\mu}'_{N1} - \boldsymbol{\mu}'_{N1}))ds = \nabla M(\boldsymbol{\mu}'_{N1})
$$

According to (4.5) and $\nabla M(\boldsymbol{\mu}'_{N1})^T \cdot c_h^d f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1} \cdot \nabla M(\boldsymbol{\mu}'_{N1}) = [c_h^d f(\mathbf{x}_0)]^{-1} G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]$,

$$
A_1 \xrightarrow{\mathcal{D}} \nabla M(\boldsymbol{\mu}'_{N1})^T \cdot N_{d+2}(\mathbf{0},\, c_h^d f(\mathbf{x}_0)\boldsymbol{\Sigma}_{u1}) \overset{\mathcal{D}}{=} N\left(\mathbf{0},\, \frac{G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}{c_h^d f(\mathbf{x}_0)}\right)
$$

Similarly, as a major component of A_2, A_3, A_4 and A_5,

$$
\int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))duds \xrightarrow{\mathcal{D}} \int_0^1 \int_0^1 s\nabla^2 M(\boldsymbol{\mu}'_{N1})duds = \frac{1}{2}\nabla^2 M(\boldsymbol{\mu}'_{N1})
$$

Since $\boldsymbol{\Delta}_N \xrightarrow{P} 0$, we have $A_2 \xrightarrow{P} 0$ as N goes to infinity. If $r < 1 + 6/p$, then $\frac{1}{2}(1 - \frac{1}{r}) < \frac{3}{rp}$ and $A_4 \xrightarrow{P} 0$, $A_5 \xrightarrow{P} 0$.

As for A_3, we need to apply the mean-value theorem to each entry of $\nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))$ and get

$$\begin{aligned} & \nabla^2 M(\boldsymbol{\mu}'_{N1} + us(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))_{ij} \\ = \; & \nabla^2 M(\boldsymbol{\mu}'_{N1})_{ij} + \sum_{l=1}^{d+2} us \int_0^1 \frac{\partial^3 M}{\partial w_i \partial w_j \partial w_l}(\boldsymbol{\mu}'_{N1} + tus(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))dt(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1})_l \end{aligned}$$

where $(\cdot)_{ij}$ denotes the (i,j)th entry of a matrix and $(\cdot)_l$ denotes the lth component of a vector. Since $\frac{1}{4}c_h^{2d+2}\boldsymbol{\mu}_{u2}^T \cdot \frac{1}{2}\nabla^2 M(\boldsymbol{\mu}'_{N1}) \cdot \boldsymbol{\mu}_{u2} = -\frac{1}{8}c_h^2 \mathbf{1}^T \nabla^2 G(\mathbf{x}_0)\mathbf{1}$, then $A_3 = \frac{1}{4}c_h^{2d+2}\mathbf{V}_N^T(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}) \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{2}{rp}} = A_6 + A_7$, where $\mathbf{V}_N = (V_{N1}, \ldots, V_{N,d+2})^T$,

$$\begin{aligned} V_{Nl} &= \sum_{i=1}^{d+2}\sum_{j=1}^{d+2}(\boldsymbol{\mu}_{u2})_i(\boldsymbol{\mu}_{u2})_j \int_0^1\int_0^1\int_0^1 us^2 \frac{\partial^3 M}{\partial w_i \partial w_j \partial w_l}(\boldsymbol{\mu}'_{N1} + tus(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1}))dtduds \\ A_6 &= \frac{1}{4}c_h^{2d+2}\mathbf{V}_N^T \cdot N^{\frac{1}{2}(1-\frac{1}{r})}(\bar{\mathbf{U}}' - \boldsymbol{\mu}'_{N1} - \boldsymbol{\Delta}_N) \cdot N^{-\frac{2}{rp}} \\ A_7 &= \frac{1}{4}c_h^{2d+2}\mathbf{V}_N^T\boldsymbol{\Delta}_N \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{2}{rp}} \end{aligned}$$

Since $\mathbf{V}_N$ is a continuous function of $\bar{\mathbf{U}}'$, then $\mathbf{V}_N \xrightarrow{\mathcal{D}} \mathbf{V} = (V_1, \ldots, V_{d+2})^T$ with

$$V_l = \frac{1}{6}\sum_{i=1}^{d+2}\sum_{j=1}^{d+2}(\boldsymbol{\mu}_{u2})_i(\boldsymbol{\mu}_{u2})_j \frac{\partial^3 M}{\partial w_i \partial w_j \partial w_l}(\boldsymbol{\mu}'_{N1})$$

As N goes to infinity, $A_6 \xrightarrow{P} 0$ due to $N^{-\frac{2}{rp}} \to 0$. If $r < 1 + 6/p$, then $\frac{1}{2}(1 - \frac{1}{r}) - \frac{2}{rp} < \frac{1}{rp}$ and $\boldsymbol{\Delta}_N \cdot N^{\frac{1}{2}(1-\frac{1}{r})-\frac{2}{rp}} \to 0$. Then $A_7 \xrightarrow{P} 0$ and thus $A_3 \xrightarrow{P} 0$. After all, the conclusion follows from (4.6). □

Similar as in Theorem 4.2.1, the condition $r < 1 + 6/p$ in Theorem 4.2.2 can be further extended. Actually, if, for example, $r < 1+8/p$, then $\boldsymbol{\mu}_N$ in Theorem 4.2.1 needs to be updated with an additional item at order $N^{-\frac{1}{r}(1+\frac{3}{d})}$, and $\boldsymbol{\Delta}_N$ in the proof of Theorem 4.2.2 needs an additional item at order $N^{-\frac{3}{rp}}$. Applying the mean value theorem to A_4 and A_7, we will obtain items at order $N^{-\frac{1}{2}(1-\frac{1}{r})-\frac{3}{d}}$, which will affect the left hand side of (4.4). Actually, by adding an item at $N^{-\frac{3}{rp}}$ inside the brackets of its left hand side, (4.4) still holds.

Similar as the proof for Corollary 4.2.1, we obtain the corollary below:

Under the same conditions of Theorem 4.2.2, if $1 < r < 1 + 4/p$, then

$$N^{\frac{1}{2}(1-\frac{1}{r})}\left[\bar{Y}_k - G(\bar{\mathbf{X}}_k)\right] \xrightarrow{\mathcal{D}} N\left(0,\ \frac{G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}{c_h^d f(\mathbf{x}_0)}\right)$$

If $r = 1 + 4/p$, then

$$N^{\frac{1}{2}(1-\frac{1}{r})}\left[\bar{Y}_k - G(\bar{\mathbf{X}}_k)\right] \xrightarrow{\mathcal{D}} N\left(\frac{1}{24}c_h^2 \cdot \mathrm{tr}(\nabla^2 G(\mathbf{x}_0)),\ \frac{G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}{c_h^d f(\mathbf{x}_0)}\right)$$

4.3 The choice of K_N

According to Theorem 4.2.2, when $1 < r < 1 + 6/p$, in terms of leading terms,

$$\begin{aligned} E[\bar{Y}_k - G(\bar{\mathbf{X}}_k)] &\sim \frac{1}{24} c_h^2 tr(\nabla^2 G(\mathbf{x}_0)) \cdot N^{-\frac{2}{rp}} \\ Var(\bar{Y}_k - G(\bar{\mathbf{X}}_k)) &\sim \frac{G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}{c_h^d f(\mathbf{x}_0)} \cdot N^{-(1-\frac{1}{r})} \end{aligned}$$

Therefore,

$$\begin{aligned} E([\bar{Y}_k - G(\bar{\mathbf{X}}_k)]^2) &= \left(E[\bar{Y}_k - G(\bar{\mathbf{X}}_k)]\right)^2 + Var(\bar{Y}_k - G(\bar{\mathbf{X}}_k)) \\ &\sim \frac{1}{576} c_h^4 [tr(\nabla^2 G(\mathbf{x}_0))]^2 \cdot N^{-\frac{4}{rp}} + \frac{G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}{c_h^d f(\mathbf{x}_0)} \cdot N^{-(1-\frac{1}{r})} \end{aligned}$$

Let $\delta(r) = \max\{-4/(rp), -(1 - 1/r)\}$ be the order of the leading term of $E([\bar{Y}_k - G(\bar{\mathbf{X}}_k)]^2)$. It can be verified that

$$\delta(r) = \begin{cases} -(1 - \frac{1}{r}) & \text{if } 1 < r < 1 + \frac{4}{p} \\ -\frac{4}{p+4} & \text{if } r = 1 + \frac{4}{p} \\ -\frac{4}{rp} & \text{if } r > 1 + \frac{4}{p} \end{cases}$$

Then $\delta(r)$ attains its minimum $-4/(p+4)$ at $r = 1+4/p$. Note that the optimal decreasing rate is the same as the decreasing rate of a MSE used in the kernel density estimation (see details in Prakasa Rao, 1983 P182). The identical decreasing rate of the MSEs verifies our result and also shows the connection between mean representative approach and kernel density estimation.

Recall that $h_N = c_h N^{-1/(rp)}$ leads to a block volume $v_k \sim N^{-1/r}$. If the blocks are roughly of the same size, then the total number of blocks $K \sim N^{1/r}$. In general, we assume that with finer

blocks (that is, K is large or r is small), the bias $E([\bar{Y}_k - G(\bar{\mathbf{X}}_k)])$ is negligible compared with the variance of $\bar{Y}_k - G(\bar{\mathbf{X}}_k)$, while when K is small or r is large, the mean square error is dominated by the bias $E([\bar{Y}_k - G(\bar{\mathbf{X}}_k)])$. The optimal rate for the number of blocks is $K \sim N^{d/(p+4)}$. Or equivalently, the optimal number of observations in each block is $n_k \sim N/K = N^{4/(p+4)}$.

For general regression problems, $K \sim N^{1/r}$ blocks or mean representatives may be used for estimating the parameters of interests, say, $\boldsymbol{\beta}$. For typical applications, $Var(\hat{\boldsymbol{\beta}}) \sim O(N^{\delta(r)}/K) = O(N^{\delta(r)-1/r})$. Note that $\delta(r) - 1/r = \max\{-(p+4)/(rp), -1\}$ attains its minimum -1 at $r \in (1,\ 1 + 4/p]$. Since smaller r indicates more blocks or more mean representatives, we recommend $r = 1 + 4/p$ again for minimizing the computational cost while keeping the same level of estimation accuracy.

When $r = 1 + 4/p$, we actually have $E([\bar{Y}_k - G(\bar{\mathbf{X}}_k)]^2) = \zeta(c_h) \cdot N^{-4/(p+4)} + O(N^{-5/(p+4)})$, where

$$\zeta(c_h) = \frac{1}{576}[tr(\nabla^2 G(\mathbf{x}_0))]^2 \cdot c_h^4 + \frac{G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}{f(\mathbf{x}_0)} \cdot c_h^{-p}$$

It can be verified that the best c_h which minimizes $\zeta(c_h)$ is

$$c_h = \left(\frac{144 \cdot p \cdot G(\mathbf{x}_0)[1 - G(\mathbf{x}_0)]}{f(\mathbf{x}_0)[tr(\nabla^2 G(\mathbf{x}_0))]^2} \right)^{\frac{1}{p+4}}$$

Technically speaking, the partitions getting from the k-means may not stratify the conditions for our CLT results. We use k-means since it is a popular clustering algorithm and easy to implement with our representative approach. Given the number of clusters specified by our theoretical study, a k-means clustering algorithm often provides us reasonable results

under our simulation setups. Exploring other types of clustering algorithm is important toward improving our representative approach but out of the scope of this dissertation study. We plan to investigate it in our future research.

CHAPTER 5

SIMULATION STUDY

In this chapter, we compare the performance of the proposed approach with other methods under four different simulation setups. Since the target is to estimate a subspace, the evaluation criteria are different from other statistical methods, i.e., linear regression. In general, we are interested in two properties of a dimension reduction method. One is the structural dimension $\mathbf{d}$. The other one is the distance measurements between estimated and true central subspaces. The details of the evaluation procedure are in Section 5.1. The results for different cases are in Section 5.3. Besides, we also discuss the running time for each SDR method in Section 5.4.

5.1 Evaluation criteria

5.1.1 Structural dimension determination

For SIR and SAVE, we use their own large sample tests (Li, 1991 and Shao et al., 2007) to estimate the d. For PLS, we adopted an eigenvalue selection procedure based on cross-validation. For PRE, we select the first d directions based on the cumulative ratio of those eigenvalues. For MRDR, we consider the procedure of MR calculation as a pre-process of the original data, so we use the same procedure to determine the $\mathbf{d}$ as the original SDR method. Therefore, we adopt the sequential tests from SIR and SAVE. Based on the simulation, the tests can work well for MRDR-SIR. But for MRDR-SAVE, it tends to have type-I error inflation when

N and p is large because SAVE is more complicated than SIR and sensitive to the choice of slice H.

5.1.2 Distance measurement of two linear spaces

The distances between two linear spaces is the distance between two basis, which are actually two matrices. Let $\mathbf{A}_{p\times n}$and $\mathbf{B}_{p\times m}$ be two basis of two linear space. We adopt the Frobenius norm between $Span(A)$ and $Span(B)$ as the distance measurement.

Frobenius norm

$$F = \|P_B - P_A\|_F,$$

where $P_A = A(A^TA)^{-1}A$ and $\|A\|_F = \sqrt{\sum_i \sum_j a_{ij}^2}$ is the Forbenius norm. The range of F is $[0, +\infty]$ and smaller value of F indicates a stronger correlation between $Span(A)$ and $Span(B)$. Note that F equals to 0 only when $Span(A) = Span(B)$. Even though $Span(A) \subset Span(B)$, F will not be equal to zero.

5.1.3 Two comparison strategies

d is known

At first, we assume that the number of true directions d is known, so we measure the distances between the first dth estimated basis and the true basis to evaluate the performance of each SDR method. For each simulation setup, we not only record the distant measurements but also their sample variances which provides us a sense about how reliable the results are.

d is unknown

To mimic the real data analysis, we also assume that d is unknown. Therefore, we need to first estimate the total number of directions and select first d directions as an estimated basis of the central subspace. After we have the estimation of the central space, we could calculate the distance of the estimated subspace and the true space.

We use the power and type-I error to evaluate the performance of direction test. In general, for each active direction (which is the column of a basis of the $\mathcal{S}_{Y|\mathbf{X}}$), we use the power to see if a test can successfully reject its null hypothesis. On the other hand, for non-active directions, we use the type-I to see if a test can stay with its null hypothesis. In this thesis, the significant level is 0.05 and let $\hat{d}$ as the estimated number of direction.

The distance measurement will be dependent on the $\hat{d}$. Therefore, the performance of the dimension detecting will have influences on the distance result. That is, if we miss the active directions or select some non-active directions, then frobenius will be large.

5.2 Simulation setup

We use the latent model for generating the simulation data. The model is

$$Y_i = sign\{H(\mathbf{X_i}) + \epsilon_i\},$$

where $H : \mathbb{R}^p \to \mathbb{R}^1$. Four different structures of H are considered. They are the following:

- $H_1(\mathbf{x}) = \frac{x_1}{0.5+(x_2+1)^2}$
- $H_2(\mathbf{x}) = \frac{\sin(x_1)}{\exp(x_2)}$

- $H_3(\mathbf{x}) = x_1^2 + |x_2|^{1/2}$
- $H_4(\mathbf{x}) = (x_1)^2 \cdot \sin(x_2) \cdot \exp(x_3)$.

For each H, we consider several different combination of N and p, $(N, p) \in \{10^3, 10^4, 10^5, 10^6\} \times \{6, 10, 20\}$. For all the simulation setups, we have $\epsilon_i \sim N(0, 1)$ and $\mathbf{X}_i \sim N(\mathbf{0}_p, I_p)$. Based on the discussion in Section 1.2, $G(\mathbf{X}) = 1 - F_\epsilon(-H(\mathbf{X})) = F_\epsilon(H(\mathbf{X}))$. The dimension reduction methods used for comparison are SIR, SAVE, Partial Least Square (PLS) (Wold, 1975), PRE-SIR and MRDR-SIR and MRDR-SAVE. For the PRE method, there are three different approaches. We use the PRE-SIR$_1$ which is recommended by the authors based on its simplicity. Note that the true space for H_1, H_2, H_3, is (e_1, e_2) and for H_4 is (e_1, e_2, e_3).

5.3 Simulation result

Recall from Section 3.1, the influence of the binary response is different for different inverse moments. Therefore, we compare SDR methods based on what inverse moment they use. In Section 5.3.1, we focus on the results of SDR methods based on the first moment, which are SIR, PRE-SIR, and MRDR-SIR. In the Section 5.3.2, we compare the results for SAVE and MRDR-SAVE.

5.3.1 First inverse moment

In this section, we evaluate the performance of the first-moment based SDR methods. The models we consider are H_1, H_2 and H_4. We skip the H_3 because both of its directions are symmetric with zero so that it cannot be found by the first- inverse-moment methods. Based on the simulation results, we show that the MRDR method can improve the performance of SIR dramatically in terms of dimension detection and distance measurement.

First of all, given d is known, Table II reports the Frobenius norm (F) of different methods. MRDR-SIR out-performances other methods when N becomes large except for model H_4, which will be discussed at the end of this section. Note that for a fixed N, the distance of PRE is smaller than MRDR's. However, they cannot handle the large data very well, see details in Section 5.4. On the contrary, the MRDR method can use the information of large data efficiently, so the distances of MRDR-SIR keeps decreasing when N increases.

Next, we assume the structural dimension is unknown, so we need to estimate d and then a basis of the central subspace. In terms of structural dimension estimation, the MRDR method can work well with the original sequential test of SIR. Table III records the direction test of SDR methods. We adopt the format of the direction test table from (Li and Wang, 2007). We use the power of a direction test when its corresponding direction is from the central space, and we use the type-I error of a direction test if the direction under testing is not from the central space. Based on the result, MRDR-SIR's test has high power for active directions and lower type-I error for the non-active directions. For sample size is small, $N = 10^3$, PRE-SIR can estimate the structural dimension well with the recommended cutoff ratio, but it may not work well for a larger sample size $N = 10^4$. A more detailed discussion is in Section 5.3.1.

For the model H_4, since x_1^2 is symmetric with the origin, which MRDR-SIR can not detect. Therefore, it can only find two directions among the (e_1, e_2, e_3). If we fixed $\hat{d} = 3$, then MRDR-SIR will add a third eigenvector into the $\hat{\eta}_{MRDR-SIR}$, which is not related to the true subspace and therefore affects the distance result in Table II. However, the F distance is improved if we

estimate the $\hat{d}$ instead of using the $d = 3$. The reason is that the direction test of MRDR-SIR selects the first two directions, which can estimate well by MRDR-SIR.

Comparison between MRDR-SIR and PRE-SIR

According to Table II, if the true structural dimension is known, PRE-SIR provides a smaller distance measurement than MRDR-SIR given the same moderate N. The main reason is that MRDR needs a bigger sample size to achieve the same level of accuracy as PRE's because of the representative calculation. Nevertheless, the distance difference between those two methods becomes smaller when N increases. When the sample size is relatively large, say, $N \geq 10^5$, PRE-SIR becomes time-consuming, while MRDR-SIR can still handle the data well, and its distance can be further reduced due to the larger sample size. Therefore, the simulation results demonstrate that our MRDR method is in favor of a larger sample size and also benefits from it.

In practice, the structural dimension is typically unknown, so we will have to estimate the dimension d first. According to Table III, MRDR can detect the active directions with higher power via the large sample test of SIR given a relatively large sample size, say $N = 10^4$ or 10^5. For PRE-SIR, it can also estimate the structural dimension accurately with a moderate sample size, say 10^3. For a larger sample size, it seems that the cutoff ratio needed for PRE-SIR might need certain adjustments.

In order to have a better comparison of these two methods in terms of structural dimension estimation, we use an adaptive cutoff ratio for PRE-SIR to choose the dimension d. The direction test results and distance measurement can be found in Table V and Table VI. Based

on our results, PRE-SIR may have a better performance with the adaptive cutoff ratio, especially when $N = 10^3$, which also shows an advantage of PRE-SIR when N is small. However, the cutoff ratio seems related to both N and p. When N and p are large, it becomes quite sensitive. So one may need to choose the cutoff ratio carefully for different datasets. Compared with PRE-SIR, the simulation results suggest that MRDR-SIR has a more stable direction detection procedure. Although it does not work well for small datasets, when the sample size is large enough, it can estimate the structural dimension correctly.

5.3.2 Second inverse moment

In this section, we investigate and compare the performance of MRDR-SAVE and SAVE. We also list the results of PLS as a reference. The models we consider are H_1, H_3 and H_4. We skip the results of H_2 because it is very similar to H_1's. Recall in Section 3.1, the central subspace estimated by SAVE is larger than other SDR methods', so it should have had a better performance. However, in practice, SAVE does not demonstrate an obvious advantage over other methods. The main reason is that it needs a large sample size to get an accurate estimation and is sensitive to the number of slices (see details in Li et al., 2007). Despite these issues, we show that SAVE and MRDR-SAVE still have the potentials to out-performance other methods when the sample size is large enough.

Table VII reports the distance measurement for those three models given the structural dimension d. Since the sample size is relatively large in our simulation setups, SAVE and MRDR-SAVE do have a good performance in terms of F distance. Under the model H_4, SAVE is affected by the binary response in terms of recovering the structural dimension and F distance.

Since MRs provide more information than the binary response, the performance MRDR-SAVE is improved compared to the original SAVE. For model H_1 and H_3, SAVE is a little better than MRDR-SAVE. Because if SAVE itself can recover the central subspace under the binary response, then MRDR-SAVE will become less efficient for having the "additional" partition step. However, based on the results, the difference between MRDR-SAVE and SAVE becomes smaller when N increases, which suggests that even if the original SDR can recover the full central subspace, MRDR methods are at least as good as the original SDR methods when N is large.

Since sequential test of SAVE is not very stable and sensitive to the choice of slice, when we assume d is unknown we observe some mixed results. In Table VIII, we find that SAVE fails to detect all directions of the structural dimension for H_1, but the MRDR-SAVE can find all of them. However, when N and p is large, the direction test of MRDR- SAVE becomes unstable. That is, it tends to select all the directions as significant. Therefore, in Table VIII and Table IX, the advantage of MRDR-SAVE in model H_4 becomes less significant. This issue is because of SAVE's sensitivity to the choice of the number of slices.

5.4 Computation efficiency

As we mentioned before, the scalability with massive data is an advantage of the proposed method, which is demonstrated by Table X. In the table, we record running time (in minutes) of different SDR methods. For each method, we try different combinations of $N = \{10^4, 10^5, 10^6\}$ and $p = \{6, 10, 20\}$. In general, the computational time of a method increases with the increase of N and p. However, the increase rate based on N and p varies from one method to another.

Based on the simulation, SIR, SAVE and PLS are fast and efficient even when N and p are large. The reason is mainly that their algorithms only involve linear operations. Note that the running time of SAVE increases faster than SIR because of the second-moment estimation. As for the PRE method, its running time is relatively large, so we only run four different simulation setups to demonstrate its computational intensity. For instance, when $N = 50000, p = 10$, the running time of PRE-SIR for one iteration is about 38 hours. The main reason is that the PRE method needs to solve the WSVM repeatedly, which becomes time-consuming when N is large. For the MRDR method, it runs much faster compared to the PRE method. If we only consider the running times on MRs, they are even faster than the original SAVE and SIR. Note that the computational time MRDR is dominated by the partition procedure. Recall in Section 3.2.2, we use K-means for partition with time complexity is up to $\mathcal{O}(pN^{1+(p/p+4)})$. Note that it is faster than $\mathcal{O}(N^2)$, but still could be improved. There are solutions available to reduce further the running time of K-means, which we will discuss in Chapter 7.

TABLE II: FROBENIUS DISTANCE GIVEN D FOR THE FIRST MOMENT

model	p	logn	MRDR-SIR	PRE-SIR	SIR	PLS
H1	6	3	1.08(0.06)	0.63(0.06)	1.23(0.05)	1.24(0.04)
		4	0.3(0.01)	0.22(0.01)	1.27(0.03)	1.24(0.04)
		5	0.1(0)	.	1.27(0.03)	1.26(0.03)
		6	0.04(0)	.	1.21(0.05)	1.23(0.04)
	10	3	1.28(0.03)	0.91(0.06)	1.32(0.01)	1.35(0.01)
		4	0.42(0.01)	0.38(0.04)	1.31(0.02)	1.34(0.01)
		5	0.11(0)	.	1.32(0.02)	1.33(0.01)
		6	0.04(0)	.	1.32(0.01)	1.32(0.01)
	20	3	1.45(0.01)	1.19(0.03)	1.37(0)	1.44(0)
		4	0.82(0.02)	0.65(0.06)	1.37(0)	1.39(0)
		5	0.22(0)	.	1.37(0)	1.37(0)
		6	0.06(0)	.	1.38(0)	1.37(0)
H2	6	3	0.96(0.09)	0.52(0.05)	1.21(0.05)	1.24(0.04)
		4	0.22(0.01)	0.18(0.01)	1.28(0.03)	1.24(0.04)
		5	0.08(0)	.	1.29(0.02)	1.26(0.03)
		6	0.03(0)	.	1.25(0.04)	1.23(0.04)
	10	3	1.2(0.05)	0.8(0.06)	1.32(0.01)	1.35(0.01)
		4	0.31(0.01)	0.3(0.02)	1.31(0.02)	1.34(0.01)
		5	0.09(0)	.	1.33(0.02)	1.33(0.01)
		6	0.03(0)	.	1.33(0.01)	1.32(0.01)
	20	3	1.41(0.01)	1.05(0.03)	1.38(0)	1.43(0)
		4	0.62(0.01)	0.49(0.04)	1.38(0)	1.39(0)
		5	0.16(0)	.	1.37(0)	1.37(0)
		6	0.05(0)	.	1.38(0)	1.37(0)
H4	6	3	1.5(0.05)	1.29(0.06)	0.64(0.26)	1.52(0.04)
		4	1.25(0.05)	1.13(0.1)	0.44(0.31)	1.54(0.05)
		5	1.25(0.05)	.	0.42(0.39)	1.53(0.03)
		6	1.31(0.02)	.	0.48(0.44)	1.52(0.02)
	10	3	1.77(0.02)	1.62(0.02)	0.89(0.22)	1.8(0.01)
		4	1.42(0.02)	1.37(0.02)	0.61(0.35)	1.77(0.02)
		5	1.34(0.01)	.	0.49(0.41)	1.76(0.02)
		6	1.34(0.01)	.	0.46(0.42)	1.73(0.02)
	20	3	1.99(0.01)	1.84(0.01)	1.13(0.16)	1.94(0.01)
		4	1.73(0.01)	1.55(0.01)	0.82(0.31)	1.9(0)
		5	1.42(0)	.	0.75(0.43)	1.89(0)
		6	1.39(0)	.	0.58(0.46)	1.88(0.01)

Frobenius norm (its sample variance in parentheses) between the $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ based on 100 independent iterations
logn: log(N)
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage

TABLE III: DIRECTION TEST FOR THE FIRST MOMENT

model	p	test	MRDR-SIR				PRE-SIR		PLS			
		logn	3.00	4.00	5.00	6.00	3.00	4.00	3.00	4.00	5.00	6.00
H1	6	0D vs 1D	0.94	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.06	0.99	1.00	1.00	0.16	0.00	1.00	1.00	1.00	1.00
		2D vs 3D	0.00	0.01	0.02	0.00	0.00	0.00	0.77	0.67	0.51	0.50
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00	0.48	0.28	0.24	0.28
	10	0D vs 1D	0.98	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.12	1.00	1.00	1.00	0.23	0.00	1.00	1.00	1.00	1.00
		2D vs 3D	0.00	0.02	0.00	0.01	0.00	0.00	0.83	0.69	0.53	0.64
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00	0.49	0.36	0.30	0.32
	20	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.03	0.67	1.00	1.00	0.68	0.00	1.00	1.00	1.00	1.00
		2D vs 3D	0.01	0.00	0.07	0.02	0.17	0.00	0.92	0.73	0.59	0.54
		3D vs 4D	0.00	0.00	0.00	0.00	0.01	0.00	0.52	0.39	0.32	0.25
H2	6	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.18	1.00	1.00	1.00	0.30	0.11	1.00	1.00	1.00	1.00
		2D vs 3D	0.03	0.01	0.01	0.00	0.00	0.00	0.74	0.59	0.54	0.53
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00	0.35	0.27	0.27	0.28
	10	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.15	1.00	1.00	1.00	0.34	0.01	1.00	1.00	1.00	1.00
		2D vs 3D	0.02	0.07	0.03	0.00	0.00	0.00	0.83	0.71	0.50	0.55
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00	0.45	0.43	0.22	0.31
	20	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.06	0.99	1.00	1.00	0.73	0.00	1.00	1.00	1.00	1.00
		2D vs 3D	0.00	0.00	0.06	0.06	0.19	0.00	0.92	0.74	0.57	0.48
		3D vs 4D	0.00	0.00	0.00	0.00	0.03	0.00	0.52	0.45	0.29	0.25
H4	6	0D vs 1D	0.72	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.04	0.78	1.00	1.00	0.25	0.00	1.00	1.00	1.00	1.00
		2D vs 3D	0.02	0.02	0.01	0.00	0.00	0.00	0.71	0.59	0.49	0.48
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00	0.39	0.30	0.25	0.28
	10	0D vs 1D	0.79	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.05	0.72	1.00	1.00	0.10	0.00	1.00	1.00	1.00	1.00
		2D vs 3D	0.00	0.01	0.04	0.01	0.00	0.00	0.68	0.57	0.57	0.52
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00	0.38	0.35	0.25	0.21
	20	0D vs 1D	0.86	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.05	0.20	1.00	1.00	0.53	0.00	1.00	1.00	1.00	1.00
		2D vs 3D	0.00	0.02	0.06	0.06	0.05	0.00	0.88	0.70	0.56	0.43
		3D vs 4D	0.00	0.00	0.00	0.00	0.02	0.00	0.46	0.39	0.35	0.23

Direction test based 100 on independent iterations.
Since the true space is (e_1, e_2) for model H_1 and H_2, we report the power of the first two direction tests and type-I error for the rest of tests. Similarly, we report the power of for first three test for H_4. The significant level is 0.05.
For simplicity, we report only the first four test results for each simulation.
logn: log(N)

TABLE IV: FROBENIUS DISTANCE BASED ON TEST FOR FIRST MOMENT

model	p	logn	MRDR-SIR	PRE-SIR	SIR	PLS
H1	6	3	1.04(0.01)	0.93(0.04)	1.02(0)	1.59(0.1)
		4	0.31(0.02)	1(0)	1(0)	1.5(0.09)
		5	0.12(0.02)	.	1(0)	1.44(0.06)
		6	0.04(0)	.	1(0)	1.43(0.09)
	10	3	1.11(0.01)	0.95(0.03)	1.04(0)	1.79(0.1)
		4	0.43(0.02)	1(0)	1(0)	1.65(0.1)
		5	0.11(0)	.	1(0)	1.57(0.08)
		6	0.05(0.01)	.	1(0)	1.6(0.07)
	20	3	1.15(0.01)	1.19(0.03)	1.08(0)	1.97(0.13)
		4	0.88(0.02)	1.01(0)	1.01(0)	1.78(0.14)
		5	0.28(0.04)	.	1(0)	1.67(0.1)
		6	0.08(0.02)	.	1(0)	1.62(0.09)
H2	6	3	1.04(0.02)	0.82(0.1)	1.02(0)	1.54(0.09)
		4	0.23(0.01)	0.9(0.08)	1(0)	1.47(0.1)
		5	0.09(0.01)	.	1(0)	1.47(0.07)
		6	0.03(0)	.	1(0)	1.42(0.08)
	10	3	1.09(0.01)	0.88(0.05)	1.04(0)	1.78(0.11)
		4	0.36(0.04)	0.99(0.01)	1(0)	1.67(0.11)
		5	0.11(0.03)	.	1(0)	1.53(0.06)
		6	0.03(0)	.	1(0)	1.56(0.08)
	20	3	1.15(0.01)	1.11(0.03)	1.07(0)	1.93(0.11)
		4	0.62(0.01)	1.01(0)	1.01(0)	1.79(0.11)
		5	0.22(0.04)	.	1(0)	1.68(0.13)
		6	0.11(0.05)	.	1(0)	1.61(0.1)
H4	6	3	1.48(0.01)	1.36(0.02)	1.43(0)	1.56(0.04)
		4	1.16(0.02)	1.42(0)	1.42(0)	1.56(0.04)
		5	1.01(0)	.	1.41(0)	1.53(0.04)
		6	1(0)	.	1.41(0)	1.5(0.03)
	10	3	1.52(0)	1.44(0)	1.46(0)	1.85(0.05)
		4	1.23(0.02)	1.42(0)	1.42(0)	1.82(0.07)
		5	1.03(0.01)	.	1.41(0)	1.73(0.05)
		6	1.01(0)	.	1.41(0)	1.71(0.04)
	20	3	1.58(0)	1.56(0.01)	1.51(0)	2.1(0.09)
		4	1.44(0.01)	1.43(0)	1.43(0)	2(0.09)
		5	1.07(0.01)	.	1.42(0)	1.9(0.09)
		6	1.03(0.01)	.	1.41(0)	1.84(0.06)

Frobenius norm (its sample variance in parentheses) between the $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ based on 100 independent iterations
logn: log(N)
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage

TABLE V: DIRECTION TEST FOR MRDR-SIR AND PRE-SIR

model	p	test	MRDR-SIR				PRE-SIR	
		logn	3.00	4.00	5.00	6.00	3.00	4.00
H1	6	0D vs 1D	0.94	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.06	0.99	1.00	1.00	0.81	0.59
		2D vs 3D	0.00	0.01	0.02	0.00	0.10	0.00
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00
	10	0D vs 1D	0.98	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.12	1.00	1.00	1.00	0.74	0.48
		2D vs 3D	0.00	0.02	0.00	0.01	0.20	0.00
		3D vs 4D	0.00	0.00	0.00	0.00	0.01	0.00
	20	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.03	0.67	1.00	1.00	0.57	0.55
		2D vs 3D	0.01	0.00	0.07	0.02	0.07	0.00
		3D vs 4D	0.00	0.00	0.00	0.00	0.01	0.00

Direction test based 100 on independent iterations.
For simplicity, we report only the first four test results for each simulation.
logn: log(N)

TABLE VI: FROBENIUS DISTANCE BASED ON TEST FOR MRDR-SIR AND PRE-SIR

model	p	logn	MRDR-SIR	PRE-SIR	SIR
H1	6	3	1.04(0.01)	0.72(0.08)	1.02(0)
		4	0.31(0.02)	0.51(0.17)	1(0)
		5	0.12(0.02)	.	1(0)
		6	0.04(0)	.	1(0)
	10	3	1.11(0.01)	0.97(0.04)	1.04(0)
		4	0.43(0.02)	0.64(0.15)	1(0)
		5	0.11(0)	.	1(0)
		6	0.05(0.01)	.	1(0)
	20	3	1.15(0.01)	1.14(0.02)	1.08(0)
		4	0.88(0.02)	0.72(0.07)	1.01(0)
		5	0.28(0.04)	.	1(0)
		6	0.08(0.02)	.	1(0)

Frobenius norm (its sample variance in parentheses) between the $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ based on 100 independent iterations
logn: log(N)
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage

TABLE VII: FROBENIUS DISTANCE GIVEN D FOR THE SECOND MOMENT

model	p	logn	MRDR-SAVE	SAVE	PLS
H1	6	4	1.35(0.04)	0.16(0)	1.24(0.04)
		5	0.12(0)	0.05(0)	1.26(0.03)
		6	0.05(0)	0.02(0)	1.23(0.04)
	10	4	1.12(0.07)	0.23(0)	1.34(0.01)
		5	0.22(0)	0.07(0)	1.33(0.01)
		6	0.06(0)	0.02(0)	1.32(0.02)
	20	4	1.39(0.01)	0.35(0)	1.39(0)
		5	0.86(0.04)	0.11(0)	1.37(0)
		6	0.13(0)	0.03(0)	1.37(0)
H3	6	4	1.12(0.07)	0.26(0.01)	1.65(0.03)
		5	0.76(0.1)	0.07(0)	1.67(0.03)
		6	0.25(0.01)	0.02(0)	1.6(0.04)
	10	4	1.24(0.04)	0.38(0.01)	1.79(0.01
		5	0.8(0.1)	0.11(0)	1.8(0.01)
		6	0.23(0)	0.04(0)	1.78(0.01)
	20	4	1.36(0.01)	0.56(0.02)	1.91(0)
		5	0.5(0.02)	0.17(0)	1.91(0)
		6	0.12(0)	0.06(0)	1.9(0)
H4	6	4	1.17(0.1)	1.23(0.06)	1.54(0.05)
		5	0.46(0.06)	1.18(0.06)	1.53(0.03)
		6	0.15(0)	1.18(0.08)	1.52(0.02)
	10	4	1.33(0.04)	1.37(0.01)	1.77(0.02)
		5	0.96(0.07)	1.32(0.01)	1.76(0.02)
		6	0.24(0)	1.29(0.03)	1.73(0.02)
	20	4	1.82(0.02)	1.47(0)	1.9(0)
		5	1.04(0.05)	1.38(0.01)	1.89(0)
		6	0.38(0)	1.38(0)	1.88(0.01)

Frobenius norm (its sample variance in parentheses) between the $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ based on 100 independent iterations
logn: log(N)
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage

TABLE VIII: DIRECTION TEST FOR THE SECOND MOMENT

model	p	test	MRDR-SAVE			SAVE			PLS		
	logn		4	5	6	4	5	6	4	5	6
H1	6	0D vs 1D	0.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.00	0.32	1.00	0.08	0.05	0.03	1.00	1.00	1.00
		2D vs 3D	0.00	0.00	0.00	0.04	0.07	0.04	0.67	0.51	0.50
		3D vs 4D	0.00	0.00	0.00	0.00	0.00	0.00	0.28	0.24	0.28
	10	0D vs 1D	0.55	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.00	0.00	0.93	0.05	0.06	0.02	1.00	1.00	1.00
		2D vs 3D	0.00	0.00	0.00	0.03	0.05	0.02	0.70	0.52	0.64
		3D vs 4D	0.00	0.00	0.00	0.00	0.02	0.01	0.37	0.30	0.33
	20	0D vs 1D	0.67	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.05	0.99	1.00	0.04	0.01	0.00	1.00	1.00	1.00
		2D vs 3D	0.00	0.95	1.00	0.06	0.07	0.05	0.71	0.60	0.53
		3D vs 4D	0.00	0.66	1.00	0.00	0.00	0.00	0.38	0.31	0.25
H3	6	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.00	0.02	0.65	1.00	1.00	1.00	0.53	0.50	0.52
		2D vs 3D	0.00	0.00	0.01	0.04	0.05	0.04	0.26	0.24	0.25
		3D vs 4D	0.00	0.00	0.00	0.00	0.01	0.00	0.15	0.12	0.08
	10	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.00	0.00	0.85	1.00	1.00	1.00	0.65	0.44	0.44
		2D vs 3D	0.00	0.00	0.00	0.04	0.04	0.03	0.33	0.12	0.20
		3D vs 4D	0.00	0.00	0.00	0.01	0.00	0.00	0.17	0.08	0.14
	20	0D vs 1D	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.31	1.00	1.00	0.79	1.00	1.00	0.68	0.52	0.53
		2D vs 3D	0.01	0.99	1.00	0.04	0.03	0.03	0.39	0.25	0.21
		3D vs 4D	0.00	0.97	1.00	0.00	0.00	0.00	0.16	0.10	0.10
H4	6	0D vs 1D	0.82	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.01	1.00	1.00	0.44	0.53	0.52	1.00	1.00	1.00
		2D vs 3D	0.00	0.03	0.96	0.06	0.03	0.05	0.59	0.49	0.48
		3D vs 4D	0.00	0.00	0.07	0.01	0.00	0.00	0.30	0.25	0.28
	10	0D vs 1D	0.79	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.00	1.00	1.00	0.25	0.43	0.44	1.00	1.00	1.00
		2D vs 3D	0.00	0.00	0.72	0.02	0.11	0.06	0.57	0.57	0.52
		3D vs 4D	0.00	0.00	0.03	0.00	0.02	0.00	0.35	0.25	0.21
	20	0D vs 1D	0.79	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
		1D vs 2D	0.15	1.00	1.00	0.18	0.20	0.36	1.00	1.00	1.00
		2D vs 3D	0.01	1.00	1.00	0.06	0.07	0.04	0.70	0.56	0.43
		3D vs 4D	0.00	0.98	1.00	0.00	0.00	0.00	0.39	0.35	0.23

Direction test based 100 on independent iterations.
Since the true space is (e_1, e_2) for model H_1 and H_3, we report the power of the first two direction tests and type-I error for the rest of tests. Similarly, we report the power of for first three test for H_4. The significant level is 0.05.
For simplicity, we report only the first four test results for each simulation.
logn: log(N)

TABLE IX: FROBENIUS DISTANCE BASED ON TEST FOR THE SECOND MOMENT

model	p	logn	MRDR-SAVE	SAVE	PLS
H1	6	4	*	0.91(0.07)	1.5(0.09)
		5	0.72(0.17)	0.94(0.05)	1.44(0.06)
		6	0.05(0)	0.95(0.05)	1.43(0.09)
	10	4	1.01(0)	0.97(0.03)	1.66(0.1)
		5	1(0)	0.96(0.05)	1.57(0.09)
		6	0.13(0.06)	0.98(0.02)	1.6(0.08)
	20	4	1.1(0.01)	0.99(0.02)	1.77(0.14)
		5	1.56(0.13)	0.95(0.05)	1.67(0.1)
		6	4.24(0)	0.96(0.04)	1.62(0.09)
H3	6	4	1.01(0)	0.29(0.03)	1.66(0.03)
		5	1(0)	0.13(0.05)	1.65(0.03)
		6	0.52(0.14)	0.06(0.04)	1.61(0.03)
	10	4	1.02(0)	0.41(0.03)	1.84(0.06)
		5	1(0)	0.15(0.03)	1.73(0.03)
		6	0.46(0.14)	0.07(0.03)	1.75(0.03)
	20	4	1.15(0.03)	0.67(0.06)	1.96(0.07)
		5	1.77(0.1)	0.2(0.02)	1.88(0.05)
		6	4.24(0)	0.08(0.03)	1.86(0.05)
H4	6	4	1.42(0)	1.25(0.04)	1.56(0.04)
		5	0.99(0.01)	1.2(0.05)	1.53(0.04)
		6	0.26(0.1)	1.19(0.05)	1.5(0.03)
	10	4	1.43(0)	1.34(0.03)	1.82(0.07)
		5	1.01(0)	1.25(0.04)	1.73(0.05)
		6	0.49(0.14)	1.24(0.04)	1.71(0.04)
	20	4	1.69(0.01)	1.39(0.02)	2(0.09)
		5	1.78(0.14)	1.33(0.03)	1.9(0.09)
		6	4.12(0)	1.27(0.04)	1.84(0.06)

Frobenius norm (its sample variance in parentheses) between the $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ based on 100 independent iterations
logn: log(N)
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage
$*$ means missing because of the direction test

TABLE X: EMPIRICAL COMPUTATION TIME FOR MRDR

p	logn	SIR	SAVE	PRE-SIR	MR-SIR	MR-SAVE	Clustering
6	4	0	0	35.4	0	0	0
	5	0.02	0.01	.	0	0	0.06
	6	0.22	0.26	.	0	0	1.8
10	4	0	0	39.8	0	0	0.01
	5	0.03	0.03	.	0	0	0.24
	6	0.44	0.57	.	0	0	11.24
20	4	0.01	0.01	55.1	0	0	0.02
	5	0.07	0.08	.	0	0	1.33
	6	0.99	1.23	.	0.03	0.03	94.24

Empirical computational time (in minutes) calculated from 100 independent iterations.
The machine equips Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz and 32GB memory
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage
We also have run a PRE-SIR under $N = 50000$ and $p = 10$ for one iteration, its ruining time is 2335.95 mintues

CHAPTER 6

APPLICATION ON ELECTRICAL GRID STABILITY DATA

In this chapter, we first introduce the background of the Electrical Grid Stability (EGS) data in Section 6.1. Then we compare the result of SDR methods on the data set. Since we do not know the true central space of the EGS data, it is not straightforward to compare those SDR results. In order to have a better view of the performances of the SDR methods on EGS data, we generate a simulated data based on EGS via a non-parametric method. Based on the simulated data, the proposed data again demonstrate the advantage over other methods. See details in Section 6.2.

6.1 EGS data

An electrical grid is an energy network with electricity producers who supply the energy and consumers who demand the energy. One property of the network is its stability. The stability is roughly defined as the balance between the supply and demand of an energy network. In order to keep a network stable, we may need to have a control system. Each control system assumes a specific model for the grid. The EGS dataset is used for studying a system called Decentral Smart Grid Control (DSGC). More details of the data and the system could be found at (Arzamasov et al., 2018). In the paper, the author has introduced one way to define the stability of a grid system. That is, a network is linearly unstable if the maximal real part of its characteristic equations' root is positive, is stable if the root is negative.

The EGS data is generated from a 4-node star system based on DSGC. The response is a binary variable indicating the stability of the gird. Among those four nodes in the network, one node is a producer and the three of them are consumers. Each node has 3 measurements:

- Reaction time of a node to a price change $\tau_j, j = 1, \ldots, 4$
- Mechanical power produced/consumed $P_j, j = 1, \ldots, 4$
- Coefficient proportional to price elasticity $\gamma_j, j = 1, \ldots, 4$.

which we believe are related to the stability of a electrical grid . Note that the original EGS data has 12 predictors in total. Based on the (Arzamasov et al., 2018), we transform each kind of predictor in to its max, average and min, such as $\gamma_{max} = \max(\gamma_j, j = 1, \ldots, 4)$, $\gamma_{ave} = \frac{\sum_j^4 \gamma_j}{4}$ and $\gamma_{min} = \min(\gamma_j, j = 1, \ldots, 4)$. Therefore, we have 9 predictors in total.

The general goal is to use the data to get insights into the structure of the DSGC system. We want to know which predictors may influence the stability of the whole grid. We apply MRDR on the EGS data and show that the proposed method helps us to understand the DSGC system.

6.2 Dimension reduction on EGS data

Since we have made a comprehensive comparison of the performance of different SDR methods in Chapter 5, we only use the first inverse moment-based methods, which are SIR, PRE-SIR, and MRDR-SIR in this section. First of all, we apply those methods on the EGS data to calculate the estimated basis of central subspace, as $\hat{\eta}_{PRE}$ and $\hat{\eta}_{MRDR}$. We show the difference and similarity of that estimated basis. However, it is hard to compare two dimension reduction

methods for real data because the true space is unknown to us. In order to have a better understanding of those two methods, we also simulate a large data set based on EGS via an additive model with the natural spline.

First of all, we apply PRE-SIR and MRDR-SIR to the EGS data and compare their results via the distance measurement. Here, we introduce another distance measurement named vector correlation. The vector correlation was introduced in (Hotelling, 1936). Let R^2 be the vector correlation, then we have

$$R^2 = 1 - \prod_{i=1}^{k} \rho_i^2,$$

where $\rho_i, i = 1, \ldots k$, is the ith non-zero eigenvalue of $B^T A A^T B$ and $k = min(rank(A), rank(B))$. Small value of R^2 indicates strong correlation between $\mathbf{A}$ and $\mathbf{B}$, therefore, close distance between $Span(A)$ and $Span(B)$. The range of R^2 is $[0, 1]$. If $Span(A) \subset Span(B)$, then $R^2 = 0$.

We calculate the distances between η_{PRE1} and η_{MRDR1}, they are 0.006 and 0.15 for R^2 and F, which suggests that the first directions found by both methods are closed to each other. But we do not know the true subspace, so there is not much we can tell futher about those results.

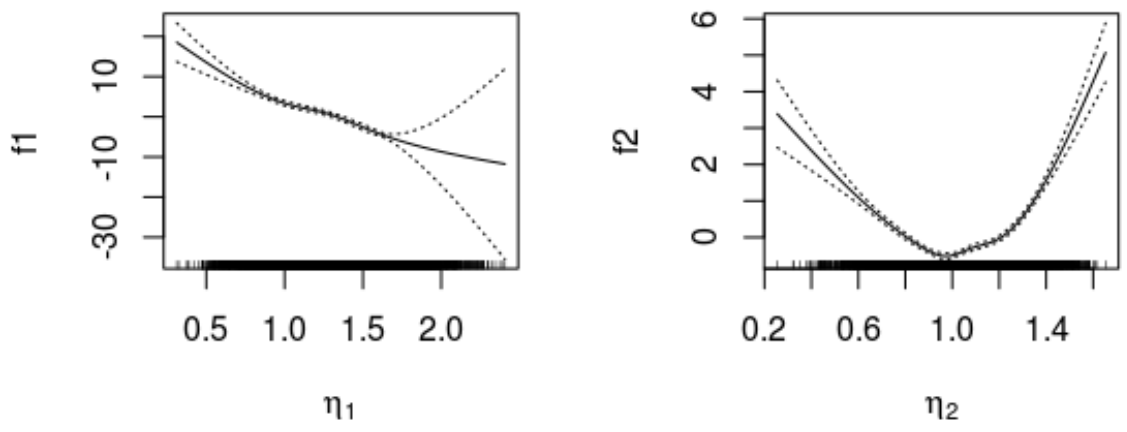
6.2.1 Directions estimated by EGS simulated data

Since the central subspace of the EGS is unknown us, we use a simulated data based on the EGS data in order to have a better understanding of the performances of those two methods. We use a generalized additive model for generating the data. The model depends on two directions of the predictors as following,

$$g(E(Y|\mathbf{X})) = f_1(\eta_1^T \mathbf{X}) + f_2(\eta_2^T \mathbf{X}),$$

where $X \sim N(\mathbf{0}_9, I_9)$, η_1 and η_2 are two 9×1 vectors and f_1 and f_2 are arbitrary functions. In order to make the simulated data as closed to EGS data as possible, we try to estimate most of the parameters from the EGS data. First of all, we apply the SAVE on the EGS data to calculate $\hat{\eta}_1$ and $\hat{\eta}_2$. We assume an additive logistic model, which is commonly used for binary data. Therefore, g is the logit function. Then, we use the natural spline to estimate f_1, f_2. We use the natural spline because it is a non-parametric method so that we could have less assumption for our data generating process. Figure 1 is the plotted curves of $\hat{f}_1$ and $\hat{f}_2$ estimated via natural splines. Note that both functions are not symmetric with the origin. Therefore the first inverse moment method should be able to find all of them.

Figure 1: Natural spline function estimated from EGS



Estimating result

We adopt a similar procedure in Chapter 5 to compare the results of the two methods. It can be shown that with the sample size N increasing, the distance measurement of the proposed method keeps decreasing because it takes advantage of the information provided by large data.

TABLE XI: FROBENIUS DISTANCE OF EGS GIVEN D

model	p	logn	PRE-SIR	MRDR-SIR
EGS	9	4	0.17(0)	0.7(0.06)
		5	.	0.19(0)
		6	.	0.06(0)

Frobenius norm (its sample variance in parentheses) between the $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ based on 100 independent iterations
logn: log(N)
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage

TABLE XII: DIRECTION TEST OF EGS

model	p	test	PRE-SIR	MRDR-SIR		
	logn		4	4	5	6
EGS	9	0D vs 1D	1	1.00	1.00	1
		1D vs 2D	0.45	0.50	1.00	1
		2D vs 3D	0	0.02	0.02	0
		3D vs 4D	0	0.00	0.00	0
		4D vs 5D	0	0.00	0.00	0
		5D vs 6D	0	0.00	0.00	0
		6D vs 7D	0	0.00	0.00	0
		7D vs 8D	0	0.00	0.00	0
		8D vs 9D	0	0.00	0.00	0

Direction test based on 100 independent iterations.
Since the true space is (e_1, e_2) for the simulated data, we report the power of the first two direction tests and type-I error for the rest of tests. The significant level is 0.05.
For simplicity, we report only the first four test results for each simulation.
logn: log(N)

TABLE XIII: FROBENIUS DISTANCE OF EGS BASED ON TEST

model	p	logn	PRE-SIR	MRDR-SIR
EGS	9	4	0.88(0.07)	0.86(0.05)
		5	.	0.21(0.02)
		6	.	0.06(0)

Frobenius norm (its sample variance in parentheses) between the $\hat{\boldsymbol{\eta}}$ and $\boldsymbol{\eta}$ based on 100 independent iterations
logn: log(N)
. means missing because a method's running time for each iteration is larger than 1 day or memory shortage
$*$ means missing because of the direction test

6.3 Classification based on the simulated data

In this section, we are trying to demonstrate the benefit that dimension reduction methods can provide for the classification task. That is, compared with the full data set, we could use

a low-dimension projection of the original data to archive a similar classification accuracy. In order to reduce the dimension, we first project the original data into the estimated subspace, which is denoted as $\mathbf{X}^*$, then we have

$$\mathbf{X}^* = \hat{\boldsymbol{\eta}}^T \mathbf{X} \in \mathbb{R}^d, d << p,$$

where $\hat{\boldsymbol{\eta}}$ is an estimated basis of the central subspace. As for the dimension reduction method, we use the SIR, PRE-SIRE and MRDR-SIR. Recall in Section 6.2, SIR can only find one direction, but MRDR-SIR finds two direction with high power, which should contain more information for predicting Y.

For classification, we use the Support Vector Machine (SVM) because it could identify a non-linear boundary. We apply SVM on the low-dimension data estimated via SIR (p = 1), PRE-SIR (p=2, after adjusting the cutoff ratio based on the simulation results), MRDR-SIR (p = 2) and full dataset (p = 9). We use the training accuracy calculated from a simulated data with sample size $N = 10^4$ to evaluate the performance of different SDR methods. Based on Table XIV, we observe that the one direction estimated by SIR have a lower accuracy because of missing one active direction. However, the two dimension subspace gives the similar accuracy as the full data set.

TABLE XIV: PREDICTION ACCURACY

SDR method	p	Accuracy
SIR	1	0.76
PRE-SIR	2	0.90
MRDR-SIR	2	0.91
Full data	9	0.92

$N = 10^4$

CHAPTER 7

CONCLUSION AND DISCUSSION

7.1 Conclusion

In this thesis, we develop two approaches to improve the performance of SIR and SAVE on large data. The online algorithms of SIR and SAVE not only remove the memory obstacle but also reduce the computational time. The simulation results demonstrate its computational efficiency. On the other hand, the mean representative dimension reduction (MRDR) focuses on statistical efficiency. In the binary response data, the mean representative of each block estimates the conditional probability, which is continuous and therefore contains more information than original data. Both theoretical study and simulation results show that mean representatives improve the performance of SDR methods under binary response. Moreover, the proposed method is less computationally intensive than the existing method. Besides, since the calculation of mean representatives can be considered as a data pre-processing procedure, the MRDR can cooperate with other inverse-moment based SDR methods.

7.2 Discussion

7.2.1 Computational time

Compared to the existing method, the MRDR method has an advantage in dealing with the large dataset. However, the result is still not fully satisfied. Based on the Table X, the computational time of partition increases fast with the increase of N and p. The main reason

is that we use K-means to cluster the data into blocks. One way to alleviate the computational intensity is to adopt a faster clustering algorithm. For instance, there are K-means algorithms that are linear in N and p, so that the time complexity will reduce to $\mathcal{O}(\mathsf{Np})$ (See details in Manning et al., 2008). Besides, there are also parallel K-means algorithms (Kumar et al., 2011, Miller and Boxer, 2012), which can dramatically reduce the running time of the partition step.

7.2.2 Structural dimension determination for MRDR

Another aspect of MRDR we would like to improve is how to decide the structural dimension based on MRs. Although the simulation studies suggest that MRDR can work well with the large sample test of SIR, the situation becomes more complicated for other SDR methods, like SAVE. In order to have a validated large sample test, one may need to study the asymptotic properties of each different SDR method on MRs.

CITED LITERATURE

Arzamasov, V., Böhm, K., and Jochem, P.: Towards concise models of grid stability. pages 1–6, 2018.

Bura, E. and Yang, J.: Dimension estimation in sufficient dimension reduction: a unifying approach. Journal of Multivariate Analysis, 102(1):130–142, 2011.

Cook, R. D.: Using dimension-reduction subspaces to identify important inputs in models of physical systems. In Proceedings of the section on Physical and Engineering Sciences, pages 18–25, 1994.

Cook, R. D.: Regression graphics: ideas for studying regressions through graphics, volume 482. John Wiley & Sons, 2009.

Cook, R. D. et al.: Testing predictor contributions in sufficient dimension reduction. The Annals of Statistics, 32(3):1062–1092, 2004.

Cook, R. D. et al.: Fisher lecture: Dimension reduction in regression. Statistical Science, 22(1):1–26, 2007.

Cook, R. D. and Lee, H.: Dimension reduction in binary response regression. Journal of the American Statistical Association, 94(448):1187–1200, 1999.

Cook, R. D. and Ni, L.: Sufficient dimension reduction via inverse regression: A minimum discrepancy approach. Journal of the American Statistical Association, 100(470):410–428, 2005.

Cook, R. D. and Weisberg, S.: Sliced inverse regression for dimension reduction: Comment. Journal of the American Statistical Association, 86(414):328–332, 1991.

DasGupta, A.: Asymptotic Theory of Statistics and Probability. Springer Science & Business Media, 2008.

Ferguson, T. S.: A Course in Large Sample Theory. Chapman & Hall/CRC, 1996.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B.: Bayesian data analysis. Chapman and Hall/CRC, 2013.

Hotelling, H.: Relations between two sets of variates. Biometrika, 28(3/4):321–377, 1936.

Kevin, L.: Sliced inverse regression for big data analysis. 2014.

Kumar, J., Mills, R. T., Hoffman, F. M., and Hargrove, W. W.: Parallel k-means clustering for quantitative ecoregion delineation using large data sets. Procedia Computer Science, 4:1602–1611, 2011.

Lee, K.-Y., Li, B., Chiaromonte, F., et al.: A general theory for nonlinear sufficient dimension reduction: Formulation and estimation. The Annals of Statistics, 41(1):221–249, 2013.

Li, B. and Wang, S.: On directional regression for dimension reduction. Journal of the American Statistical Association, 102(479):997–1008, 2007.

Li, K.-C.: Sliced inverse regression for dimension reduction. Journal of the American Statistical Association, 86(414):316–327, 1991.

Li, K.-C.: On principal hessian directions for data visualization and dimension reduction: Another application of stein's lemma. Journal of the American Statistical Association, 87(420):1025–1039, 1992.

Li, K. and Yang, J.: Score-matching representative approach for big data analysis with generalized linear models. arXiv preprint arXiv:1811.00462, 2018.

Li, Y., Zhu, L.-X., et al.: Asymptotics for sliced average variance estimation. The Annals of Statistics, 35(1):41–69, 2007.

Ma, Y. and Zhu, L.: A review on dimension reduction. International Statistical Review, 81(1):134–150, 2013.

Manning, C. D., Raghavan, P., and Schütze, H.: Introduction to information retrieval. Cambridge university press, 2008.

Miller, R. and Boxer, L.: Algorithms sequential & parallel: A unified approach. Cengage Learning, 2012.

Prakasa Rao, B. L.: Nonparametric functional estimation. 1983.

Shao, Y., Cook, R. D., and Weisberg, S.: Marginal tests with sliced average variance estimation. Biometrika, 94(2):285–296, 2007.

Shin, S. J., Wu, Y., Zhang, H. H., and Liu, Y.: Probability-enhanced sufficient dimension reduction for binary classification. Biometrics, 70(3):546–555, 2014.

Wold, H.: Soft modelling by latent variables: the non-linear iterative partial least squares (nipals) approach. Journal of Applied Probability, 12(S1):117–142, 1975.

Xia, Y., Tong, H., Li, W. K., and Zhu, L.-X.: An adaptive estimation of dimension reduction space. In Exploration Of A Nonlinear World: An Appreciation of Howell Tong's Contributions to Statistics, pages 299–346. World Scientific, 2009.

Zhang, T. and Yang, B.: Big data dimension reduction using pca. In 2016 IEEE International Conference on Smart Cloud (SmartCloud), pages 152–157. IEEE, 2016.

VITA

NAME Xuelong Wang

EDUCATION B.S., Xi'an University of Technology, China, 2011
M.S., West Virginia University, West Virginia, 2013

PRESENTATIONS An Introduction to Parallel Computation using Foreach package
Chicago R User Group
Chicago, IL, 10/2017

Estimation of Weak Effects in Environmental Health Data (poster)
National Institutes of Health (NIH)
Cary, NC, 04/2019

Representative Approach For Big Data Dimension Reduction with Binary Responses
Joint Statistical Meetings (JSM)
Denver, CO, 07/2019

Representative Approach For Big Data Dimension Reduction with Binary Responses
Merck
North Wales, PA, 10/2019

REWARDS Student travel rewards
Chicago, IL, 2019