New Statistical Learning for Next-Generation Functional Data and Spatial Data

by

Shuoyang Wang

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

Auburn, Alabama
May 7, 2022

Keywords: Deep neural network, Empirical likelihood, Functional data analysis, Functional data classification, Geo data, Nonparametric regression

Approved by

Guanqun Cao, Chair, Associate Professor of Mathematics and Statistics
Ash Abebe, Professor of Mathematics and Statistics
Nedret Billor, Professor of Mathematics and Statistics
Peng Zeng, Associate Professor of Mathematics and Statistics
Bo Liu, Assistant Professor of Computer Science and Software Engineering

Abstract

Advancements of modern technology have enabled the collection of sophisticated, high-dimensional data sets, such as 3D images, high dimensional data and other objects living in a functional space. As such, boosting the investigation of function data, and functional data analysis (FDA) has become one of the most active fields of research in statistics during the last decades. Nevertheless, although estimations and classifications of FDA using non-parametric methods such as kernels, splines, and wavelets, are already well investigated, most of approaches still focus on 1D functional data.

With the rapid growth of modern techonology, many large-scale imaging studies have been or are being conducted to collect massive datasets with large volumes of imaging data, thus boosting the investigation of "next-generation" functional data. Beyond first-generation functional data such as random curves, it is natural to expand the concept of functional data to higher dimension and view the data as smooth surfaces, or hypersurfaces evaluated at a finite subset of some intervals in multi-dimension (e.g., some range of pixels or voxels and so on).

Deep learning allows computational models that are composed of multiple processing layers to learn from the data with multiple levels of abstraction. Many applications of deep learning use feedforward neural network architectures. For example, deep neural networks (DNNs) contain many hidden layers of neurons between the input and output layers, and have been found to exhibit superior performance across a variety of contexts. The specific structure of DNNs has turned out to be very good at discovering intricate structures in high-dimensional data. Although considerable advances have been achieved in deep learning research, from the statistical perspective its application and theoretical research is still in its infancy. There are many technical challenges left for statisticians.

In Chapter 2, we propose a DNNs based method to perform nonparametric regression for multi-dimensional functional data. This work has been published in [118]. The proposed estimators are based on sparsely connected DNNs with ReLU activation function. We provide

the convergence rate of the proposed DNNs estimator in terms of the empirical norm. We discuss how to properly select of the architecture parameters by cross-validation. Through Monte Carlo simulation studies we examine the finite-sample performance of the proposed method. Finally, the proposed method is applied to analyze positron emission tomography images of patients with Alzheimer disease obtained from the Alzheimer Disease Neuroimaging Initiative (ADNI) database.

In Chapter 3, we propose a robust estimator for the location function from multi-dimensional functional data. The proposed estimators are based on the DNNs with ReLU activation function. At the meanwhile, the estimators are less susceptible to outlying observations and model-misspecification. For any multi-dimensional functional data, we provide the uniform convergence rates for the proposed robust DNNs estimators. Simulation studies illustrate the competitive performance of the robust DNN estimators on regular data and their superior performance on data that contain anomalies. The proposed method is also applied to analyze 2D and 3D images of patients with Alzheimer's disease obtained from the ADNI database.

In Chapter 4, we exploit the optimal classification problem when data functions are Gaussian processes. Sharp nonasymptotic convergence rates for minimax excess misclassification risk are derived in both settings that data functions are fully observed and discretely observed. We explore two easily implementable classifiers based on discriminant analysis and DNN, respectively, which are both proven to achieve optimality in Gaussian setting. Our DNN classifier is new in literature which demonstrates outstanding performance even when data functions are non-Gaussian. In case of discretely observed data, we discover a novel critical sampling frequency that governs the sharp convergence rates. The proposed classifiers perform favorably in finite-sample applications, as we demonstrate through comparisons with other functional classifiers in simulations and one real data application.

In Chapter 5, we exploit the optimal functional data classification problem via DNNs in a more general framework. A sharp non-asymptotic estimation error bound on the excess misclassification risk is established which achieves the minimax rates of convergence. In contrast to existing literature, the proposed DNN classifier is proven to achieve optimality without the

knowledge of likelihood functions. This framework is further extended to accommodate general multi-dimensional functional data classification problems. We demonstrate the favorable finite sample performance of the proposed classifiers in various simulations and two real data applications, including the speech recognition data and the brain imaging data.

In Chapter 6, varying-coefficient models for spatial data distributed over two-dimensional domains are investigated and our work has been published in [120]. First, we approximate the univariate components and the geographical component in the model using univariate polynomial splines and bivariate penalized splines over triangulation, respectively. The spline estimators of the univariate and bivariate functions are consistent, and their convergence rates are also established. Second, we propose empirical likelihood-based test procedures to conduct both pointwise and simultaneous inferences for the varying-coefficient functions. We derive the asymptotic distributions of the test statistics under the null and local alternative hypotheses. The proposed methods perform favorably in finite-sample applications, as we show in simulations and an application to adult obesity prevalence data in the United States.

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my advisor Professor Guanqun Cao. I could never have reached the heights without her generous help, tremendous support and patient guidance. Her prompt inspirations, timely suggestions and infectious enthusiasm have enabled me to complete every challenging projects, including this thesis. There would never be a better advisor, mentor and friend for my life at AU.

I would like to thank my dissertation committee: Professor Ash Abebe, Professor Nedret Billor, and Professor Peng Zeng for sparing their precious time to serve on my committee and giving valuable comments and suggestions.

I would like to express my gratitude to Professor Zuofeng Shang from New Jersey Institute of Technology, for offering invaluable assistance and guidance. His profundity of knowledge and unlimited zeal have been motivating me through my graduate study. I also wish to express my gratitude to Professor Honglang Wang from Indiana University–Purdue University Indianapolis, for his voluminous knowledge and generous help. This thesis could not be completed without his immense support. My great thanks to Professor Yingru Li from University of Central Florida, for generously sharing adult obesity prevalence data, which is one of pivotal motivations for spatial data analysis in this thesis.

Besides that, I am grateful to the entire faculty and staff in the Department of Mathematics and Statistics who have taught me and assisted me during my study at AU. My special thanks go to Professor Dmitry Glotov for his interesting course and valuable support.

Thanks to the College of Sciences and Mathematics and the Department of Mathematics and Statistics who provided me with the COSAM Travel award, Emily Haynsworth Endowed Mathematics Fellowship and Baskervill Endowed Mathematics Fellowship for working on the dissertation. This dissertation is also supported in part by NSF award DMS 1736470.

Finally and most importantly, I would express my deepest gratitude to my beloved wife Wanyu Zhang, my parents Wei Wang and Pei Yang, and my grandparents Naizhi Yang and

Huanzhi Wang, for their everlasting love, endless support and unbreakable faith in me in all of my endeavors.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Functional data analysis

1.1.1 First-generation functional data

Functional data analysis (FDA) deals with the analysis of data which has function forms, and it was first introduced by [87]. Functional data are intrinsically infinite dimension, which brings challenges for both theory and computation. First-generation functional data typically consist of a random sample of independent real-valued functions $X_1(t) \ldots, X_n(t)$, on a compact interval $\mathcal{T}$ on the real line, which can be treated as a one-dimensional stochastic process. These functions are often assumed to be in a Hilbert space $L^2(\mathcal{T})$, such that $\mathbb{E}\left(\int_{\mathcal{T}} X^2(t)dt < \infty\right)$, with mean and covariance functions $\mu(t) = \mathbb{E}X(t)$ and $G(t,t') = \mathrm{Cov}(X(t), X(t'))$. Mercer's theorem implies the spectral decomposition of the symmetric and non-negative definite $G(t,t')$, such that $G(t,t') = \sum_{j=1}^{\infty} \lambda_j \psi_j(t)\psi_j(t')$, where λ_j are the eigenvalues satisfying $\lambda_1 \geq \lambda_2 \geq \ldots \geq 0$ and ψ_j are the corresponding orthogonal eigenfunctions. In FDA problems, estimation of mean functions is the fundamental first step; see [22, 88, 41] for example. Various methods exist that allow to estimate the regression function nonparametrically. [88] adopted the mixed effect models where the mean function and the eigenfunctions were represented with B-splines and the spline coefficients were estimated by the EM algorithm; [126] applied the local linear smoothers to estimate the mean and the covariance functions. [79] generalized the linear mixed model to the functional mixed model framework, with model fitting done by using a Bayesian wavelet-based approach. In [21], a polynomial spline estimator is proposed for the mean function of functional data together with a simultaneous confidence band. These

nonparametric methods apply the pre-specified basis expansion, e.g., polynomial spline, local linear smoother, wavelet and so on, to fit the unknown mean function. The convergence rates achieve either optimal nonparametric rate or parametric rate dependents on how dense of the observed points for each subject. Another popular method is functional principal component analysis (FPCA) which is an extension of multivariate principal component analysis, see [43, 127] for example.

1.1.2 Next-generation functional data

With the rapid growth of modern techonology, many large-scale imaging studies have been or are being conducted to collect massive datasets with large volumes of imaging data, thus boosting the investigation of "next-generation" functional data. Beyond first-generation functional data such as random curves, it is natural to expand the concept of functional data to higher dimension and view the data as smooth surfaces, or hypersurfaces evaluated at a finite subset of some intervals in multi-dimension (e.g., some range of pixels or voxels and so on). Without loss of generality, d-dimension functional data consist of a random sample of independent real-valued functions $X_1(\boldsymbol{t}) \ldots, X_n(\boldsymbol{t})$, on a hypercube $[0,1]^d$. In light of Mercer's decomposition, the ith subject among all samples can be decomposed as $X_i(\boldsymbol{t}) = \mu(\boldsymbol{t}) + \eta(\boldsymbol{t}) + \epsilon_i(\boldsymbol{t})$, such that $\mathbb{E}X_i(\boldsymbol{t}) = \mu(\boldsymbol{t})$, $\mathrm{Cov}\left(X(\boldsymbol{t}), X(\boldsymbol{t}')\right) = \mathrm{Cov}\left(\eta(\boldsymbol{t}), \eta(\boldsymbol{t}')\right)$, $\epsilon_i(\boldsymbol{t})$ are independent random hypersurfaces. Even though FDA has received considerable attention over the last decade, most approaches still focus on 1D functional data. There are few existing work for estimation of mean functions $\mu(\boldsymbol{t})$ when the data for each variable are viewed on multi-dimension. Recently, several attempts have been made to extend these nonparametric methods for spatial and image data. [121] used bivariate splines over triangulations to handle an irregular domain of the images that is common in brain imaging studies. The proposed spline estimators of the mean functions are shown to be consistent and asymptotically normal. However, the triangularized bivariate splines are designed for 2D functions only. Extending spline basis functions for general d-dimensional data observed on an irregular domain is very sophisticated and becomes extremely complex as d increases. [113] proposed a regularized Haar wavelet-based approach for the analysis of 3D brain image data in the framework of functional linear regression model.

For FPCA, [131] proposed a smooth FPCA for 2D functions on irregular planar domains; their approach is based on a mixed effects model that specifies the principal component functions as bivariate splines on triangulations and the principal component scores as random effects. [64] proposed a FPCA model that can handle real functions observable on a 2D manifold. [25] extended it to analyze functional/longitudinal data observed on a general d-dimensional domain. When applying FPCA, how to choose the number of eigenfunctions is an important practical issue without a satisfactory theoretical solution. Presumably, the larger the number of eigenfunctions, the more flexible the approximation would be, and hence, the closer to the true curve. However, a large number of eigenfunctions always result in a complex model which introduces difficulties to follow-up analysis.

1.1.3 Functional regression model

In FDA problems, estimation of mean functions is the fundamental first step. There are increasing needs for estimations of high dimensional functions with data in function forms. For instance, when analyzing positron emission tomography images with Alzheimer Disease Neuroimaging Initiative (ADNI), one is always interested in estimating the underlying regression function. We consider the classical functional regression model:

$$Y_{ij} = f_0(\mathbf{X}_j) + \eta(\mathbf{X}_j) + \epsilon_i(\mathbf{X}_j), i = 1, 2, \ldots, n, j = 1, 2, \ldots, N,$$

where $\mathbf{X}_j \in \mathbb{R}^d$, $f_0 : \mathbb{R}^d \to \mathbb{R}$, $E(Y_{ij}) = f_0(\mathbf{X}_j)$ and for the i-th subject, there are N observations. $\eta(\cdot)$ is a random process with mean zero and $\mathrm{Cov}(\eta(\mathbf{X}_j), \eta(\mathbf{X}_{j'})) := G(\mathbf{X}_j, \mathbf{X}_{j'})$. $\epsilon_i(\cdot)$ is a centered measurement error function.

The challenges of the problem are, on the one hand, it is well known that when the observed points come from a hypercube, that is $[0, 1]^d$, $d = 3$ for 3D imaging study, the non-parametric convergence rates are slower than the optimal non-parametric rate. On the other hand, even though some existing works such as B-splines over triangulation are able to handle 2D or 3D functional regression problem, there is no existing method which can estimate the above function in a uniform way.

In Chapter 2 and Chapter 3, under different scenarios, we propose a uniform deep neural networks (DNNs) estimator with favorable convergence rate. By borrowing the advantage from the deep learning domain, both convergence rates do not depend on the dimension d, and the proposed DNNs estimator is unified for any dimensional functional data implicating broader and more flexible applications. Moreover, instead of assuming additional or complex structure for the true mean function, we only assume it is constructed in a modular form and the modularity of the system can be fairly complex thus resolving the misspecification issue.

1.1.4 Functional data classification

Another fundamental problem in FDA is to classify a data function based on training samples. For instance, in the speech recognition data extracted from the TIMIT database, the training samples are digitized speech curves of American English speakers from different phoneme groups, and the task is to predict the phoneme of a new speech curve. Classic multivariate analysis techniques such as logistic regression or discriminant analysis are not directly applicable, since functional data are intrinsically infinite-dimensional.

Despite the impressive performances of the existing methods for functional data classification, one is often interested in knowing whether and which of these approaches are statistically optimal, and if not, how to construct an optimal functional classifier with better performances. The term "optimality" refers to minimizing the excess misclassification risk relative to the oracle Bayes rule, which provides a theoretical understanding on the nature of the problem as well as a benchmark to measure the performance of various classifiers. In other words, minimax theory helps us to understand how the "best" functional classifier looks, as well as provides a guidance to find the "best" classifiers.

Even though it has been investigated in multivariate settings, optimal classification in functional setting is more challenging due to the infinite-dimensional characteristic of the data. In Chapters 4 and 5, we explore the optimal functional data classification problems and propose the optimal classifiers for both one-dimensional and multi-dimensional functional data.

1.2 Deep neural networks

With the development of modern technology and increasing demands for big data, the use of neural networks has been one of the most promising approaches in connection with applications related to estimation of multivariate functions (see, e.g., [4, 89]) and classification of multi-dimension data (see, e.g., [52]). The corresponding techniques of multilayer neural networks are called deep learning. Recently, it has been proved by [92] and [70] that the L_2 risk of the least squares neural network regression estimator achieves the same minimax rate of convergence (up to a logarithmic factor) as proposed in [101]. Furthermore, this neural network estimator does not suffer the curse-of-dimensionality which is a classical drawback in the traditional nonparametric regression framework. At almost the same time, [52] shows that neural network classifiers under three cases achieve fast convergence rates, and the convergence rate in smooth conditional class probability case is minimax optimal (up to a logarithmic factor) as proposed in [7]. Although considerable advances have been achieved in deep learning research, from the statistical perspective its application and theoretical research is still in its infancy stage. There are many technical challenges left for statisticians. For example, the availability of scalable computing and stochastic optimization techniques are challenge for developing statistical asymptotic properties.

From Chapters 2 to 5, we propose novel DNNs-based methodologies to investigate a large scale of FDA problems, including functional regressions and functional classifications.

1.3 Complex spatial data

The unequal food retail environment (FRE) has been recognized as a critical contextual factor contributing to geographic disparities in obesity. However, there is no clear conclusion on the relationship between the FRE and obesity, owing to diverse measures of the FRE and socioeconomic disparities. In order to resolve this challenge, multiple types of food stores, restaurants, and Supplemental Nutrition Assistance Program (SNAP) stores are considered to assess the FRE from two important perspectives: X_1, availability, and X_2, healthfulness. In particular, X_1 is a composite index of the densities of food stores, restaurants, and SNAP stores, and X_2

is a composite index of the ratios of healthy to unhealthy food stores, full service restaurants to fast food restaurants, and healthy to unhealthy SNAP stores. Data are collected from $3,091$ counties in the United States in 2018. For each county, $\boldsymbol{S}_i = (S_{i1}, S_{i2})$ is their geographical location, and Z_i is their median household income.

Based on this data set, socioeconomists attempt to disentangle how county-level associations between the food environment and obesity rates change with median household income levels. This leads to modeling the effect of food retail environments as functions of household income levels. However, owing to the geographic dependence, the classical varying coefficient model (VCM) is not sufficient. To address this issue, we propose the varying-coefficient geo model (VCGM), and model the county-level obesity rate (Y) as the following:

$$Y_i = \beta_0(Z_i) + X_{i1}\beta_1(Z_i) + X_{i2}\beta_2(Z_i) + \alpha(\boldsymbol{S}_i) + \epsilon_i, i = 1, \cdots, 3,091.$$

In Chapter 6, we apply bivariate splines over triangulations [56] for estimating $\alpha(\cdot)$, because they can handle irregular 2D domains with complex boundaries and they are computationally efficient. In addition, we propose both pointwise (at a specific z) and simultaneous (for all $z \in [a, b]$) testing procedures for the following hypothesis:

$$H_0 : H\{\boldsymbol{\beta}_0(z)\} = 0 \text{ v.s. } H_1 : H\{\boldsymbol{\beta}_0(z)\} \neq 0,$$

where $H(\mathbf{b})$ is a q-dimensional function of $\mathbf{b} = (b_1, \ldots, b_p) \in \mathbb{R}^p$, such that $\mathbf{C}(\mathbf{b}) := \partial H(\mathbf{b})/\partial \mathbf{b}^\top$ is a $q \times p$ full-rank matrix ($q \leq p$), for all $\mathbf{b}$. The above hypothesis is very general, owing to the choice flexibility of $H(\mathbf{b})$. It includes many interesting hypotheses as special cases, for instance, $H_0 : \beta_{0,k}(z) = 0$ for all k if $H(\mathbf{b}) = \mathbf{b}$, a test for any arbitrary linear constraints on $\boldsymbol{\beta}_0$ if $H(\mathbf{b}) = \boldsymbol{\Lambda}\mathbf{b} - \mathbf{c}_0$ for a $q \times p$ known matrix $\boldsymbol{\Lambda}$ and a known vector $\mathbf{c}_0$, and even tests with nonlinear constraints. See [6] for explicit examples of nonlinear hypotheses. Both tests are based on the empirical likelihood (EL), which is a nonparametric likelihood, introduced by [82, 83]. In spite of its nonparametric construction based on observed data points,

the EL shares some convenient merits of the parametric likelihood, and has many desirable advantages in deriving confidence sets for unknown parameters.

Chapter 2

Estimation of the Mean Function of Functional Data via Deep Neural Networks

2.1 Introduction

Functional data refer to curves or functions, i.e. the data for each variable are viewed as smooth curves, surfaces, or hypersurfaces evaluated at a finite subset of some interval in 1D and 2D (e.g., some period of time, some range of pixels or voxels and so on). Functional data means intrinsically infinite-dimensional but are usually measured discretely. The high intrinsic dimensionality of these data poses challenges both for theory and computation. Functional data analysis (FDA) has been a topic of increasing interest in the statistics community for recent decades. [87] and [115] gave a comprehensive overview of FDA.

In FDA problems, estimation of mean functions is the fundamental first step; see [22, 88, 41] for example. Various methods exist that allow to estimate the regression function nonparametrically. [88] adopted the mixed effect models where the mean function and the eigenfunctions were represented with B-splines and the spline coefficients were estimated by the EM algorithm; [126] applied the local linear smoothers to estimate the mean and the covariance functions. [79] generalized the linear mixed model to the functional mixed model framework, with model fitting done by using a Bayesian wavelet-based approach. In [21], a polynomial spline estimator is proposed for the mean function of functional data together with a simultaneous confidence band. These nonparametric methods apply the pre-specified basis expansion, e.g., polynomial spline, local linear smoother, wavelet and so on, to fit the unknown mean function. The convergence rates achieve either optimal nonparametric rate or parametric rate dependents on how dense of the observed points for each subject.

Even though FDA has received considerable attention over the last decade, most approaches still focus on 1D functional data. The high intrinsic dimensionality of these data poses challenges both for theory and computation; these challenges vary with how the functional data were sampled. Hence, few are developed for general multi-dimensional functional data. Recently, several attempts have been made to extend these nonparametric methods for spatial and image data. [121] used bivariate splines over triangulations to handle an irregular domain of the images that is common in brain imaging studies. The proposed spline estimators of the mean functions are shown to be consistent and asymptotically normal. However, the triangularized bivariate splines are designed for 2D functions only. Extending spline basis functions for general d-dimensional data observed on an irregular domain is very sophisticated and becomes extremely complex as d increases. [113] proposed a regularized Haar wavelet-based approach for the analysis of 3D brain image data in the framework of functional linear regression model.

Another popular method is functional principal component analysis (FPCA) which is an extension of multivariate principal component analysis, see [43, 127] for example. Recently, there are a few studies on 2D FDA. [131] proposed a smooth FPCA for 2D functions on irregular planar domains; their approach is based on a mixed effects model that specifies the principal component functions as bivariate splines on triangulations and the principal component scores as random effects. [64] proposed a FPCA model that can handle real functions observable on a 2D manifold. [25] extended it to analyze functional/longitudinal data observed on a general d-dimensional domain. When applying FPCA, how to choose the number of eigenfunctions is an important practical issue without a satisfactory theoretical solution. Presumably, the larger the number of eigenfunctions, the more flexible the approximation would be, and hence, the closer to the true curve. However, a large number of eigenfunctions always result in a complex model which introduces difficulties to follow-up analysis.

For many years, the use of neural networks has been one of the most promising approaches in connection with applications related to approximation and estimation of multivariate functions (see, e.g., [4, 89]). Recently, the focus is on multilayer neural networks, which use many

hidden layers, and the corresponding techniques are called *deep learning*. Under the nonparametric regression model, via sparsely connected deep neural networks, [92] and [70] showed that the L_2 risk of the least squares neural network regression estimator achieves the same minimax rate of convergence (up to a logarithmic factor) as proposed in [101]. Furthermore, this neural network estimator does not suffer the curse-of-dimensionality which is a classical drawback in the traditional nonparametric regression framework. [11] has also obtained the similar results under deep learning framework via a different activation function. [69] further removed the logarithmic factors to achieve exact optimal nonparametric rate.

Although considerable advances have been achieved in deep learning research, from the statistical perspective its application and theoretical research is still in its infancy stage [35]. There are many technical challenges left for statisticians. For example, the availability of scalable computing and stochastic optimization techniques are challenge for developing statistical asymptotic properties. Recently, there are some works proposed for deep learning algorithms from statistical point of view [104, 105]. Motivated by these desiderata, the main goal of this article is to provide a novel method of FDA in the neural network framework.

The contributions of this work are three-fold. First, to our best knowledge, this is the first work on proposing deep neural networks (DNN) based estimator for FDA. An R package "`FDADNN`" has been developed and is available from GitHub website. Second, we develop the convergence rate (in empirical norm) of the proposed neural networks estimator. It is well-known that when the observed points come from a hypercube, i.e., $[0,1]^d$, $d = 3$ for 3D imaging study, the nonparametric convergence rates are slower than the optimal nonparametric rate. This means that no statistical procedure can perfectly recover the signal pointwisely. However, by borrowing the advantage from the *deep learning* domain, the convergence rate of the proposed DNN estimator does not depend on the dimension d. Finally, we do not assume additional or complex structure for the true mean function, for example, additive models or single-index models. As in the deep learning domain, the true regression functions are assumed to be constructed in a modular form and the modularity of the system can be fairly complex, which resolve the misspecification issue.

Different from the existing neural network literature on nonparametric regression [11], [92] and [70] which only handle i.i.d. data, we focus on FDA, where each subject is a random curve in a hypercube. Because of this special data structure, the major challenge becomes to deal with the correlation among the N evaluation points in the framework of neural network, which has not been achieved in the existing works. It is not surprising that the convergence rate decreases with n (number of subjects) as well as N.

This chapter is structured as follows. Section 2.2 provides the model setting in FDA and introduces multilayer feed-forward artificial neural networks and discusses mathematical modeling. The implementation on hyperparameter selections also be included in Section 2.2. The theoretical properties of the proposed DNN estimator can be found in Section 2.4. In Section 2.5, it is shown that the finite sample performance of proposed neural network estimator. The proposed method is applied to the spatially normalized positron emission tomography (PET) data from Alzheimer Disease Neuroimaging Initiative (ADNI) in Section 2.6 and make some concluding remarks in Section 2.7. The proof of the main result together with additional discussion can be found in Appendix.

2.2 The model and the deep neural network estimator

2.2.1 FDA model

Denote by Y_{ij} the j-th observation of the random curve $\xi_i(\cdot)$ at grid points $\mathbf{X}_{ij}$, $1 \leq i \leq n, 1 \leq j \leq N_i$. For simple notations, we examine the equally spaced design, in other words, $\mathbf{X}_{ij} = \mathbf{X}_j = j/N$. The main results can be extended to irregularly spaced design. Without loss of generality, let $\mathbf{X}_j = (X_{j1}, \ldots, X_{jd}) \in [0,1]^d$. For the i-th subject, its sample path $\{\mathbf{X}_j, Y_{ij}\}$ consists of the noisy realization of the Gaussian process $\xi_i(\mathbf{X})$ in the sense that $Y_{ij} = \xi_i(\mathbf{X}_j) + \epsilon_i(\mathbf{X}_j)$, and $\left\{\xi_i(\mathbf{X}), \mathbf{X} \in [0,1]^d\right\}$ are i.i.d. copies of the process $\left\{\xi(\mathbf{X}), \mathbf{X} \in [0,1]^d\right\}$ which is L^2, i.e., $E\int_{[0,1]^d} \xi^2(\mathbf{X}) d\mathbf{X} < +\infty$. The error term $\epsilon_i(\mathbf{X}_j)$ has mean zero and finite variance.

In this work, we consider the following classical FDA model:

$$\begin{aligned} Y_{ij} &= \xi_i(\mathbf{X}_j) + \epsilon_i(\mathbf{X}_j) \\ &= f_0(\mathbf{X}_j) + \eta(\mathbf{X}_j) + \epsilon_i(\mathbf{X}_j), \quad i = 1, 2, \ldots, n, j = 1, 2, \ldots, N, \end{aligned} \tag{2.1}$$

where $f_0 : \mathbb{R}^d \to \mathbb{R}$, $E(Y_{ij}) = f_0(\mathbf{X}_j)$, $\eta(\cdot)$ is a Gaussian process characterizing individual curve variations from $f_0(\cdot)$ with mean zero and $\mathrm{Cov}(\eta(\mathbf{X}_j), \eta(\mathbf{X}_{j'})) := G(\mathbf{X}_j, \mathbf{X}_{j'})$. Let $\epsilon_i(\mathbf{X}_j) = \tau(\mathbf{X}_j)\varepsilon_{ij}$, where ε_{ij}'s are independent normal random variables and $\tau(\mathbf{X})$ is the standard deviation function bounded above zero for any $\mathbf{X} \in [0,1]^d$. By Mercer's Theorem, covariance function $G(\mathbf{X}, \mathbf{X}')$ has the following spectrum decomposition

$$G(\mathbf{X}, \mathbf{X}') = \sum_{k=1}^{\infty} \lambda_k \psi_k(\mathbf{X}) \psi_k(\mathbf{X}'),$$

where $\{\lambda_k\}_{k=1}^{\infty}$ and $\{\psi_k(\mathbf{X})\}_{k=1}^{\infty}$ are the eigenvalues and eigenfunctions of $G(\mathbf{X}, \mathbf{X}')$, and $\{\psi_k(\mathbf{X})\}_{k=1}^{\infty}$ are orthonormal bases in $L_2([0,1]^d)$.

2.2.2 Deep Neural Networks

Before conducting the estimation of the mean function f_0 in (2.1) via the DNN, let us briefly introduce the necessary notations and terminologies used in the neural networks. From the high level, typical DNN use a composition of a series of simple nonlinear functions to model nonlinearity, i.e.,

$$\mathbf{h}_L = \mathbf{g}_L \circ \mathbf{g}_{L-1} \circ \ldots \circ \mathbf{g}_1(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^d, \tag{2.2}$$

where $\circ$ denotes composition of two functions and L is called the number of hidden layers or depth of a DNN model. One can define $\mathbf{h}_l = \mathbf{g}_l(\mathbf{h}_{l-1})$ for each $1 \le l \le L$ recursively and $\mathbf{h}_0 = \mathbf{x}$. "Deep" in deep neural networks refers to the use of multiple layers in the network. In the feed-forward neural network, the information moves in only one direction-forward-from the input layers, through the hidden layers and to the output nodes layers. In this kind of neural nets, there is a specific choice of $\mathbf{g}_l$: $\mathbf{g}_l(\mathbf{h}_{l-1}) = \boldsymbol{\sigma}(\mathbf{W}_l \mathbf{h}_{l-1})$, $l = 1, \ldots, L$, where $\mathbf{W}_l$ is a $p_l \times p_{l+1}$ weight matrix in the l-th layer and $\mathbf{p} = (p_0, \ldots, p_{L+1})$ is the width vector. The

nonlinear function $\boldsymbol{\sigma}$ is called the activation function. Here, we study the popular rectifier linear unit (ReLU) activation function applied element-wise $[\boldsymbol{\sigma}(\mathbf{x})]_j = \max(x_j, 0)$, $j = 1, \ldots, d$. For any vector $\mathbf{v} = (v_1, \ldots, v_d) \in \mathbb{R}^d$, define the shifted activation function $\sigma_{\mathbf{v}} : \mathbb{R}^d \to \mathbb{R}^d$ as: $[\sigma_{\mathbf{v}}(\mathbf{x})]_j = \sigma(x_j - v_j)$, $j = 1, \ldots, d$. We call $\mathbf{v}$ the activation vector. The DNN model is then any function of the form is defined as $f : \mathbb{R}^{p_0} \to \mathbb{R}^{p_{L+1}}$,

$$f(\mathbf{x}) = \mathbf{W}_L \sigma_{\boldsymbol{V}_L} W_{L-1} \sigma_{\boldsymbol{V}_{L-1}} \ldots \mathbf{W}_1 \sigma_{\boldsymbol{V}_1} \mathbf{W}_0 \mathbf{x}. \tag{2.3}$$

To fit networks with data generated from the d-dimensional hypercube functional data model, we must have $p_0 = d$ and $p_{L+1} = 1$. Without loss of generality, we assume for any $f \in \mathcal{F}$, its empirical norm is bounded, i.e., $\|f\|_N = \left(\frac{1}{N}\sum_{j=1}^N f^2(\mathbf{x}_j)\right)^{1/2} \leq C_f < \infty$.

Although the depth and width of the neural nets can be extremely deep and wide, overfitting and computational burden are serious problems in such networks. To overcome these issues, the networks are modeled by assuming that each unit will be active only for a small fraction of the data to avoid overfitting. Smaller weights in a neural network can result in a model that is more stable and less likely to overfit the training dataset, in turn having better performance when making a prediction on new data [99]. Therefore, we assume that there are only few non-zero network parameters. Equivalently, we define the sparse neural networks and add constrains on the maximum-entry norm and non-zero entries of weight matrix $\mathbf{W}_l$ and activation vector $\mathbf{v}_l$. The sparse neural networks for our functional data model are given by

$$\begin{aligned} &\mathcal{F}(L, \mathbf{p}, s) \\ = &\left\{ f(\cdot) \text{ of the form (2.3)} : \max_{l=0,\ldots,L} \|\mathbf{W}_l\|_\infty + \|\mathbf{v}_l\|_\infty \leq 1, \sum_{l=0}^{L} \|\mathbf{W}_l\|_0 + \|\mathbf{v}_l\|_0 \leq s \right\}, \end{aligned} \tag{2.4}$$

where $s > 0$, $\|\cdot\|_\infty$ denotes the maximum-entry norm and $\|\cdot\|_0$ denotes the number of non-zero entries, respectively. Let $\mathbf{v}_0$ be a zero vector for simply notation. The selecting procedures of unknown tuning parameters $(L, \mathbf{p}, s)$ shall be given in the Section 2.3.

2.2.3 Deep neural network estimator

In the functional data regression model, the common objective is to find an optimal estimator by least-square loss function. In the neural network setting, this coincides with training neural networks by minimizing the empirical risk over all the training data. In particularly, given the networks in (2.4) and denote $\mathcal{F} = \mathcal{F}(L, \mathbf{p}, s)$, the proposed DNN estimator is defined as

$$\widehat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{j=1}^{N} \left\{ \overline{Y}_{\cdot j} - f\left(\mathbf{X}_j\right) \right\}^2, \tag{2.5}$$

where $\overline{Y}_{\cdot j} = \frac{1}{n} \sum_{j=1}^{n} Y_{ij}$. Different from classical nonparametric estimators, $\widehat{f}$ has no analytical expression or basis expansion expression. Hence, to better understand the reasons that this DNN estimator has excellent performance, we first project f_0 onto the network space $\mathcal{F}$, namely, $f^* := \arg\min_{f \in \mathcal{F}} \|f_0 - f\|_\infty$. In other words, f^* is the best possible approximation of f_0 in $\mathcal{F}$. Note that

$$\frac{1}{N} \sum_{j=1}^{N} \left(\overline{Y}_{\cdot j} - \widehat{f}(\mathbf{X}_j) \right)^2 \leq \frac{1}{N} \sum_{j=1}^{N} \left(\overline{Y}_{\cdot j} - f^*(\mathbf{X}_j) \right)^2,$$

which is equivalent to

$$\frac{1}{N} \sum_{j=1}^{N} \left(f_0(\mathbf{X}_j) - \widehat{f}(\mathbf{X}_j) + \overline{\rho}_{\cdot j} \right)^2 \leq \frac{1}{N} \sum_{j=1}^{N} \left(f_0(\mathbf{X}_j) - f^*(\mathbf{X}_j) + \overline{\rho}_{\cdot j} \right)^2,$$

where $\overline{\rho}_{\cdot j} = \frac{1}{n} \sum_{i=1}^{n} \rho_{ij} = \frac{1}{n} \sum_{i=1}^{n} \eta_i\left(\mathbf{X}_j\right) + \frac{1}{n} \sum_{i=1}^{n} \epsilon_i\left(\mathbf{X}_j\right)$. Hence, we follow the conventional approximation-estimation decomposition (or bias-variance tradeoff) to decompose the empirical norm $\|\widehat{f} - f_0\|_N = \frac{1}{N} \sum_{j=1}^{N} \left(\widehat{f}(\mathbf{X}_j) - f_0(\mathbf{X}_j) \right)^2$ as

$$\|\widehat{f} - f_0\|_N \leq \underbrace{\frac{1}{N} \sum_{j=1}^{N} \left(f^*(\mathbf{X}_j) - f_0(\mathbf{X}_j) \right)^2}_{\text{approximation error}} + \underbrace{\frac{2}{N} \sum_{j=1}^{N} \left(\widehat{f}(\mathbf{X}_j) - f^*(\mathbf{X}_j) \right) \overline{\rho}_{\cdot j}}_{\text{estimation error}}. \tag{2.6}$$

The above equation indicates that the empirical norm of the estimator is bounded by two items. The first item is the approximation error and essentially determined by the distance between the

network class $\mathcal{F}$ and true function class f_0, which can be arbitrarily small according to [128]. From statistical point of view, the second item is the estimation error and is a weighted average of a random process. It is affected by the parameters in $\mathcal{F}$, true mean function class, and the characteristic of the error terms.

2.3 Implementation

In this section, we discuss the detailed computational procedure for the proposed DNN estimator in (2.5). The following proposed computational procedure can be easily realized via R package "`FDADNN`" which is available at `https://github.com/FDASTATAUBURN/FDADNN`.

2.3.1 Neural network's architecture selection

Tuning parameters are crucial as they control the overall behavior of the proposed estimator and the learning process. In machine learning, those parameters are called network architecture parameters. A neural network's architecture can simply be defined as the number of layers L, and the number of hidden neurons within these layers $\mathbf{p}$. In our considered sparse neural network space $\mathcal{F}$, sparse parameter s should also be carefully selected. Note that in the practice, it's unrealistic to control the exact number of inactive nodes, so instead of using sparse parameter s, we add an L_1 penalty to control the number of active nodes in each layer during optimization procedure. Denote ζ as the L_1 regularization factor. In the following, we utilize ζ to replace sparse parameter s in the numerical analysis. The ultimate goal is to find an optimal combination of $(L, \mathbf{p}, \zeta)$ that minimizes a pre-defined loss function to give better results. There are fairly large numbers of literature discussing the optimization selection, such as grid search, random search, and Bayesian optimization. Considering the computational efficiency and statistical properties, we recommend the following data-adaptive selection procedure in the practical application. The further justification of the optimization algorithm is beyond the scope of this work and shall be anther interesting and challenge topic for the future work.

We set the same neuron numbers for each layer for simplicity, i.e. $\mathbf{p} = (p, \ldots, p)$, and we follow the rule that p is increasing as n and N are increasing. We use K-fold cross-validation

to choose (L, p, ζ), i.e.,

$$(L_{opt}, p_{opt}, \zeta_{opt}) = \arg \min_{(L,p,s)\in\Theta} \sum_{k=1}^{K} \sum_{j=1}^{N} \left(\widehat{\overline{Y}}_{\cdot j}^{(-k)} (L, p, \zeta) - \overline{Y}_{\cdot j}^{(k)} \right)^2 ,$$

where Θ is a architecture parameter space which contains pre-selected choices of (L, p, ζ). Typically, $K = 5$ or 10. For the k-th cross-validation, at the j-th grid point, $\widehat{\overline{Y}}_{\cdot j}^{(-k)} (L, p, \zeta)$ denotes the estimated output given (L, p, ζ) and $\overline{Y}_{\cdot j}^{(k)}$ is the average of observations.

2.3.2 Training neural networks

The minimization in (2.5) is usually done via stochastic gradient descent (SDG). In a way similar to gradient descent, in each update, a small sub-sample called a *batch* which is typically of size $B = 32$ to 512, is randomly drawn and the gradient calculation is only on the sub-sample instead of the whole training dataset. This saves considerably the computational cost in calculation of gradient. By the law of large numbers, this stochastic gradient should be close to the full sample one, albeit with some random fluctuations. We choose $B = 32$ or 64 batches depending on the performance of convergence. A pass of the whole training set is called an *epoch*. Typical choices of epochs are 200, 300 and 500. The number of epochs which defines the number times that the learning algorithm works through the entire training data set.

There are certainly some challenges for SGD to train DNN. For example, albeit good theoretical guarantees for well-behaved problems, SGD might converge very slowly; the learning rates are difficult to tune [2]. To address these challenges, several variants gradient-based optimization algorithms are introduced, such as `Adam`, `RMSprop` and `Adadelta`. Instead of the classical SGD procedure, `Adam` is a method for efficient stochastic optimization that only requires first-order gradients with little memory requirement. Hence, it is well suited for problems when there are large sample size and parameters [53]. In our numerical studies, `Adam` provides the best results and is the most computationally efficient among these candidates. We recommend `Adam` in the real life applications for FDA.

2.4 Theoretical properties of the DNN estimator

In this section, we develop the convergence rate of the proposed DNN estimator in (2.5). For simple notations, $\log$ means the logarithmic function with base 2. For sequences $(a_n)_n$ and $(b_n)_n$, $a_n \asymp b_n$ means $a_n \leq c_1 b_n$ and $a_n \geq c_2 b_n$ where c_1 and c_2 are absolute constants for any n. Let $\mathbf{C}_N = [G(\mathbf{X}_j, \mathbf{X}_{j'})/N]_{j,j'=1}^N$ be the $N \times N$ kernel matrix corresponding to covariance function G. We now introduce the main assumptions:

(A1) The true regression function $f_0 \in \mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K})$. (The definition of $\mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K})$ is given in the Appendix.)

(A2) The standard deviation function $\tau(\cdot)$ is bounded for any $\mathbf{x} \in [0,1]^d$.

(A3) The eigenvalues of $G(\cdot,\cdot)$ satisfy $\lambda_1 \geq \lambda_2 \geq \ldots \geq 0$ and $\sum_{k=1}^{\infty} \lambda_k < \infty$. Moreover, the maximal eigenvalue of the kernel matrix $\mathbf{C}_N$ satisfies $\lambda_{1,N} = O(N^{-\varrho})$ for some constant $\varrho \geq 0$.

(A4) The DNN estimator $\widehat{f} \in \mathcal{F}(L, \mathbf{p}, s)$, where $L \asymp \log(nN^{\varrho})$, $s \asymp (nN^{\varrho})^{\frac{1}{\theta+1}}$, $\min_{l=1,\ldots,L} p_l \asymp (nN^{\varrho})^{\frac{1}{\theta+1}}$, for $\theta = \min_{i=0,\ldots,q} \frac{2\beta_i^*}{t_i}$.

Assumption (A1) is a natural definition for neural network, which is fairly flexible and many well known function classes are contained in it. For example, the additive model $f_0(\mathbf{x}) = \sum_{i=1}^d f_i(x_i)$, can be written as a composition of two functions $f_0 = g_1 \circ g_0$, with $g_0(\mathbf{x}) = (f_1(x_1), \ldots, f_d(x_d))^\top$ and $g_1(\mathbf{x}) = \sum_{j=1}^d x_j$, such that $g_0 : [0,1]^d \to \mathbb{R}^d$ and $g_1 : \mathbb{R}^d \to \mathbb{R}$. Here $\mathbf{d} = (d, d, 1)$ and $\mathbf{t} = (1, d)$. The generalized additive model $f_0(\mathbf{x}) = h\left(\sum_{i=1}^d f_i(x_i)\right)$, it can be written as a composition of three functions $f_0 = g_2 \circ g_1 \circ g_0$, with g_0, g_1 described above, and $g_2 = h$.

Assumption (A2) is a standard assumption for the variance of measurement errors, which requires the bounded variance of measurement error over the whole space. This assumption has been widely used in functional data nonparametric regression literature, see [21, 126] for example. Assumption (A3) is a standard eigenvalue assumption for Mercer kernel and it is widely used assumption for covariance functions in FDA literature, see [20, 63] for example.

We also provide two examples to demonstrate (A3) is a reasonable assumption in the supplementary file. By [16], Assumption (A3) trivially holds for $\varrho = 0$ (see Proposition A.1 and A.2), and may even hold for some positive ϱ as revealed by Examples 1 in Section A.1. Assumption (A4) depicts the architecture and parameters' setting in the network space.

We assume a natural compositional function class for the true mean function f_0:

$$f_0 = g_q \circ g_{q-1} \circ \ldots \circ g_1 \circ g_0,$$

where $g_i : [a_i, b_i]^{d_i} \to [a_{i+1}, b_{i+1}]^{d_{i+1}}$, $g_i = (g_{ij})_{j=1,\ldots,d_{i+1}}^{\top}$, $i = 1, \ldots, q$, with unknown parameters d_i and q.

The following theorem establishes the convergence rate of the DNN estimator $\widehat{f}$ under the empirical norm. Its proof and some technical lemmas will be provided in the supplementary file.

Theorem 2.1. *Under Assumptions (A1)-(A4), with probability greater than* $(1-\frac{2}{nN^{\varrho}})^{\lceil \log(nN^{\varrho})\rceil+1} \to 1$, *we have*

$$\|\widehat{f} - f_0\|_N^2 \leq c(nN^{\varrho})^{-\frac{\theta}{\theta+1}} \log^6(nN^{\varrho}), \tag{2.7}$$

where $\varrho \geq 0$, $\theta = \min_{i=0,\ldots,q} \frac{2\beta_i^*}{t_i}$, c *is a constant only depends on* $\mathbf{t}$, $\mathbf{d}$, $\boldsymbol{\beta}$ *which are defined in (A1) in the Appendix.*

2.5 Simulation

To illustrate how the introduced nonparametric regression estimators based on our proposed neural networks method behave in case of finite sample sizes, we conduct substantial simulations for both 2D and 3D functional data.

2.5.1 2D simulation

In this simulation, the 2D images are generated from the model:

$$Y_{ij} = f_0(\mathbf{X}_j) + \eta(\mathbf{X}_j) + \epsilon_i(\mathbf{X}_j), \tag{2.8}$$

where $\mathbf{X}_j = (X_{1j}, X_{2j}) = (j_1/N_2, j_2/N_2)$, $1 \leq j_1, j_2 \leq N_2$ are equally spaced grid points on the $[0,1]^2$ and $N_2^2 = N$. To demonstrate the practical performance of our theoretical results, we consider the following two mean functions:

- Case 1 : $f_0(x_{1j}, x_{2j}) = \frac{-8}{1+\exp\left(\cot(x_{1j}^2)\cos(2\pi x_{2j})\right)}$,
- Case 2 : $f_0(x_{1j}, x_{2j}) = \log\left(\sin(2\pi x_{1j}) + 2|\tan(2\pi x_{2j})| + 2\right)$,

and the corresponding images are shown in the first row of Figure 2.1. To simulate the within-subject dependence for each subject i, we generate $\eta_i(\cdot)$ from a Gaussian process, with mean 0, and covariance function $G_0(\mathbf{x}_j, \mathbf{x}_{j'}) = \sum_{k=1}^{2} \cos\left(2\pi(x_{kj} - x_{kj'})\right)$, $j, j' = 1, \ldots, N$. We generate $\epsilon_i(\mathbf{x}_j) = \varepsilon_{ij} \sim^{\text{i.i.d.}} \mathcal{N}(0, \sigma^2)$ for $i = 1, \ldots, n$, $j = 1, \ldots, N$. The noise level is set to be $\sigma = 1, 2$. We consider sample size $n = 50, 100, 200$ and for each image, let $N_2 = 15$ or 25, which means for each 2D image, the number of observational points (pixels) is set to be $N = N_2^2 = 225$ or 625. The neural network (2.5) is trained through optimizer `Adam` with architecture parameters (L, p, ζ) selected from $L \in \{3, 4\}$, $p \in \{100, 300, 500, 1000, 2000\}$, $\zeta \in \{10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}\}$. 10-fold cross-validation method discussed in Section 2.3 is applied to select the optimal architecture parameters in each Monte Calro simulation. Epochs are selected from 300 to 500 and batch size is chosen as 32. We find the convergence of algorithms is promising.

The alternative approach for 2D case we considered is a 2D regression spline method (bivariate spline). With regard to the variety of modifications of this approach known in the literature, we focus on the version for 2D FDA in [57]. Let $\mathbf{B}^\top(\mathbf{x}) = \{B_m(\mathbf{x})\}_{m \in \mathcal{M}}$ be the set of bivariate Bernstein basis polynomials, where $\mathcal{M}$ stands for an index set of Bernstein basis polynomials. Then we can represent any bivariate function $f(\mathbf{x})$ by $f(\mathbf{x}) \approx \mathbf{B}^\top(\mathbf{x})\boldsymbol{\gamma}$ where $\boldsymbol{\gamma}^\top = (\gamma_m, m \in \mathcal{M})$ is the bivariate spline coefficient vector. The estimator $\widehat{f}_{BS}$ is implemented by the R package `BPST`, which was developed by the authors of [57].

The second and the third rows in Figure 2.1 depicts the proposed neural network estimator $\widehat{f}_{DNN}$ and bivariate spline estimator $\widehat{f}_{BS}$ when $n = 200$, $N = 625$ and $\sigma = 1$. Table 2.1 summarizes the empirical L_2 risk and standard deviation of estimators $\widehat{f}_{DNN}$ and $\widehat{f}_{BS}$ under 100 simulations for two different noise levels. From the above figures and table, one can see

that our method and the bivariate spline method have fairly similar estimation performances. As the bivariate spline estimator is able to achieve the optimal nonparametric convergence rate [121], the comparable estimation results in Tables 2.1 and 2.2 also support the asymptotic convergence rate of our proposed estimator $\widehat{f}_{DNN}$ in Theorem 2.1.

Table 2.1: The average empirical L_2 risk and their standard deviations of f_0 (Case 1) across 100 simulation runs (2D case).

$f_0(x_{1j}, x_{2j}) = \frac{-8}{1+\exp\left(\cot(x_{1j}^2)\cos(2\pi x_{2j})\right)}$						
σ	N	n	DNN		bivariate spline	
			L_2 risk	SD	L_2 risk	SD
1	225	50	0.1327	0.1905	0.6030	0.0418
		100	0.0797	0.1244	0.5757	0.0249
		200	0.0432	0.0574	0.5584	0.0120
	625	50	0.0770	0.0497	0.1497	0.0462
		100	0.0535	0.0368	0.1136	0.0214
		200	0.0352	0.0295	0.0987	0.0098
2	225	50	0.1880	0.1521	0.6564	0.1009
		100	0.0918	0.0793	0.6035	0.0619
		200	0.0593	0.0529	0.5765	0.0316
	625	50	0.1594	0.1555	0.2241	0.1218
		100	0.0862	0.0755	0.1430	0.0557
		200	0.0420	0.0412	0.1098	0.0232

Table 2.2: The average empirical L_2 risk and their standard deviations of f_0 (Case 2) across 100 simulation runs (2D case).

$f_0(x_{1j}, x_{2j}) = \log\left(\sin(2\pi x_{1j}) + 2\lvert\tan(2\pi x_{2j})\rvert + 2\right)$						
σ	N	n	DNN		bivariate spline	
			L_2 risk	SD	L_2 risk	SD
1	225	50	0.0731	0.0446	0.0804	0.0382
		100	0.0437	0.0249	0.0517	0.0186
		200	0.0254	0.0217	0.0351	0.0100
	625	50	0.0560	0.0206	0.0751	0.0351
		100	0.0351	0.0128	0.0541	0.0254
		200	0.0245	0.0085	0.0383	0.0110
2	225	50	0.1190	0.0975	0.1290	0.0950
		100	0.0829	0.0681	0.0931	0.0597
		200	0.0348	0.0276	0.0464	0.0251
	625	50	0.0573	0.0264	0.1213	0.0859
		100	0.0331	0.0132	0.0827	0.0630
		200	0.0139	0.0059	0.0502	0.0251

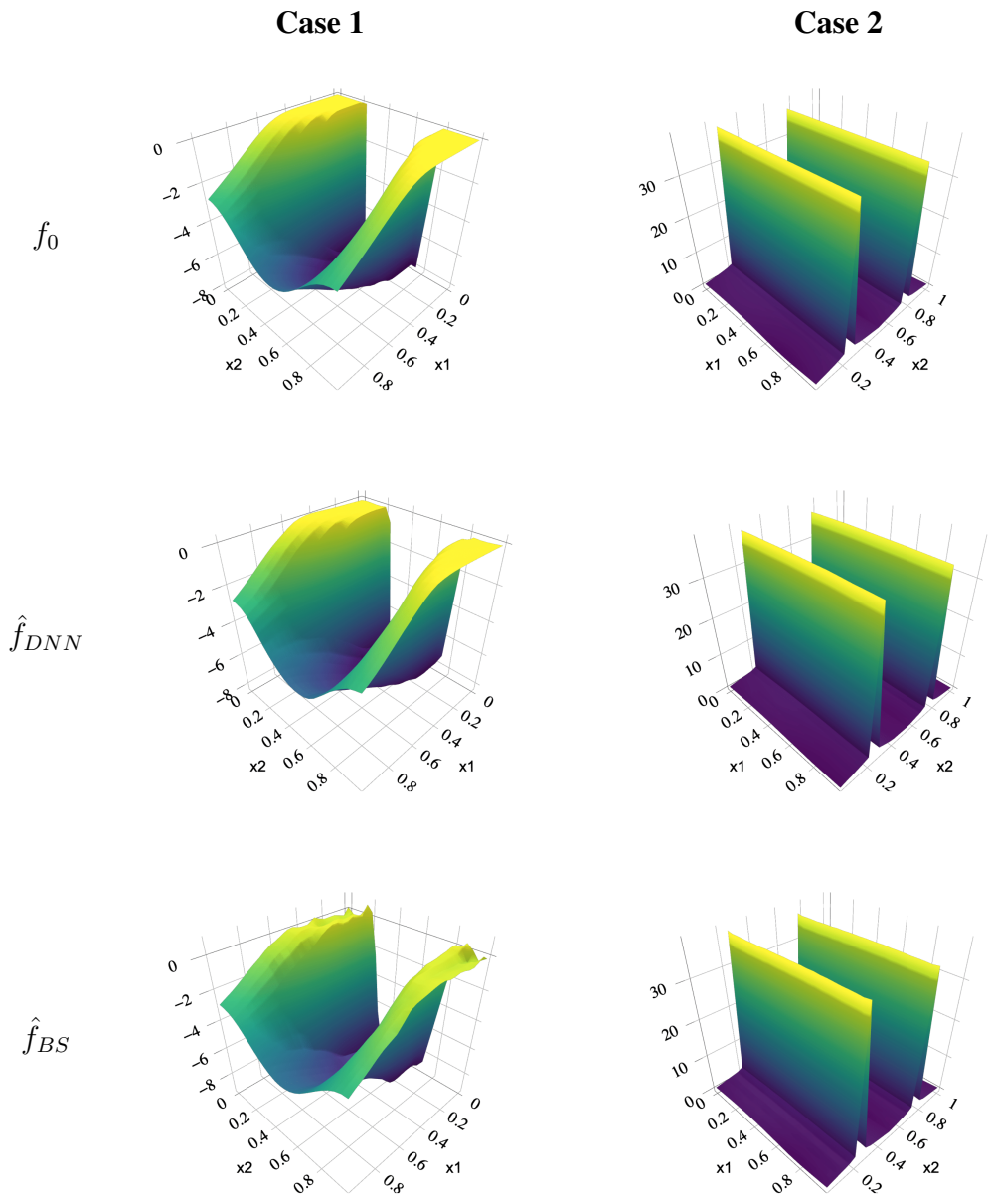

Figure 2.1: 2D simulation. Left: from the top to bottom, they are true function f_0 (Case 1) and its estimators $\widehat{f}_{DNN}$ and $\widehat{f}_{BS}$. Right: from the top to bottom, they are true function f_0 (Case 2) and its estimators $\widehat{f}_{DNN}$ and $\widehat{f}_{BS}$. ($n = 200$, $N = 625$ and $\sigma = 1$)

2.5.2 3D simulation

For 3D simulation, the images are generated from the model (2.8) in 2D case. The true mean function is $f_0(x_{1j}, x_{2j}, x_{3j}) = \exp\left(\frac{1}{3}x_{1j} + \frac{1}{3}x_{2j} + \sqrt{x_{3j} + 0.1}\right)$, where $(x_{1j}, x_{2j}, x_{3j}) = \left(\frac{j_1}{N_3}, \frac{j_2}{N_3'}, \frac{j_3}{N_3''}\right)$, $1 \leq j_1 \leq N_3$, $1 \leq j_2 \leq N_3'$, $1 \leq j_3 \leq N_3''$ are equally spaced grid points in each dimension on $[0,1]^3$ and $N_3 N_3' N_3'' = N$. Here, we mimic the number of voxels of the real data, which usually have different values for N_3, N_3' and N_3''. For each subject, the within-imaging dependence $\eta_i(\cdot)$ is generated from a Gaussian process with mean 0, and covariance function $G_0(\mathbf{x}_j, \mathbf{x}_{j'}) = \sum_{k=1}^{3} \cos(2\pi(x_{kj} - x_{kj'}))$, $j, j' = 1, \ldots, N$. Measurement errors $\epsilon_i(\cdot)$ are generated the same as 2D case. We consider sample size $n = 50, 100, 200$ and $N = 3,000$ ($20 \times 15 \times 10$) and $4,500$ ($30 \times 15 \times 10$). Results of each setting are based on 100 simulations. The training of neural networks architecture (L, p, s) follows the same procedures as in 2D case. Architecture parameters (L, p, ζ) selected from $L \in \{3, 4\}$, $p \in \{100, 300, 500, 1000, 2000, 5000\}$, $\zeta \in \{10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}\}$. The triangularized bivariate splines method proposed in [121] are designed for 2D functions only. Extending spline basis functions for 3D functional data is very sophisticated and to our best knowledge, it is not available for 3D FDA yet. Hence, we only conduct 3D numerical analysis with our proposed DNN method. To exam the performance of the estimator $\widehat{f}$, we also summarizes the empirical L_2 risk and standard deviation of estimators $\widehat{f}_{DNN}$ in Table 2.3. It is clear to find that the empirical risk decrease when sample sizes or observed voxels numbers increase for both noise levels, which supports our theoretical findings. The mean function f_0 and its DNN estimator are presented in Figure 2.2. It is easy to conclude that the DNN estimator follows the the same pattern as the true mean function.

2.6 Real data analysis

The dataset used in the preparation of this article were obtained from the ADNI database (`adni.loni.usc.edu`). The ADNI is a longitudinal multicenter study designed to develop clinical, imaging, genetic, and biochemical biomarkers for the early detection and tracking of AD. From this database, we collect PET data from 79 patients in AD group. This PET dataset

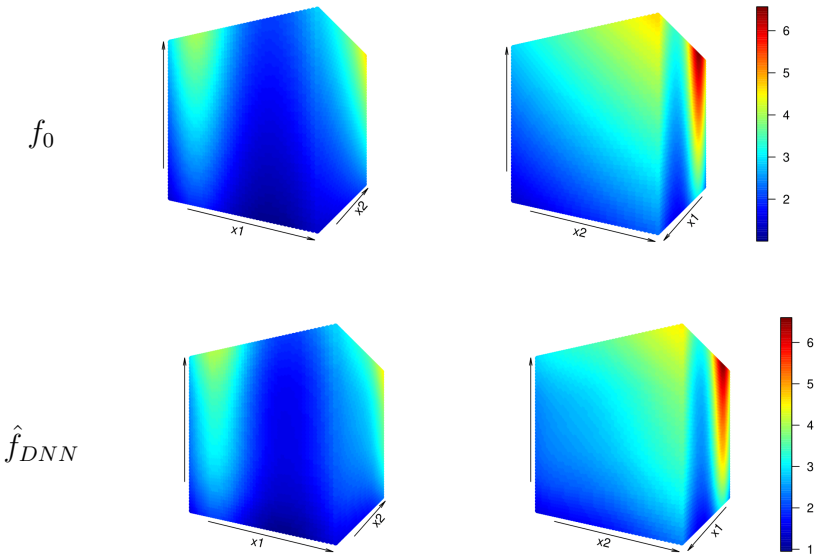

Figure 2.2: Two different angles (Left and Right panels) to view the true mean function and the DNN estimator in 3D simulation case. ($n = 200, N = 4,500, \sigma = 1$)

Table 2.3: The average empirical L_2 risk and their standard deviations of f_0 across 100 simulation runs (3D case).

σ	N	n	L_2 risk	SD
1	3000	50	0.0028	0.0020
		100	0.0011	0.0006
		200	0.0006	0.0004
	4500	50	0.0007	0.0007
		100	0.0005	0.0007
		200	0.0003	0.0004
2	3000	50	0.0030	0.0024
		100	0.0012	0.0007
		200	0.0007	0.0005
	4500	50	0.0009	0.0007
		100	0.0005	0.0008
		200	0.0003	0.0005

has been spatially normalized and post-processed. These AD patients have three to six times doctor visits and we only select the PET scans obtained in the third visits. Patients' age ranges from 59 to 88 and average age is 76.49. There are 33 females and 46 males among these 79 subjects. All scans were reoriented into $79 \times 95 \times 69$ voxels, which means each patient has 69 sliced 2D images with 79×95 pixels. For 2D case, it means each subject has $N = 7,505 = 79 \times 95$

observed pixels for each selected image slice. For 3D case, the observed number of voxels for each patient's brain sample is $N = 79 \times 95 \times 69$, which is more than 0.5 million.

For 2D case, we select the 20-th, 40-th and 60-th slices from 69 slices for each patient. We first take average across 79 patients for each slices (the first row in Figure 2.3). Then, based on the averaged images, we obtain the proposed DNN estimators for each slice (the second row in Figure 2.3). We also recover the image with higher resolutions 512×512 pixels, instead of the original 95×69 pixels for each slice (the third row in Figure 2.3). The neural network (2.5) is trained through optimizer `Adam` with architecture parameters (L, p, ζ) selected from $L \in \{3, 4\}$, $p \in \{500, 1000\}$, $\zeta \in \{10^{-5}, 10^{-6}, 10^{-7}\}$. 10-fold cross-validation method selects the optimal architecture parameters $L_{opt} = 3$, $p_{opt} = 1000$ and $\zeta_{opt} = 10^{-7}$. We used 300 to 500 epochs and 2 to 8 as batch size given different slices.

In 3D case, on 79 patients, and total $79 \times 95 \times 69$ voxels. Same as 2D case, we first average the total 79 3D scans into one 3D scan, and then perform neural network to train the model based on the averaged 3D image. In the bottom row of Figure 2.3, we break down the recovered 3D image and show the recovered 20-th, 40-th and 60-th slices. The neural network (2.5) is trained through optimizer `Adam` with architecture parameters (L, p, ζ) selected from $L \in \{3, 4\}$, $p \in \{1000, 1500\}$, $\zeta \in \{10^{-5}, 10^{-6}, 10^{-7}\}$. 10-fold cross-validation method selects the optimal architecture parameters $L_{opt} = 4$, $p_{opt} = 1500$ and $\zeta_{opt} = 10^{-7}$. According to our numerical experience, we find 300 epochs and 64 batch size providing the reasonable well results.

In Figure 2.4, we also recover the image in higher resolutions $128 \times 128 \times 128$ voxels, which means instead of the original $79 \times 95 \times 69$ voxels, we can provide the estimated image slices with higher resolution (128×128 pixels, instead of the original 79×95 pixels) at finer grid points (128 points, instead of the original 69 points).

2.7 Discussion

In this work, we resolve the model misspecification issue in multi-dimensional FDA via the promising technique from the deep learning domain. By properly choosing network architecture, our estimator achieves the optimal nonparametric convergence rate in empirical norm. To

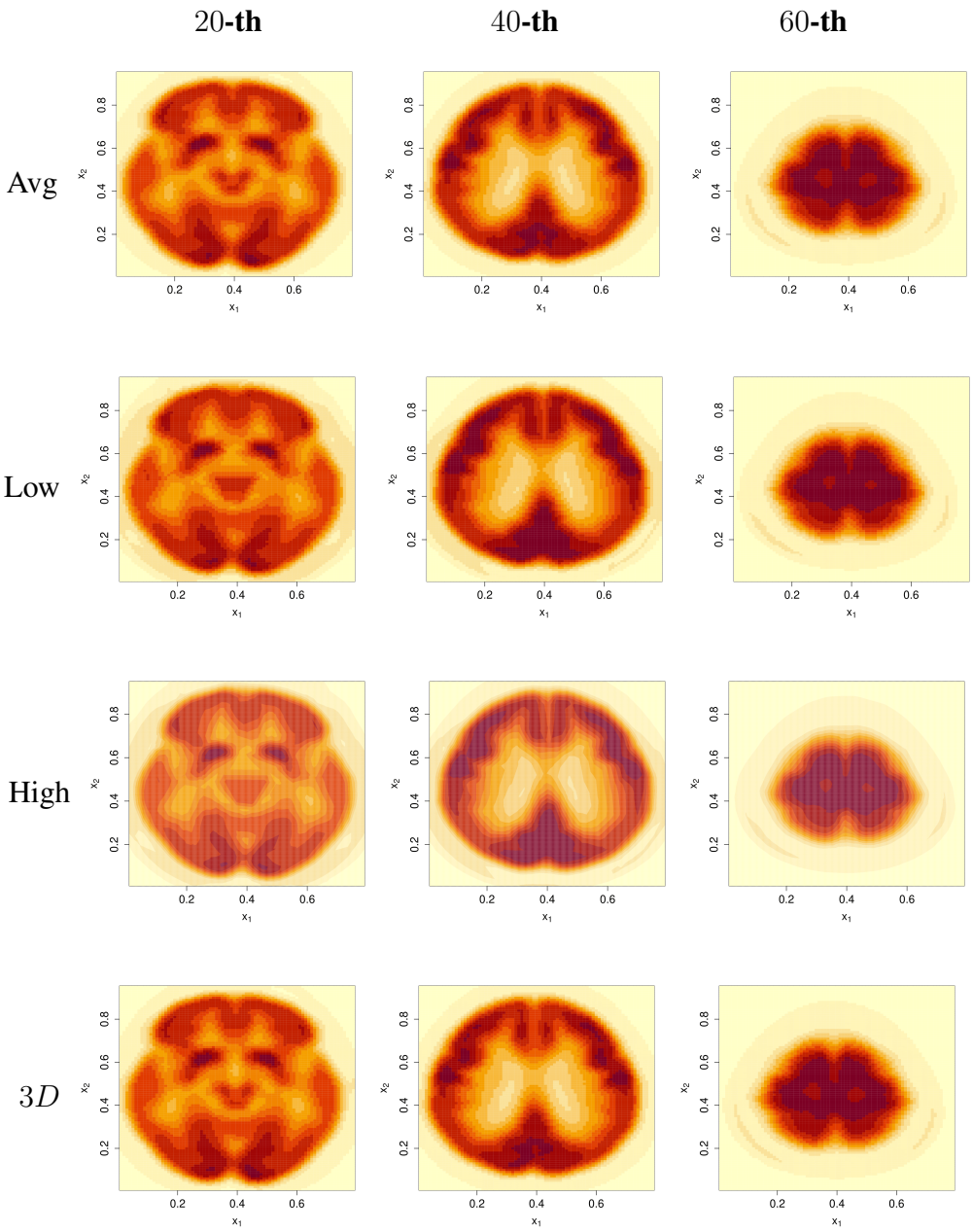

Figure 2.3: From top to bottom are averaged images $\{\overline{Y}_{\cdot j}\}_{j=1}^{7505}$, recovered images $\widehat{f}(x_{1j}, x_{2j'})$, $j = 1, \ldots, 79$, $j' = 1, \ldots, 95$, recovered high resolution (128×128) images $\widehat{f}(x_{1j}, x_{2j})$, $j = 1, \ldots, 128$ and recovered images from $3D$ image. Left: The 20-th slices; Middle: The 40-th slices; Right: The 60-th slices.

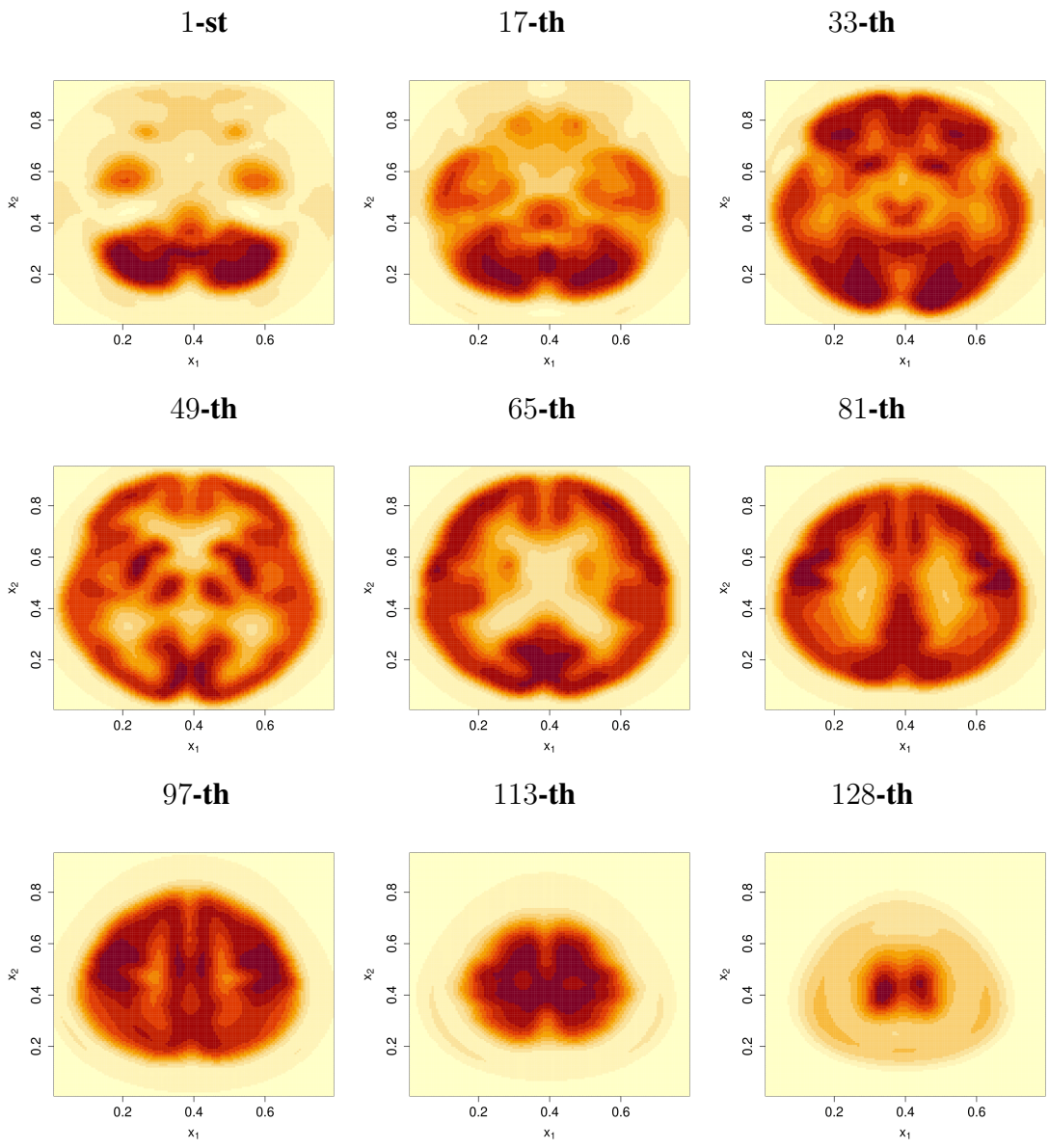

Figure 2.4: Recovered higher resolutions of selected nine slices in 3D case.

our best knowledge, this is the first piece of work in FDA, which yields attractive empirical convergence rate for multi-dimensional FDA, and at meanwhile is free from model misspecification. Numerical analysis demonstrates that our approach is useful in recovering the signal for imaging data. Some interesting future works may include the functional linear regression model and classification problems in the framework of DNN.

Chapter 3

Robust Deep Neural Network Estimation for Multi-Dimensional Functional Data

3.1 Introduction

We consider the problem of robust estimation of the location function from a collection of functional observations defined over $\mathbb{R}^d$ ($d \geq 1$) a multi-dimensional domain. To be precise, let $\xi = \{\xi(\mathbf{X}) : \mathbf{X} \in \mathcal{I}\}$ be a compactly supported random field, i.e., a real-valued second-order stochastic process on a compact set $\mathcal{I} \subset \mathbb{R}^d$. Such data are nowadays commonly referred to as functional data. In many applications, data are collected over one-dimensional domains (i.e., $d = 1$) such as time-varying trajectories and relevant research has been enjoying considerable popularity. The readers are referred to some monographs [87, 115, 54, 46] for a comprehensive overview of functional data analysis (FDA). Thanks to the improved capabilities of data recording and storage, as well as advances in scientific computing and data science, many new forms of functional data have emerged. Instead of traditional unidimensional functional data, multi-dimensional functional data becomes increasingly common in various fields, such as geographical science and neuroscience. For example, for the early detection and tracking of Alzermer's diease, the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (`adni.loni.usc.edu`) contains each individual's 3D brain-scans. Despite the promising of multi-dimensional functional data, statistical methods for such data are limited, except for very few existing works, for example, [24, 131, 25].

A fundamental problem in FDA is the estimation of central tendency, yet most current estimation procedures either lack robustness with respect to the many kinds of anomalies one can encounter in the functional setting or only focus on the robustness for unidimensional

scenario. The fact that robust estimation has not been widely investigated for multi-dimensional scenario is certainly not owing to lack of interesting applications, but to the greater technical difficulty to handle such loss function for multi-dimensional functional data and establish their theoretical properties.

To give some background on our proposed method for multi-dimensional functional data, we first review several relevant robust FDA methods that have been developed for analyzing unidimensional functional data. [9, 59] proposed robust estimators for the functional principal components by adapting the projection pursuit approach and based on MM estimation, respectively. [76] established a robust version of spline-based estimators for a linear functional regression model. [96] proposed a robust procedure based on convex and non-convex loss functions in functional linear regression models. Recently, [65, 66] proposed robust estimators and associated simultaneous confidence bands for the mean function of functional data using least absolute deviation and M-estimation, respectively.

We notice that there are few exiting works on robust methods for analyzing so-called two-way functional data which consist of a data matrix whose row and column domains are both structured, as when the data are time series collected at different locations in space. For example, [130] develop a robust regularized singular value decomposition method for analyzing such special type functional data. It is formulated as a penalized loss minimization problem and a pre-decided two-way roughness penalty function is used to ensure smoothness along each of the two functional domains. As this method is only designed for the special two-way functional data, it can not be adopted to the general multi-dimensional FDA directly. Furthermore, a lack of theoretical analysis provides inadequate assurance to robust methods practitioners.

To remedy these deficiencies, we introduce the first class of optimal robust location estimators based on the deep neural network (DNN) method. DNN is one of the most promising and vibrate areas in deep learning. DNN has been recently applied in various nonparametric regression problems recently, they have been shown to successfully overcome the curse of dimensionality in nonparametric regression; see [92, 11, 69, 70]. There are also some works proposed for deep learning algorithms for FDA from the statistical point of view [104]. Based on the sparsely connected DNN, [118] proposed a DNN estimator for the mean function from

functional data based on the least squares neural network regression. However, none of them works on the robust statistics, not to mention the proven theoretical results for robust FDA.

The contributions of this work are three-fold. First, to the best of our knowledge, this is the first work on proposing DNN based robust estimator for FDA. We propose a broad class of M-type RDNN (robust DNN) estimators to estimate location functions for multi-dimensional functional data. Second, RDNN estimators come with theoretical guarantees. In particular, we study the rate of convergence of the estimator under weak assumptions and show that the estimator is rate-optimal even for any d-dimensional functional data. By borrowing the strength from the DNN, the convergence rate of the proposed RDNN estimator does not depend on the dimension d. Finally, our analyses are fully nonparametric. At the meanwhile, RDNN estimator does not suffer the curse-of-dimensionality which is a classical drawback in the traditional nonparametric regression framework.

This chapter is structured as follows. Section 3.2 provides the model setting in FDA and introduces multilayer feed-forward artificial neural networks and discusses mathematical modeling. The implementation on hyperparameter selections also be included in Section 3.2. The theoretical properties of the proposed RDNN estimator can be found in Section 3.3. Section 3.4 provides the detailed implementation on neural network's architecture selecting and training. In Section 3.5, it is shown that the finite sample performance of proposed neural network estimator. The proposed method is applied to the spatially normalized positron emission tomography (PET) data from ADNI in Section 3.6 and make some concluding remarks in Section 3.7. Technical proofs are collected in the Appendix.

3.2 The model and the robust deep neural network estimator

3.2.1 FDA model

Let us first assume the process $\left\{\xi(\mathbf{X}), \mathbf{X} \in [0,1]^d\right\}$ is L^2, i.e., $E\int_{[0,1]^d} \xi^2(\mathbf{X})d\mathbf{X} < +\infty$. In the classical FDA setting, $d = 1$ refers to the index variable as time. When $d = 2, 3$, it could also be a spatial variable, such as in image or geoscience applications. We model the multi-dimensional functional data as noisy sampled points from a collection of trajectories that are

assumed to be independent realizations of a smooth random function $\xi(\mathbf{X})$, with unknown mean function $f_0(\mathbf{X}) = \mathbb{E}\{\xi(\mathbf{X})\}$. We consider a version of the model that incorporates uncorrelated measurement errors. Let $\xi_1, \ldots, \xi_n$ denote n independent and identically distributed (i.i.d.) copies of ξ at points $\mathbf{X} = (X_1, \ldots, X_d)$, $1 \leq i \leq n$. Our goal is to recover the mean function $f_0(\mathbf{X}_j)$ from the noisy observations of the discretized functional data:

$$Y_{ij} = \xi_i(\mathbf{X}_j) + e_i(\mathbf{X}_j), \quad i = 1, 2, \ldots, n, j = 1, 2, \ldots, N, \tag{3.1}$$

where $e_i(\mathbf{X}_j)$ are random noise variables. In [126, 21, 20], it is assumed that the noise variables $e_i(\mathbf{X}_j)$ are independent of the ξ_i and i.i.d. with zero mean and finite variance. However, we allow for correlated errors that are not necessarily independent of the functional curves.

In terms of mean-deviations, model (3.1) can be equivalently written as

$$Y_{ij} = f_0(\mathbf{X}_j) + \epsilon_i(\mathbf{X}_j), \quad i = 1, 2, \ldots, n, j = 1, 2, \ldots, N, \tag{3.2}$$

where $\epsilon_i(\mathbf{X}_j) = \xi_i(\mathbf{X}_j) - \mathbb{E}\{\xi_i(\mathbf{X}_j)\} + e_i(\mathbf{X}_j)$ denotes the error process associated with the i-th response evaluated at $\mathbf{X}_j$. The problem is thus reformulated as a regression problem with repeated measurements and possibly correlated errors. In the following, for simple notations, we consider the equally spaced design. The main results can be extended to irregular spaced design.

3.2.2 Robust deep neural network estimator

We first briefly introduce the necessary notations and terminologies used in the neural networks. Popular choice of activation functions includes rectified linear unit (ReLU), sigmoid, and tanh. In this article, we will mainly focus on neural networks with the ReLU activation function, i.e., $\sigma(x) = (x)_+$ for $x \in \mathbb{R}$. For any real vector $\boldsymbol{y} = (y_1, \ldots, y_r)^\top$, define the shift activation function $\sigma(\boldsymbol{y}) = (\sigma(y_1), \ldots, \sigma(y_r))^\top$. For an integer $L \geq 1$ and $\boldsymbol{p} = (p_0, p_1, \ldots, p_L, p_{L+1}) \in \mathbb{N}^{L+2}$, let $\mathcal{F}(L, \boldsymbol{p})$ denote the class of DNN, with L hidden layers and p_l nodes on hidden layer l, for $l = 1, \ldots, L$. We consider the feed-forward neural network class, and any $f \in \mathcal{F}(L, \boldsymbol{p})$

has a composition structure, i.e.,

$$f(\mathbf{x}) = \mathbf{W}_L \sigma\left(\mathbf{W}_{L-1} \ldots \sigma\left(\mathbf{W}_1 \sigma\left(\mathbf{W}_0 \mathbf{x} + \mathbf{u}_0\right) + \mathbf{u}_1\right) + \ldots + \mathbf{u}_{L-1}\right) + \mathbf{u}_L, \quad \mathbf{x} \in \mathbb{R}^d, \quad (3.3)$$

where $\mathbf{W}_l \in \mathbb{R}^{p_{l+1} \times p_l}$ are weight matrices and $\boldsymbol{u}_l \in \mathbb{R}^{p_l}$ are shift vectors, for $l = 1, \ldots, L$. Owing to the large capacity of neural network class, it tends to overfit the training dataset easily. To avoid the overfitting and reduce the computational burden, we train the robust estimator using the following s-sparse ReLU DNN class:

$$\begin{aligned} & \mathcal{F}(L, \boldsymbol{p}, s) \\ = & \left\{ f \in \mathcal{F}(L, \boldsymbol{p}) : \sum_{l=0}^{L} \|\mathbf{W}_l\|_0 + \|\mathbf{u}_l\|_0 \leq s, \max_{l=0,\ldots,L} \|\mathbf{W}_l\|_\infty + \|\mathbf{u}_l\|_\infty \leq 1, \right. \\ & \left. \|f\|_\infty \leq 1 \right\}, \end{aligned} \quad (3.4)$$

where $\|\cdot\|_\infty$ denotes the maximum-entry norm of a matrix/vector or supnorm of a function, $\|\cdot\|_0$ denotes the number of non-zero entries of a matrix or vector, $s > 0$ controls the number of nonzero weights and shift. The selecting procedures of unknown tuning parameters $(L, \mathbf{p}, s)$ shall be given in Section 3.4. To simplify the notations, we write $\mathcal{F}$ instead of $\mathcal{F}(L, \boldsymbol{p}, s)$ in the following.

In the regression model, the common objective is to find an optimal estimator by minimizing a loss function. In the DNN setting, this coincides with training neural networks by minimizing the empirical risk over all the training data. In particularly, given the networks in (3.4), the proposed RDNN estimator is defined as

$$\widehat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{nN} \sum_{i=1}^{n} \sum_{j=1}^{N} \rho\left(Y_{ij} - f\left(\mathbf{X}_j\right)\right), \quad (3.5)$$

where ρ is some convex nonnegative loss function satisfying $\rho(0) = 0$ and $\mathcal{F}$ is some function class. This formulation is very general, allowing the flexibility in the choice of the loss function, so that better resistance towards outlying observations is achieved. One of the well-known examples of such loss functions is Huber's loss function given by $\rho_k(x) = x^2/2\mathbb{I}(|x| \leq k) +$

$k(|x| - k/2)\mathbb{I}(|x| > k)$, where $\mathbb{I}(\cdot)$ is the indicator function, and $k > 0$ controls the blending of square and absolute losses. Furthermore, the symmetry of the loss function in (3.5) is not required, such versatile estimators may be readily incorporated into the present framework. Indeed, to estimate conditional quantiles, one would only need to select the loss function as $\rho(x) = x(\tau - \mathbb{I}(x < 0))$ for some $\tau \in (0, 1)$. Finally, the asymptotic properties of quantile estimators are covered by the theory developed in Section 3.3.

3.3 Theoretical properties of the RDNN estimator

3.3.1 Definitions and notations

Define the ball of β-Hölder functions with radius K as

$$\mathcal{C}_d^{\beta}(D, K) = \left\{ f : D \subset \mathbb{R}^d \to \mathbb{R} : \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|<\beta} \|\partial^{\boldsymbol{\alpha}} f\|_\infty + \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|=\lfloor\beta\rfloor} \sup_{\boldsymbol{x},\boldsymbol{y}\in D,\boldsymbol{x}\neq\boldsymbol{y}} \frac{|\partial^{\boldsymbol{\alpha}} f(\boldsymbol{x}) - \partial^{\boldsymbol{\alpha}} f(\boldsymbol{y})|}{|\boldsymbol{x} - \boldsymbol{y}|_\infty^{\beta-\lfloor\beta\rfloor}} \leq K \right\},$$

where $\partial^{\boldsymbol{\alpha}} = \partial^{\alpha_1} \ldots \partial^{\alpha_d}$ with $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}^d$ and $|\boldsymbol{\alpha}| := |\boldsymbol{\alpha}|_1$.

We assume the true location function f_0 has the natural composition structure, i.e.,

$$f_0 = g_q \circ g_{q-1} \circ \ldots \circ g_1 \circ g_0,$$

where $g_\ell : [a_\ell, b_\ell]^{d_\ell} \to [a_{\ell+1}, b_{\ell+1}]^{d_{\ell+1}}$, $g_\ell = (g_{\ell j})_{j=1,\ldots,d_{\ell+1}}^\top$, $\ell = 1, \ldots, q$, with unknown parameters d_ℓ and q. We assume each $g_{\ell j}$ is β_ℓ-Hölder function with radius K_ℓ. Let t_ℓ be the maximal number of variables on which each of the $g_{\ell j}$ depends on t_ℓ, and $t_\ell \leq d_\ell$. Since $g_{\ell j}$ is also t_ℓ-variate, the true underlying function space becomes

$$\begin{aligned} & \mathcal{G}\left(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K}\right) \\ := \; & \left\{ f = g_q \circ \ldots \circ g_0 : g_\ell = (g_{\ell j})_j : [a_\ell, b_\ell]^{d_\ell} \to [a_{\ell+1}, b_{\ell+1}]^{d_{\ell+1}}, \right. \\ & \left. g_{\ell j} \in \mathcal{C}_{t_\ell}^{\beta_\ell}\left([a_\ell, b_\ell]^{t_\ell}, K_\ell\right), |a_\ell|, |b_\ell| \leq K_\ell \right\}, \end{aligned} \tag{3.6}$$

with $\mathbf{d} := (d_0, \ldots, d_{q+1})$, $\mathbf{t} := (t_0, \ldots, t_q)$, $\boldsymbol{\beta} := (\beta_0, \ldots, \beta_q)$, $\mathbf{K} := (K_0, \ldots, K_q)$ and $\beta_\ell^* := \beta_\ell \prod_{k=\ell+1}^{q} (\beta_k \wedge 1)$.

3.3.2 Assumptions

In this section, we develop the convergence rate of the proposed RDNN estimator in (3.5). For simple notations, $\log$ denotes the logarithmic function with base 2. For sequences $(a_n)_n$ and $(b_n)_n$, $a_n \asymp b_n$ means $a_n \leq c_1 b_n$ and $a_n \geq c_2 b_n$ where c_1 and c_2 are absolute constants for any n.

We now introduce the main assumptions:

(A1) The true regression function $f_0 \in \mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K})$.

(A2) The RDNN estimator $\widehat{f} \in \mathcal{F}(L, \mathbf{p}, s)$, where $L \asymp \log(nN^\nu)$, $s \asymp (nN^\nu)^{\frac{1}{\theta+1}}$, $\min_{l=1,\ldots,L} p_l \asymp (nN^\nu)^{\frac{1}{\theta+1}}$, for $\theta = \min_{\ell=0,\ldots,q} \frac{2\beta_\ell^*}{t_\ell}$ and $\nu \geq 0$.

(A3) The loss function ρ is an absolutely continuous convex function on $\mathbb{R}$ with derivative ψ existing almost everywhere.

(A4) There exist finite constants κ and c_1 such that for all $x \in \mathbb{R}$ and $|x'| < \kappa$, $|\psi(x + x') - \psi(x)| < c_1$.

(A5) There exist a finite constant c_2 such that $\sup_{j \leq N} \mathbb{E}\{|\psi(\epsilon_{1j} + u) - \psi(\epsilon_{1j})|^2\} < c_2|u|$, as $|u| \to 0$.

(A6) $\sup_{j \leq N} \mathbb{E}\{(\psi(\epsilon_{1j}))^2\} = O(N^{-\nu})$, for some constant $\nu \geq 0$, and $\mathbb{E}\{\psi(\epsilon_{1j})\} = 0$. There exist finite constants δ_j, $j = 1, \ldots, N$ such that $0 < \inf_{j \leq N} \delta_j \leq \sup_{j \leq N} \delta_j < \infty$ and $\sup_{j \leq N} |\mathbb{E}\{\psi(\epsilon_{1j} + u)\} - \delta_j u| = o(u)$, as $|u| \to 0$.

Assumption (A1) is a natural definition for the neural network, which is fairly flexible and many well known function classes are contained in it. For example, the generalized additive model $f_0(\mathbf{x}) = h\left(\sum_{i=1}^d f_i(x_i)\right)$, can be written as a composition of three functions $f_0 = g_2 \circ g_1 \circ g_0$, with $g_0(x_1, \ldots, x_d) = (f_1(x_1), \ldots, f_d(x_d))$, $g_1(x_1, \ldots, x_d) = \sum_{i=1}^d x_i$, and $g_2 = h$. Assumption (A2) depicts the architecture and parameters' setting in the network space. To use

discontinuous score functions, Assumptions (A3)-(A6) impose some regularity on the error process and its finite-dimensional distributions. In particularly, Assumptions (A3) guarantees the existence of the solution of the optimization problem in (3.5). Most of the loss functions chosen in practice satisfy this condition, such as the Huber loss function. Assumptions (A4) and (A5) require boundedness and some regularity of the score function, which are standard conditions for M-estimation procedures for FDA, see the similar conditions required in [65]. For the first part of Assumption (A6), when considering the classical L_2 loss, it essentially makes sure the largest element of the covariance function is finite and decreases when the number of measurements increases. They are standard regularity conditions for the covariance functions in FDA literature, see [20, 65, 118] for example. The second part of Assumption (A6) essentially requires that for any $j = 1, \ldots, N$, function $h_j(u) = E\{\psi(\epsilon_{1j} + u)\}$, is differentiable with strictly positive derivative at the origin. This is a necessary condition for the minimum to be well-separated in the limit. Assumption (A6) on the score function ψ is also standard conditions in M-estimation for functional data literature, see [65, 50]. It is also not stringent assumptions for errors, for example, ϵ_{ij}'s following a zero mean Gaussian process or mixture Normal–Cauchy distribution. We provide more detailed examples for ϵ_{ij}'s in Section 3.5.

3.3.3 Unified rate of convergence

The following theorem establishes the unified convergence rate of the RDNN estimator $\widehat{f}$ for any multi-dimensional functional data under the empirical norm. Its proof and some technical lemmas are provided in the Appendix.

Theorem 3.1. *Under Assumptions (A1)-(A6), we have*

$$\|\widehat{f} - f_0\|_N^2 = O_p(nN^\nu)^{-\frac{\theta}{\theta+1}} \log^6(nN^\nu), \tag{3.7}$$

where $\nu \geq 0$, $\theta = \min_{\ell=0,\ldots,q} \frac{2\beta_\ell^*}{t_\ell}$.

It is interesting to observe that Theorem 3.1 obtains the same rate of convergence derived in non-robust estimation in [118].

3.4 Implementation

Different from classical nonparametric estimators, $\widehat{f}$ has no analytical expression or basis expansion expression. The proposed robust estimator is constructed using the neural network class which is fully characterized by the architectures $(L, \boldsymbol{p}, s)$. In this section, we provide the detailed computational procedure for the proposed RDNN estimator in (3.5).

3.4.1 Neural network's architecture selection

In the DNNs' computations, tuning parameters are crucial as they control the overall behavior of the proposed estimator and the learning process. The tuning parameters are so-called network architecture parameters, which include the number of layers L, the number of hidden neurons within these layers $\boldsymbol{p}$, and sparse parameter s. There are fairly rich literature discussing the optimization selection, such as grid search, random search, and Bayesian optimization. Nevertheless, the selection of network architecture parameters has been rarely discussed. In practice, some model selection methods such as cross-validation may have good performances, but with huge computational burdens. For this reason, considering both the computational efficiency and the theoretical guarantee, we select architecture parameters based on Theorem 3.1. Particularly, let $\nu = \frac{1}{2}$ and $\theta = \frac{1}{2}$ in Assumption (A2), and choose $L = \lceil 0.5 \log(nN^{1/2}) \rceil$, $p_l = \lceil 10 n^{1/2} N^{1/4} \rceil$, $s = \lceil 5 n^{1/2} N^{1/4} \rceil L$. The specific choice of ν and θ includes a large scope of true function classes. Note that in our considered sparse neural network space $\mathcal{F}$, the sparse parameter s should be carefully selected. When designing the network architecture practically, the dropout rate is suggested as $\lceil 5 n^{1/2} N^{1/4} \rceil (\lceil 10 n^{1/2} N^{1/4} \rceil)^{-1}$ in each layer during the optimization procedure.

3.4.2 Training neural networks

The minimization in (3.5) is generally a computational cumbersome optimization problem owing to non-linearities and non-convexities. The most commonly used solution utilises stochastic gradient descent (SGD) to train a neural network. SGD uses a batch of a specific size, that is, a small subset of the data (typical size $B = 2^2$ to 2^{10}) is randomly drawn at each iteration

of optimization to evaluate the gradient, to alleviate the computation hurdle. Our input size of network is nN, thus we choose relatively large batches B from 256 to 512 depending on the performance of convergence. A pass of the whole training set is called an epoch. Typical choices of epochs are 200, 300 and 500. The number of epochs defines the number of times that the learning algorithm works through the entire training dataset. The step of the derivative at each epoch is controlled by the learning rate which is 0.001. The readers are referred to recent monographs ([35]) for a general discussion of these numerical challenges. There are certainly some challenges for SGD to train DNN. For example, albeit good theoretical guarantees for well-behaved problems, SGD might converge very slowly; the learning rates are difficult to tune ([2]). To address these challenges, several variants gradient based optimization algorithms are introduced, such as Adam, RMSprop and Adadelta. Instead of the classical SGD procedure, Adam is a method for efficient stochastic optimization that only requires first-order gradients with little memory requirement . Hence, it is well suited for problems when there are large sample sizes and parameters ([53]), and is widely used in network training for functional data, such as [118]. In our numerical studies, Adam provides the best results and is the most computationally efficient among these candidates. We recommend Adam in the real-life applications.

3.5 Simulation

To illustrate the finite sample performance of the introduced RDNN estimators based on our proposed neural networks method, we conduct substantial simulations for both 2D and 3D functional data. All experiments are conducted in `R`. We summarize `R` codes and examples for the proposed RDNN algorithms on `GitHub` (`https://github.com/FDASTATAUBURN/RDNN`).

3.5.1 2D simulation

The 2D functional data are generated from the model:

$$Y_{ij} = f_0\left(\mathbf{X}_j\right) + \epsilon_{ij}, \tag{3.8}$$

where the true mean function $f_0(\mathbf{x}_j) = -8\left[1+\exp\left\{\cot(x_{1j}^2)\cos(2\pi x_{2j})\right\}\right]^{-1}$, and $\mathbf{x}_j = (j_1/N_2, j_2/N_2)$, $1 \le j_1, j_2 \le N_2$, are the equally spaced grid points on $[0,1]^2$, and $N_2^2 = N$. The error term is $\epsilon_{ij} = \eta(\mathbf{X}_j) + e_{ij}$, where $\eta(\cdot)$ is generated from a Guassian process, with zero mean and covariance function $G_0(\mathbf{x}_j, \mathbf{x}_{j'}) = \sum_{k=1}^{2}\cos(2\pi(x_{kj} - x_{kj'}))$, $j, j' = 1, \ldots, N$. The measurement errors e_{ij}'s are i.i.d. standard normal random variables.

Under the proposed functional model (3.5.1), we introduce outlier hyper-surfaces to the generated functional sample by randomly contaminating a subset, R^o, of the original sample. The contamination proportion r is chosen to be 0, 0.1 and 0.2. The similar simulation setting has been considered in [65, 66]. We consider the following four types of outliers, i.e., two surface outliers and two heavy-tailed distributed outliers. They mimic the types of noised data usually encountered in the real dataset in Section 2.6.

Case 1: *Stripe outliers* To simulate outliers on a stripe in 2D regions, the contamination occurs on a line segment $a_0 \times \mathcal{I}$, that is,

$$Y_{ij^*}^o = Y_{ij^*} + \epsilon_{ij^*}^o,\ i \in R^o, \quad j_1^*/N_2 = a_0, \quad j_2^*/N_2 \in \mathcal{I},$$

where $\epsilon_{ij^*}^o \sim U(10, 20)$. In this simulation, $a_0 = 0.2$, and we choose (i) $\mathcal{I} = \cup_{k=1}^{5}\left[\frac{2k-2}{10}, \frac{2k-1}{10}\right)$, and (ii) $\mathcal{I} = [0,1]$.

Case 2: *Square outliers* To simulate outliers on a consecutive 2D region, the contamination occurs on a square $[a_0, a_1]^2$, that is,

$$Y_{ij^*}^o = Y_{ij^*} + \epsilon_{ij^*}^o,\ i \in R^o, \quad (j_1^*/N_2, j_2^*/N_2) \in [a_0, a_1]^2$$

where $\epsilon_{ij^*}^o \sim U(10, 20)$. In the simulation, we choose (i) $[a_0, a_1]^2 = [0.1, 0.3]^2$, and (ii) $[a_0, a_1]^2 = [0.1, 0.5]^2$.

Case 3: *Mixture Normal–Cauchy* To simulate outliers with heavy-tailed distribution, the distribution of $\epsilon_{ij^*}^o$'s follow a mixture of a normal distribution $N(0,1)$ and a Cauchy distribution with location 0 and scale 0.5. The mixture rates are (i) 0.3, and (ii) 0.5.

Case 4: *Mixture Normal–Slash* Similar to previous case, but using a mixture of a normal distribution $N(0,1)$ and a Slash distribution with location 0 and scale 0.5. The mixture rates are (i) 0.3, and (ii) 0.5.

We consider sample size $n = 50, 100, 200$. For each image, let $N_2 = 10$, implicating the number of observational points (pixels) is set to be $N = N_2^2 = 100$. The network architecture is determined in a data driven way as suggested in Section 3.4.1, and we use Huber's loss function with tuning parameter 1 for RDNN estimator in (2.5). The results of each setting are based on 100 Monte Carlo simulations. Figures 3.1 presents heat maps of a typical set of the true mean function and abnormal observations, along with the estimations of RDNN and DNN estimators. From Table 3.1, we can see that when training the clean data, DNN method has comparable L_2 risks with RDNN estimators. These risks decrease as the sample size n increases. However, when contamination is involved, Table 3.2 shows that the risks of DNN estimators elevated drastically, while RDNN ones keep consistent results. In addition, although increasing either contamination rate r or contamination areas on a stripe raises the risks, we can see that RDNN estimators perform steady and remains relatively small L_2 risks even given 20% data contain anomalies. From Table 3.2, we can also see that when contamination occurs in a square region, the same trend is revealed, as previous discussion. It is worth mentioning that when $r = 0.2$, for the contaminated region $[0.1, 0.5]^2$, DNN estimators has extremely large risks, which are more than 10 times of ones of RDNN. Similar findings can be concluded from Table 3.3, where the random errors following non-Gaussian heavy-tail distributions. The RDNN estimator best mitigates the effect of this contamination relative to its competitors. Overall, the present simulation experiments suggest that RDNN perform well in clean data and safeguard against outlying observations either in the form of outlying surfaces or heavy-tailed measurement errors.

Table 3.1: Empirical L_2 risk of 2D uncontaminated data with standard errors in brackets.

n	RDNN	DNN
50	0.114 (0.040)	0.125 (0.049)
100	0.059 (0.029)	0.055 (0.028)
200	0.034 (0.016)	0.031 (0.017)

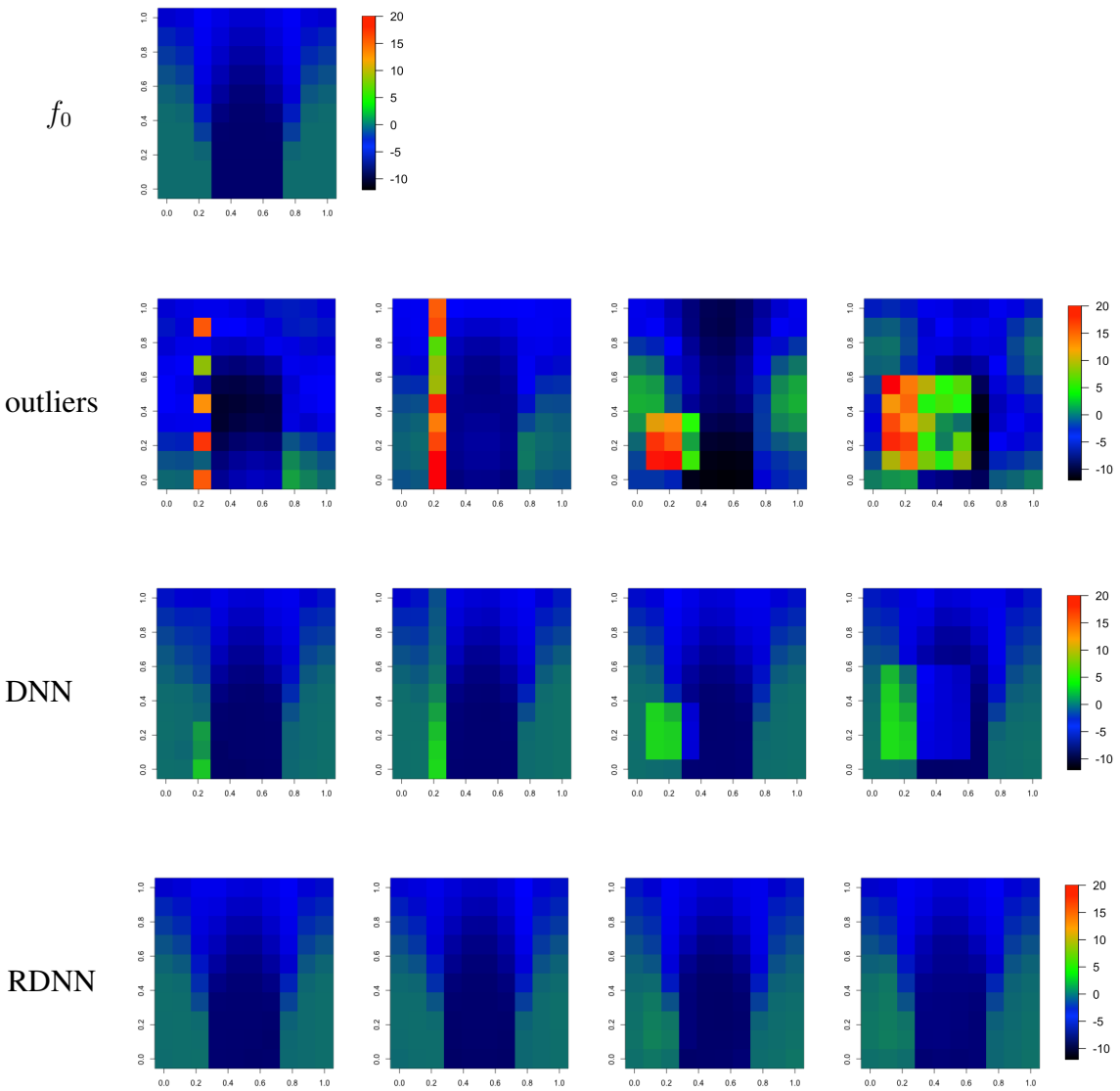

Figure 3.1: 2D simulation for mixed Cauchy and mixed Slash distribution. The first row: true function f_0; The second row to forth row present the contaminated data Y^O, DNN estimations , RDNN estimations. From left to right, the observed data are generated from Case 1 (i) and (ii), Case 2 (i) and (ii).

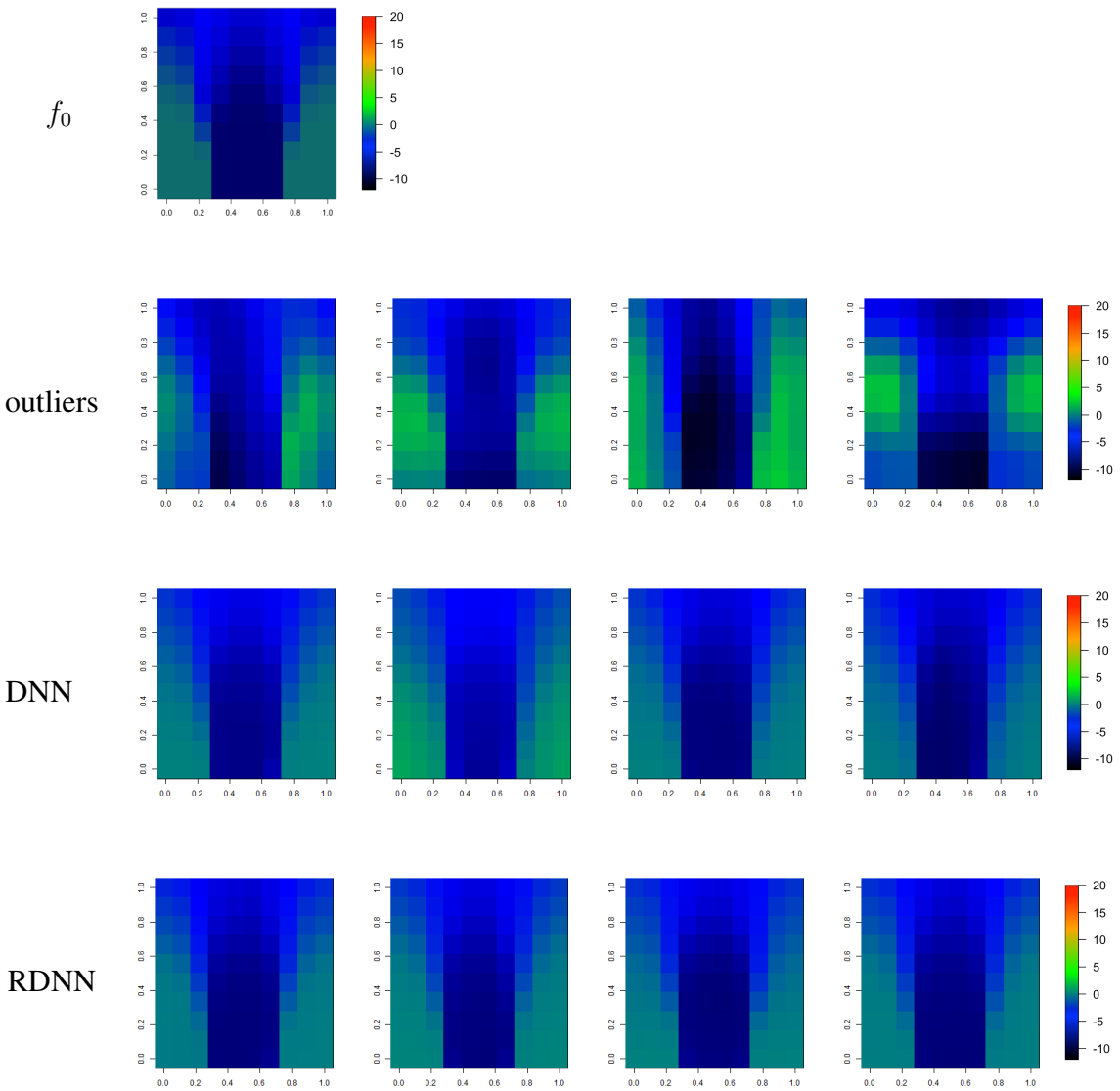

Figure 3.2: 2D simulation for mixed Cauchy and mixed Slash distribution. The first row: true function f_0; The second row to forth row present the contaminated data Y^O, DNN estimations , RDNN estimations. From left to right, the observed data are generated from Case 3 (i) and (ii), Case 4 (i) and (ii).

Table 3.2: Empirical L_2 risk of 2D contaminated data in Cases 1 and 2 with standard errors in brackets.

contaminated regions		n	$r = 0.1$		$r = 0.2$	
			RDNN	DNN	RDNN	DNN
stripe	$\cup_{k=1}^{5}\left[\frac{2k-2}{10}, \frac{2k-1}{10}\right)$	50	0.115 (0.048)	0.179 (0.078)	0.128 (0.055)	0.329 (0.095)
		100	0.055 (0.023)	0.102 (0.033)	0.065 (0.033)	0.252 (0.055)
		200	0.032 (0.015)	0.081 (0.024)	0.041 (0.018)	0.257 (0.043)
	$[0, 1]$	50	0.137 (0.066)	0.311 (0.081)	0.151 (0.051)	0.864 (0.164)
		100	0.064 (0.027)	0.240 (0.055)	0.088 (0.029)	0.842 (0.112)
		200	0.036 (0.015)	0.226 (0.032)	0.064 (0.023)	0.848 (0.084)
square	$[0.1, 0.3]^2$	50	0.118 (0.048)	0.260 (0.093)	0.154 (0.071)	0.664 (0.170)
		100	0.065 (0.033)	0.195 (0.059)	0.069 (0.025)	0.754 (0.107)
		200	0.038 (0.019)	0.195 (0.042)	0.054 (0.022)	0.752 (0.091)
	$[0.1, 0.5]^2$	50	0.151 (0.060)	0.657 (0.159)	0.234 (0.080)	2.014 (0.297)
		100	0.078 (0.042)	0.533 (0.111)	0.134 (0.063)	2.070 (0.248)
		200	0.046 (0.023)	0.550 (0.091)	0.108 (0.042)	2.172 (0.191)

Table 3.3: Empirical L_2 risk of 2D contaminated data in Cases 3 and 4 with standard errors in brackets.

error types	n	mixture rate$= 0.3$		mixture rate$= 0.5$	
		RDNN	DNN	RDNN	DNN
Cauchy	50	0.186 (0.069)	0.665 (0.959)	0.191 (0.069)	1.343 (3.193)
	100	0.097 (0.044)	0.289 (0.265)	0.104 (0.586)	0.586 (0.799)
	200	0.051 (0.029)	0.140 (0.175)	0.053 (0.024)	0.104 (0.066)
Slash	50	0.142 (0.065)	0.456 (0.686)	0.136 (0.071)	0.949 (2.022)
	100	0.074 (0.033)	0.419 (0.948)	0.071 (0.033)	0.822 (1.533)
	200	0.054 (0.027)	0.304 (0.617)	0.055 (0.029)	0.544 (1.004)

3.5.2 3D simulation

For 3D simulation, the functional data are generated from the model (3.8). The true mean function is $f_0(\mathbf{x}_j) = f_0(x_{1j}, x_{2j}, x_{3j}) = \exp\left(\frac{1}{3}x_{1j} + \frac{1}{3}x_{2j} + \sqrt{x_{3j} + 0.1}\right)$, where $\mathbf{x}_j = (x_{1j}, x_{2j}, x_{3j}) = (j_1/N_3, j_2/N_3, j_3/N_3)$, $1 \leq j_1, j_2, j_3 \leq N_3$, are equally spaced grid points in $[0, 1]^3$ and $N = N_3^3 = 5^3$. Generate $\eta(\cdot)$ from a Guassian process, with zero mean and covariance function $G_0(\mathbf{x}_j, \mathbf{x}_{j'}) = \sum_{k=1}^{3} \cos(2\pi(x_{kj} - x_{kj'}))$, $j, j' = 1, \ldots, N$, and the measurement errors e_{ij}'s are i.i.d. random variables generated from standard normal distribution. To contaminate the clean data, we apply the similar settings in Section 3.5.1.

Case 5 To simulate outliers on a consecutive 3D region, the contamination occurs on a square $[a_0, a_1]^3$, that is,

$$Y^o_{ij^*} = Y_{ij^*} + \epsilon^o_{ij^*},\ i \in R^o, \quad (j_1^*/N_3, j_2^*/N_3, j_3^*/N_3) \in [a_0, a_1]^3$$

where $\epsilon^o_{ij^*} \sim U(10, 20)$. In the simulation, we choose $[a_0, a_1]^3 = [0.10, 0.20]^3$ and $[0.10, 0.30]^3$ for different contamination proportions.

Case 6 *Mixture Normal–Cauchy* Similar to case 3, the distribution of $\epsilon^o_{ij^*}$'s follow a mixture of a normal distribution $N(0, 1)$ and a Cauchy distribution with location 0 and scale 0.5. The mixture rates are (i) 0.3, and (ii) 0.5.

Case 7: *Mixture Normal–Slash* Similar to case 4, the distribution of $\epsilon^o_{ij^*}$'s follow a mixture of a normal distribution $N(0, 1)$ and a Slash distribution with location 0 and scale 0.5. The mixture rates are (i) 0.3, and (ii) 0.5.

The results of each setting are based on 100 Monte Carlo simulations for sample sizes are $50, 100$, and 200. For reference, Table 3.4 shows the average of empirical L_2 risks for clean data. We find that when data are clean, both of RDNN and DNN provide comparable estimations results, and the empirical risk decreases as the sample size increases. Tables 3.5 and 3.6 report the average of empirical L_2 risks for cases 6 and 7. As expected, non-robust DNN estimator has explosive risks, which are around three times of those for uncontaminated data. Similar to the 2D cases, either enlarging the contaminated region or the contamination proportion increases risk with DNN estimators. The precision of the RDNN estimator is kept at the same level as all outlier types and the clean dataset. This provides strong evidence that the proposed RDNN estimator is less sensitive to the presence of outliers, maintaining precision. In the worst case, the risks of RDNN estimator has increased no more than four times, however, the non-robust one has increased around 20 times compared with the clean data scenarios.

Table 3.4: Empirical L_2 risk of 3D uncontaminated data with standard errors in brackets.

n	RDNN	DNN
50	0.103 (0.050)	0.090 (0.045)
100	0.055 (0.033)	0.047 (0.023)
200	0.027 (0.013)	0.026 (0.018)

Table 3.5: Empirical L_2 risk of 3D contaminated data for cases 5 with standard errors in brackets.

Contaminated regions	n	$r = 0.1$		$r = 0.2$	
		RDNN	DNN	RDNN	DNN
$[0.10, 0.20]^3$	50	0.111 (0.049)	0.204 (0.066)	0.119 (0.052)	0.515 (0.107)
	100	0.056 (0.028)	0.155 (0.041)	0.078 (0.033)	0.539 (0.067)
	200	0.033 (0.018)	0.148 (0.029)	0.049 (0.017)	0.571 (0.058)
$[0.10, 0.30]^3$	50	0.118 (0.060)	0.463 (0.104)	0.173 (0.055)	1.598 (0.212)
	100	0.066 (0.032)	0.472 (0.092)	0.135 (0.052)	1.925 (0.160)
	200	0.042 (0.017)	0.478 (0.077)	0.103 (0.033)	1.942 (0.156)

3.6 Real data analysis

The dataset used in the preparation of this article were obtained from the ADNI database (`adni.loni.usc.edu`). The ADNI is a longitudinal multicenter study designed to develop clinical, imaging, genetic, and biochemical biomarkers for the early detection and tracking of AD. From this database, we collect PET data from 85 patients in AD group. This PET dataset has been spatially normalized and post-processed. These AD patients have three to six times doctor visits and we only select the PET scans obtained in the third visits. Patients' age ranges from 59 to 88 and average age is 76.49. All scans were reoriented into $79 \times 95 \times 69$ voxels,

Table 3.6: Empirical L_2 risk of 3D contaminated data for cases 6 and 7 with standard errors in brackets.

error types	n	mixture rate$= 0.3$		mixture rate$= 0.5$	
		RDNN	DNN	RDNN	DNN
Cauchy	50	0.130 (0.072)	0.526 (1.421)	0.134 (0.073)	0.804 (2.805)
	100	0.066 (0.035)	0.459 (0.953)	0.062 (0.036)	0.535 (1.295)
	200	0.043 (0.023)	0.163 (0.267)	0.045 (0.026)	0.418 (0.907)
Slash	50	0.128 (0.062)	0.753 (2.220)	0.125 (0.057)	0.787 (1.938)
	100	0.066 (0.042)	0.403 (0.887)	0.068 (0.049)	0.760 (1.458)
	200	0.049 (0.036)	0.321 (0.771)	0.047 (0.030)	0.587 (1.312)

which means each patient has 69 sliced 2D images with 79×95 pixels. For 2D case, it indicates that each subject has $N = 7,505 = 79 \times 95$ observed pixels for each selected image slice.

In this imaging dataset, we observe that there exists a few abnormal observations, which have different pattern from the majority of data. In Figure 3.3, the first row demonstrates the averaged images of the 20-th, 30-th, 40-th, and 50-th slices across all patients. In the second row, images are taken from different individuals, where extreme small values showing in certain regions, which lead to blur boundaries. For the 2D case, we select the 20-th, 30-th, 40-th and 50-th slices from 69 slices for each patient, and apply the proposed RDNN for each slice, respectively, with loss function $\rho_\tau(x) = x\,(\tau - \mathbb{I}(x < 0))$ with $\tau = 0.1, 0.3, 0.5, 0.7, 0.9$. The neural network (2.5) is trained through optimizer `Adam` with architecture parameters (L, p, s) selected as discussed in 3.4.1. We used 100 epochs and 128 as batch size given different data. Based on the images, we obtain the proposed RDNN estimators for each slice, and also recover the image with the original resolution 79×95 pixels and a higher resolution 128×128. To visualize the estimates, Figures 3.4 provides the heat maps of the RDNN estimator of different quantiles for all four slices in 2D scenario, Figure 3.5 depicts the same estimates but with a finer resolution (128×128). For 3D scenario, we combine all the four slices together, hence, the 3D data totally contains $79 \times 95 \times 4$ voxels. We first obtain the RDNN estimators with the original resolution and recover them also in a higher resolution $128 \times 128 \times 4$. Figures 3.6 and 3.7 depict the RDNN estimators in the original resolution and higher resolution for each slice and quantile, respectively. The estimated quantiles serve to confirm the suspected multi-modality in this imaging data. According to the heat maps, in 20-th, 30-th, and 40-th slices, higher quantiles significantly differ from lower ones in that there are much larger value presenting in the bottom regions. In 50-th slice, higher quantiles can be easily distinguished from lower ones in terms of overall larger values and wider boundaries.

3.7 Discussion

In this work, we resolve the robust estimation for functional data on multi-dimensional domains via the promising technique from the deep learning. By properly choosing network architecture, our estimator achieves the optimal nonparametric convergence rate in empirical norm. To

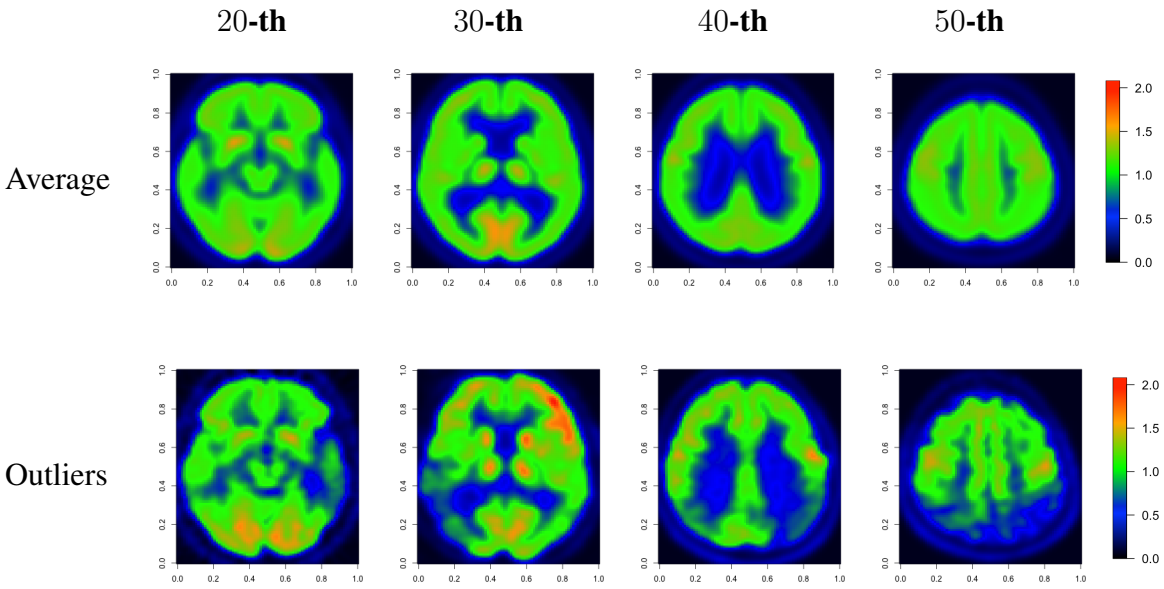

Figure 3.3: The first row are the averaged images for 20-th, 30-th, 40-th and 50-th slices across all patients. The rest are some abnormal data for each slices from some patients.

the best of our knowledge, the present work is the first work on multi-dimensional functional data robust estimation with provable guarantees. Numerical analysis demonstrates that our approach is useful in recovering the signal for imaging data.

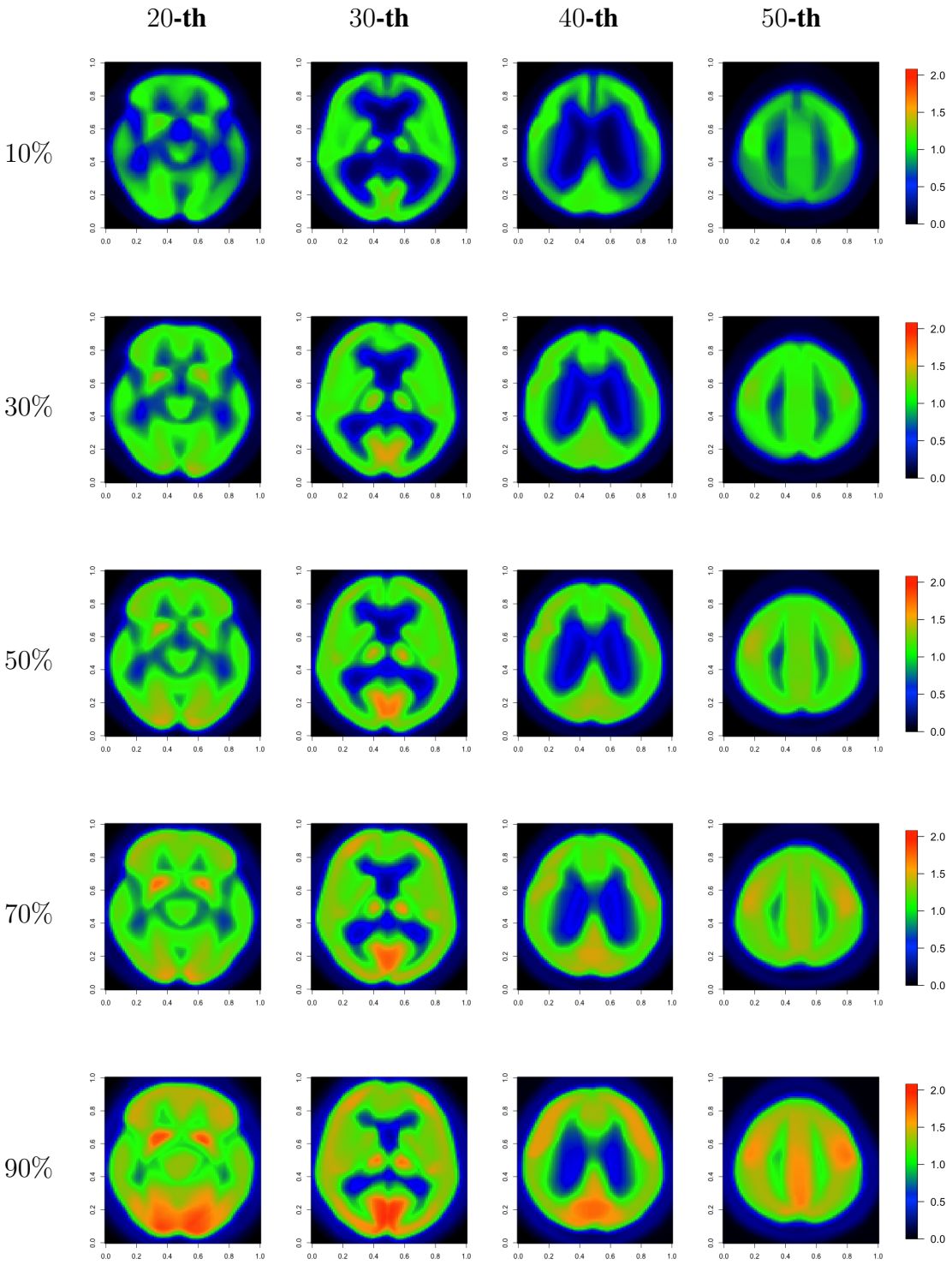

Figure 3.4: 2D quantile esimators with 79×95 pixels. From the left to the right: the 20-th, 30-th, 40-th, and 50-th slices. From the top to the bottom: (10% , 30% , 50% , 70%, 90%)-quantiles.

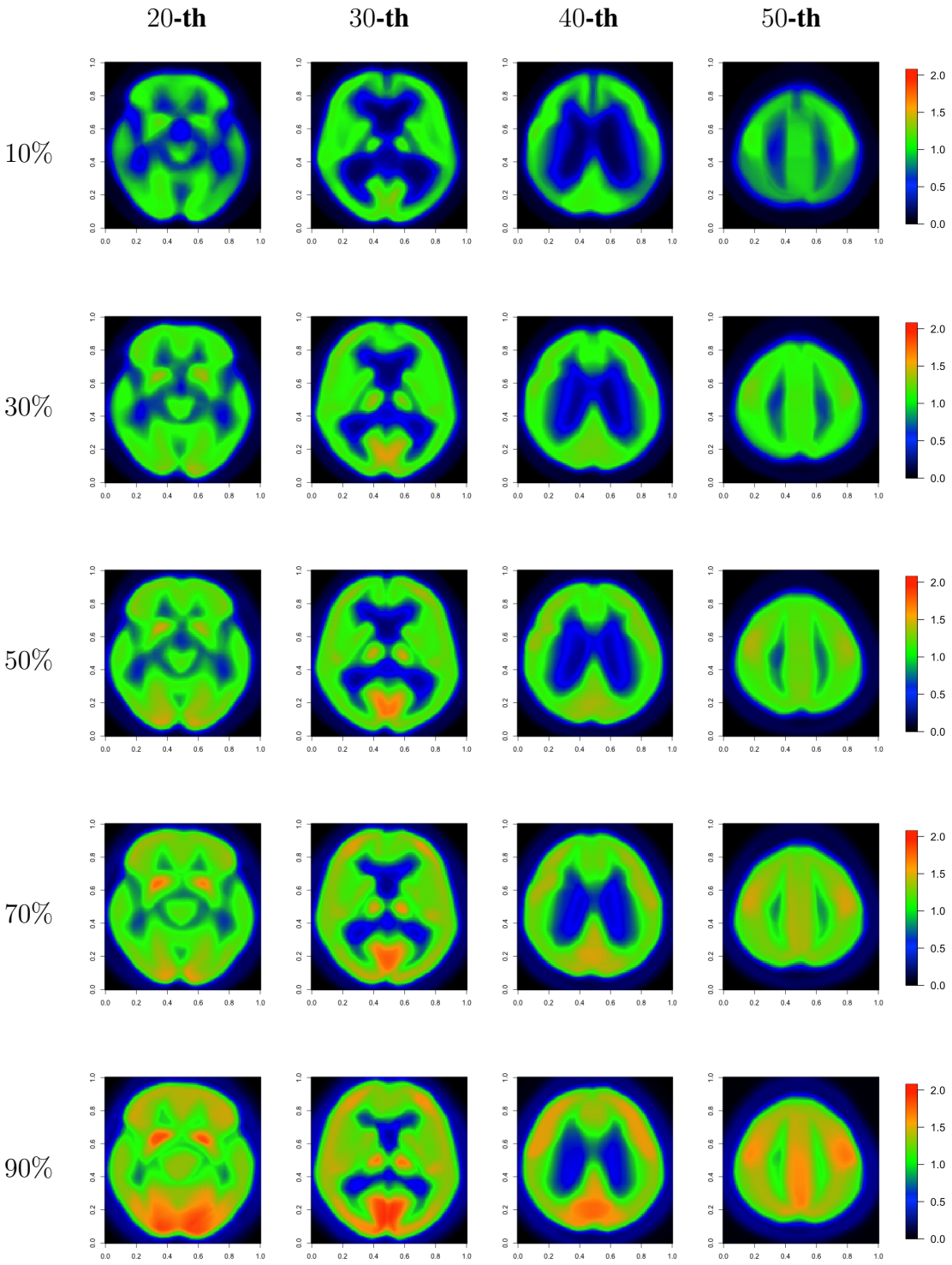

Figure 3.5: 2D quantile esimators with 128×128 pixels. From the left to the right: the 20-th, 30-th, 40-th, and 50-th slices. From the top to the bottom: (10% , 30% , 50% , 70%, 90%)-quantiles.

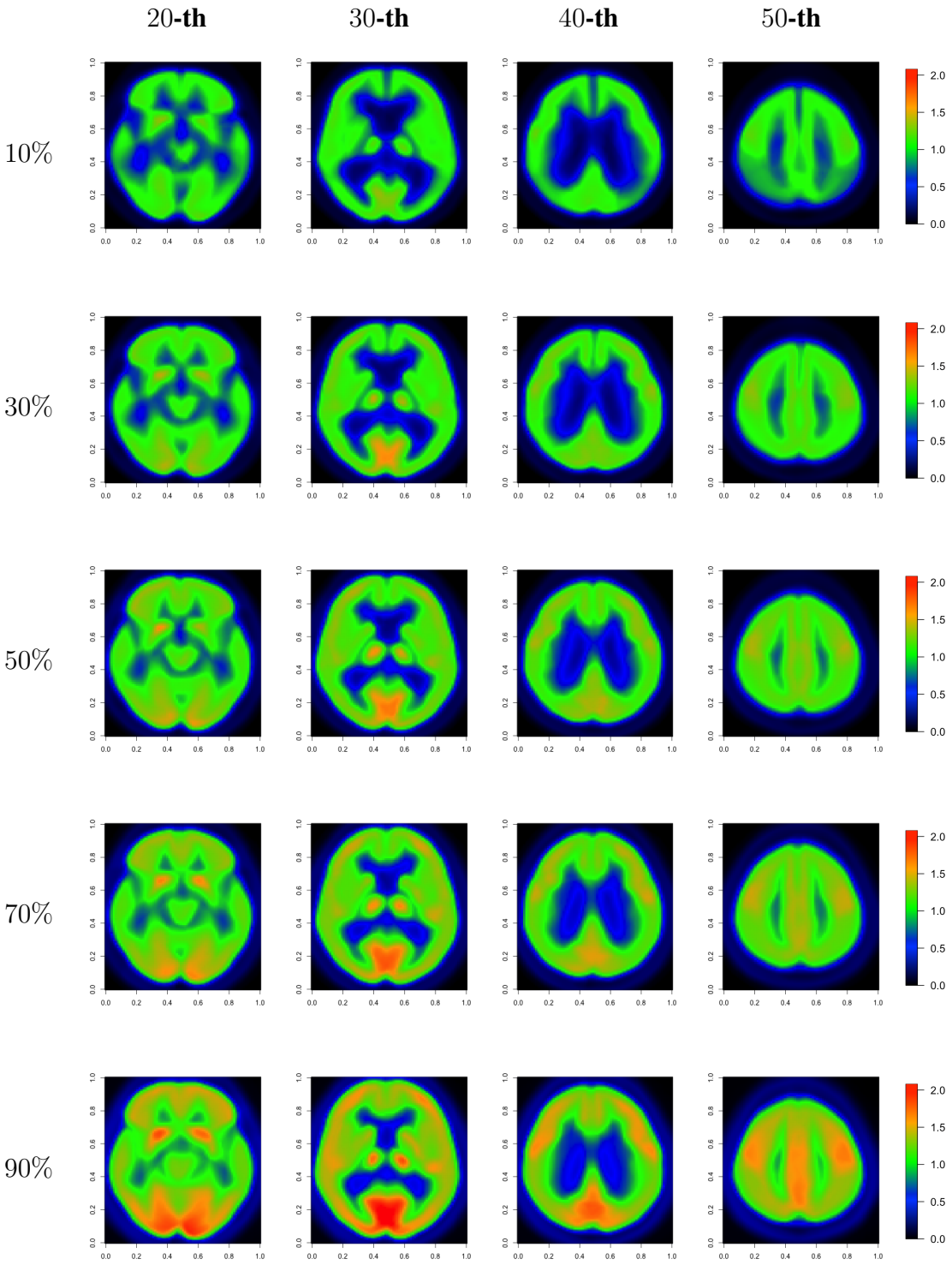

Figure 3.6: 3D quantile estimators with 79×95 pixels. From the left to the right: the 20-th, 30-th, 40-th, and 50-th slices. From the top to the bottom: (10% , 30% , 50% , 70%, 90%)-quantiles.

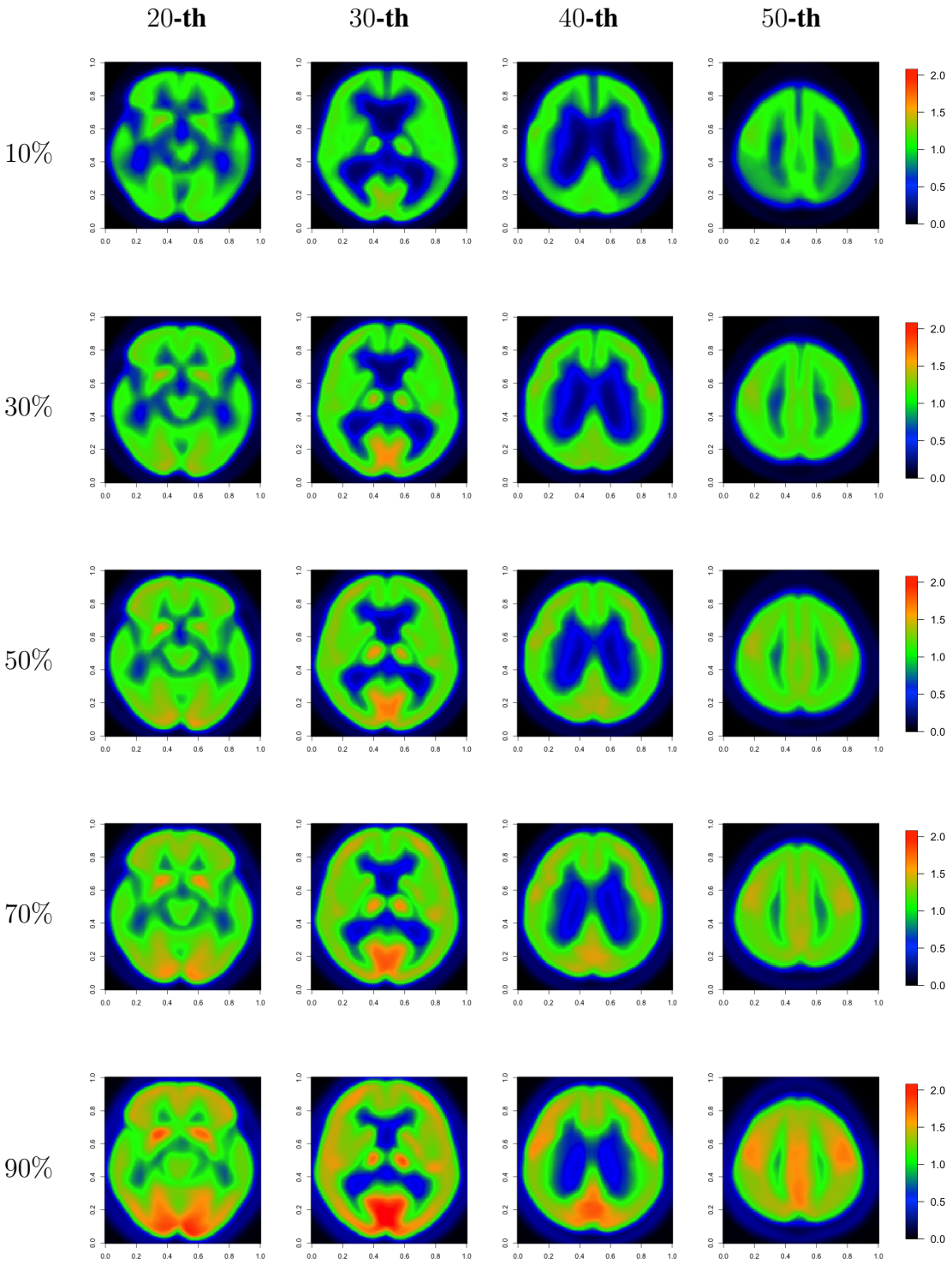

Figure 3.7: 3D quantile estimators with 128×128 pixels. From the left to the right: the 20-th, 30-th, 40-th, and 50-th slices. From the top to the bottom: (10% , 30% , 50% , 70%, 90%)-quantiles.

Chapter 4

Optimal Classification for Functional Data

4.1 Introduction

Functional classification has applications in many areas such as machine learning, chemimetrics and artificial intelligence [98, 60, 90, 23]. A variety of functional classification techniques have been proposed such as logistic regression [5], distance-based classifiers [40, 42], k-nearest neighbor classifiers [13, 14], Bayesian classifiers [62, 30], data depth based classifiers [29, 93], functional linear or quadratic discriminant analysis [95, 85], and projection-based methods [28, 32]. Recent monographs by [80, 115] provide comprehensive and general discussions on this topic.

Existing works on functional classification have been merely focusing on the construction of classifiers that achieve *perfect classification* phenomenon in the sense that the probability of misclassifying a new data function converges to zero asymptotically. For instance, [31] established perfect classification of a linear centroid classifier and suggested a practical representation using components obtained from functional principal component analysis and partial least squares; [30] proposed to use density ratios of projections on a sequence of eigenfunctions that are common to the two populations and showed the perfect classification property; [33] studied perfect classification property when the data functions are observed on different domains; [12] further clarified the near-perfect classification phenomenon in a reproducing kernel Hilbert space framework for Gaussian processes. As revealed by [30], perfect classification is only achievable when the two populations are sufficiently separated from each other in the

sense that the infinite series characterizing the distance between the mean functions or covariance functions is divergent. When both series are convergent, perfect classification is no longer achievable which we call it as the *imperfect classification* problem. In imperfect classification, the risk of any classifier, including the optimal Bayes classifier, does not tend to zero. Therefore, the theory of perfect classification is no longer valid.

In this chapter, we study functional classification problem through minimax optimality which will address the above issue. Specifically, we derive sharp convergence rates of minimax excess risk (MER) for classifying Gaussian functional data. MER framework is able to accommodate imperfect classification since it allows for nonvanishing classification risk. Moreover, the derived optimal rates for MER may serve as a gold standard to evaluate the performance of any functional classifiers.

Discriminant analysis is one of the most popular classification techniques in statistics and machine learning due to its simplicity and effectiveness, therefore, it would be interesting to investigate whether it is able to achieve minimax optimality. In low-dimensional regime, linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) have been well studied in both theoretical and applied aspects [3]. In high-dimensional regime, despite recent methodological development, there has been relatively few fundamental studies on minimax theory for discriminant analysis. Recently, [19, 18] developed minimax theory for LDA and QDA in high-dimensional setting by imposing sparsity assumptions on discriminating directions. Extension of discriminant analysis to functional data appears to be challenging due to the infinite-dimensional feature of the data. When data are fully observed, [42] proposed a closely related functional quadratic method for discriminating two general Gaussian populations by making use of a suitably defined Mahalanobis distance for functional data, and [31] considered Gaussian populations with equal covariance using functional linear discriminant. However, minimax optimality of these techniques is unclear. The present chapter will demonstrate the minimax optimality of discriminant analysis techniques in imperfect classification where data are either fully or discretely observed. The latter scenario is practically meaningful, since in real-world problems, functional data can only be observed at discrete sampling points.

We will investigate MER in both scenarios when data functions are Gaussian and propose computationally efficient classifiers that achieve minimax optimality.

Our main idea is to project the functional data onto an orthogonal system that is common in both populations, then to construct the so-called Functional Quadratic Discriminant Analysis (FQDA) classifiers based on the projection scores. When data are fully observed, we derive an explicit upper bound for the excess risk of the proposed FQDA classifiers. Moreover, we establish a lower bound for the MER which matches the upper bound, demonstrating the minimax optimality of our method. We extend these results to the scenario of discretely observed data in which the rate of MER demonstrates a phase transition phenomenon jointly characterized by the number of data curves and sampling frequency. Our analysis reveals that when sampling frequency is relatively small, the number of data curves has little effect on the rate of MER. When sampling frequency is relatively large, the rate of MER more depends on the number of data curves. In other words, there exists a critical sampling frequency that governs the performance of the minimax optimal classifier. In functional data estimation, existence of a critical sampling frequency has been discovered by [17]. The present work has made a relevant discovery in functional data classification. Both simulation and real data studies are carried out to demonstrate the performance of our FQDA classifiers

The rest of the chapter is organized as follows. Section 4.2 presents in detail the data-driven classification procedure FQDA. Theoretical properties of FQDA are investigated in Section 4.3. The upper and lower bounds together show that the FQDA rule achieves the optimal rate in terms of the classification error. The corresponding classifier are investigated. Simulation studies are given in Section 4.4 where we compare the performance of the proposed algorithms to other existing classification methods in the literature. In addition, the merits of the FQDA and FQDA classifiers are illustrated through an analysis of the speech recognition dataset in Section 4.5. We summarize the proposed methodology and discuss the future work in Section 4.6. Major technical details for the proofs of main results are included in the Appendix.

Notation and Terminologies. We introduce basic notations and definitions that will be used throughout the rest of the chapter and Chapter C.7. Vectors and matrices are denoted by boldface letters. For a matrix $\mathbf{A} \in \mathbb{R}_{p\times p}$, $|\mathbf{A}|$ is the determinant of $\mathbf{A}$, and $\mathbf{I}_p$ is the $p \times p$

identity matrix. For two sequences of positive numbers a_n and b_n, $a_n \lesssim b_n$ means that for some constant $c > 0$, $a_n \leq cb_n$ for all n, and $a_n \asymp b_n$ means $a_n \lesssim b_n$ and $b_n \lesssim a_n$. $a_n \ll b_n$ means $\lim_{n\to\infty} |a_n|/|b_n| = 0$. We also use $c, c_1, c_2, \ldots, C, C_1, C_2, \ldots$ to denote absolute constants whose values may vary from place to place.

4.2 Model assumptions and functional quadratic discriminant analysis

4.2.1 Model assumptions and oracle QDA

Let $Z(t), t \in \mathcal{T}$ be a random function defined on a compact interval $\mathcal{T}$ which belongs to class k if $Z \sim \mathcal{GP}(\eta_k, \Omega_k)$ for $k = 1, 2$, where $\mathcal{GP}(\eta_k, \Omega_k)$ is a Gaussian process model with unknown mean function $\eta_k(\cdot)$ and unknown covariance function $\Omega_k(\cdot, \cdot)$. Let $L(Z)$ denote the class label of Z with probability distribution $P(L(Z) = k) = \pi_k$, where $\pi_k \in (0, 1)$ are unknown satisfying $\pi_1 + \pi_2 = 1$. When $L(Z) = k$, suppose that Z admits a basis expansion $Z(t) = \sum_{j=1}^{\infty} \xi_j^{(k)} \psi_j(t)$, where $\{\psi_j\}_{j\geq 1}$ is an orthonormal basis in $L^2(\mathcal{T})$ and $\xi_j^{(k)}, j \geq 1$ are uncorrelated projection scores. Hence,

$$\eta_k(t) = \sum_{j=1}^{\infty} \mu_{kj}\psi_j(t) \text{ and } \Omega_k(t, s) = \sum_{j=1}^{\infty} \lambda_j^{(k)} \psi_j(t)\psi_j(s), \ t, s \in \mathcal{T},$$

where $\mu_{kj} = E\xi_j^{(k)}$ is the j-th mean projection score and $\lambda_j^{(k)} = Var(\xi_j^{(k)})$ is the j-th eigenvalue. For $J \geq 1$ and $k = 1, 2$, let $\boldsymbol{z} = (z_1, \ldots, z_J)^\top$, $\boldsymbol{\mu}_k = (\mu_{k1}, \ldots, \mu_{kJ})^\top$, $\boldsymbol{\Sigma}_k = \text{diag}\left(\lambda_1^{(k)}, \ldots, \lambda_J^{(k)}\right)$, and $\boldsymbol{\theta} = (\pi_1, \pi_2, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2)$. For predicting $L(Z)$, the *oracle QDA* classifier is defined by

$$G_{\boldsymbol{\theta}}^*(Z) = \begin{cases} 1, & Q(\boldsymbol{z}; \boldsymbol{\theta}) \geq 0, \\ 2, & Q(\boldsymbol{z}; \boldsymbol{\theta}) < 0, \end{cases} \tag{4.1}$$

where $Q(\boldsymbol{z}; \boldsymbol{\theta})$ is the discriminant function defined by

$$Q(\boldsymbol{z}; \boldsymbol{\theta}) \ = \ (\boldsymbol{z} - \boldsymbol{\mu}_1)^\top \mathbf{D}(\boldsymbol{z} - \boldsymbol{\mu}_1) - 2\boldsymbol{\delta}^\top \boldsymbol{\Sigma}_2^{-1}(\boldsymbol{z} - \bar{\boldsymbol{\mu}}) - \log(|\boldsymbol{\Sigma}_1|/|\boldsymbol{\Sigma}_2|) + 2\log(\pi_1/\pi_2),$$

with $\bar{\boldsymbol{\mu}} = \frac{\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2}{2}$, $\boldsymbol{\delta} = \boldsymbol{\mu}_2 - \boldsymbol{\mu}_1$, and $\mathbf{D} = \boldsymbol{\Sigma}_2^{-1} - \boldsymbol{\Sigma}_1^{-1}$. The performance of (4.1) depends on the discriminating direction $\boldsymbol{\Sigma}_2^{-1/2}\boldsymbol{\delta}$ and the differential graph $\mathbf{D}$. The necessity of considering

the covariance difference for functional data analysis has been shown by [20]. If $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$, the quadratic classification boundary in (4.1) becomes linear, and (4.1) degenerates to LDA. In this work, we focus on QDA for functional data which allows different covariances.

To obtain a consistent estimator for the oracle rule $G^*_{\boldsymbol{\theta}}$, we begin by noting an important observation that $\log(|\mathbf{\Sigma}_1|/|\mathbf{\Sigma}_2|) = \log(|\mathbf{D}\mathbf{\Sigma}_1 + \mathbf{I}_J|)$, and rewriting $Q(\boldsymbol{z};\boldsymbol{\theta})$ as

$$Q(\boldsymbol{z};\boldsymbol{\theta}) = (\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \mathbf{D}(\boldsymbol{z}-\boldsymbol{\mu}_1) - 2\boldsymbol{\beta}^\top(\boldsymbol{z}-\bar{\boldsymbol{\mu}}) - \log(|\mathbf{D}\mathbf{\Sigma}_1 + \mathbf{I}_J|) + 2\log(\pi_1/\pi_2), \quad (4.2)$$

where $\boldsymbol{\beta} = \mathbf{\Sigma}_2^{-1}\boldsymbol{\delta}$. A simple but essential observation of (4.2) is that the first three quantities in (4.2) depends on either $\mathbf{D}$ or $\boldsymbol{\beta}$, and the fourth term $\log(\pi_1/\pi_2)$ represents the log odds ratio, and all terms will be estimated in next sections.

4.2.2 FQDA for fully observed functional data

Suppose we observe a training sample $\{X_i^{(k)}(t) : 1 \le i \le n_k, k = 1, 2\}$, where n_k is the sample size for class k, $X_i^{(k)}(t) \overset{i.i.d.}{\sim} \mathcal{GP}(\eta_k, \Omega_k)$ and all $X_i^{(k)}(t)$'s and $Z(t)$ are independent. We first consider the ideal case where the entire function curves $X_i^{(k)}(t)$ are fully observed. Recall the basis expansion $X_i^{(k)}(t) = \sum_{j=1}^{\infty} \xi_{ij}^{(k)}\psi_j(t)$ with $E(\xi_{ij}^{(k)}) = \mu_{kj}$ and $Var(\xi_{ij}^{(k)}) = \lambda_j^{(k)}$. We estimate the mean vector, differential graph and discriminant vector as follows:

$$\begin{aligned}
\widehat{\boldsymbol{\mu}}_k &= (\bar{\xi}_{\cdot 1}^{(k)}, \ldots, \bar{\xi}_{\cdot J}^{(k)})^\top, \\
\widehat{\mathbf{D}} &= \widehat{\mathbf{\Sigma}}_2^{-1} - \widehat{\mathbf{\Sigma}}_1^{-1}, \qquad (4.3) \\
\widehat{\boldsymbol{\beta}} &= \widehat{\mathbf{\Sigma}}_2^{-1}(\widehat{\boldsymbol{\mu}}_1 - \widehat{\boldsymbol{\mu}}_2), \qquad (4.4)
\end{aligned}$$

where $\bar{\xi}_{\cdot j}^{(k)} = {n_k}^{-1}\sum_{i=1}^{n_k}\xi_{ij}^{(k)}$, $\widehat{\mathbf{\Sigma}}_k = \text{diag}\left(\widehat{\lambda}_1^{(k)}, \ldots, \widehat{\lambda}_J^{(k)}\right)$, $\widehat{\lambda}_j^{(k)} = n_k^{-1}\sum_{i=1}^{n_k}(\xi_{ij}^{(k)} - \bar{\xi}_{\cdot j}^{(k)})^2$. We then propose the following classification rule, called as FQDA:

$$\widehat{G}_J^{FQDA}(Z) = \begin{cases} 1, & \widehat{Q}(\boldsymbol{z},\boldsymbol{\theta}) \ge 0, \\ 2, & \widehat{Q}(\boldsymbol{z},\boldsymbol{\theta}) < 0, \end{cases} \quad (4.5)$$

where

$$\widehat{Q}(\boldsymbol{z},\boldsymbol{\theta}) := (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top \widehat{\mathbf{D}}(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1) - 2\widehat{\boldsymbol{\beta}}^\top(\boldsymbol{z}-\widehat{\bar{\boldsymbol{\mu}}}) - \log\left(|\widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1 + \mathbf{I}_J|\right) + 2\log\left(\widehat{\pi}_1/\widehat{\pi}_2\right),$$

$\widehat{\bar{\boldsymbol{\mu}}} = (\widehat{\boldsymbol{\mu}}_1 + \widehat{\boldsymbol{\mu}}_2)/2$ and $\widehat{\pi}_k = n_k/(n_1+n_2)$ is the sample proportion of class k. The tuning parameter J is chosen to minimize the cross-validation estimator of classification error [32]. In Section 4.3, the desirable asymptotic property of (4.5) is presented.

4.2.3 FQDA for discretely observed functional data

Consider a more practical scenario where each data curve can only be observed on M evenly spaced discrete sampling points $\{t_1, t_2, \ldots, t_M\} \subset \mathcal{T}$. Choose $1 \le J \le M$ and define

$$\mathbf{B} = \begin{pmatrix} \psi_1(t_1) & \psi_2(t_1) & \cdots & \psi_J(t_1) \\ \psi_1(t_2) & \psi_2(t_2) & \cdots & \psi_J(t_2) \\ \vdots & \vdots & & \vdots \\ \psi_1(t_M) & \psi_2(t_M) & \cdots & \psi_J(t_M) \end{pmatrix}.$$

Heuristically, when J is suitably large, the data vector $\mathbf{X}_i^{(k)} = (X_i^{(k)}(t_1), \ldots, X_i^{(k)}(t_M))^\top$ has an approximate expression $\mathbf{X}_i^{(k)} \approx \mathbf{B}\boldsymbol{\mu}_k$ for $i = 1, \ldots, n_k$. For technical convenience, we focus on the special case that ψ_j's are Fourier basis or Haar wavelet basis so that $\mathbf{B}^\top\mathbf{B} = \mathbf{I}_J$. This leads to $\boldsymbol{\mu}_k \approx \boldsymbol{\zeta}_i^{(k)} := \mathbf{B}^\top\mathbf{X}_i^{(k)}$. Therefore, we propose the following estimator of $\boldsymbol{\mu}_k$ as well as the estimators of the differential graph and the discriminating direction:

$$\begin{aligned} \widehat{\boldsymbol{\mu}}_{sk} &= \frac{1}{n_k}\sum_{i=1}^{n_k} \boldsymbol{\zeta}_i^{(k)}, \\ \widehat{\mathbf{D}}_s &= \widehat{\boldsymbol{\Sigma}}_{s2}^{-1} - \widehat{\boldsymbol{\Sigma}}_{s1}^{-1}, \qquad (4.6) \\ \widehat{\boldsymbol{\beta}}_s &= \widehat{\boldsymbol{\Sigma}}_{s2}^{-1}\left(\widehat{\boldsymbol{\mu}}_{s2} - \widehat{\boldsymbol{\mu}}_{s1}\right), \qquad (4.7) \end{aligned}$$

where $\widehat{\boldsymbol{\Sigma}}_{sk} = \mathrm{diag}\left(\widehat{\lambda}_{s1}^{(k)}, \ldots, \widehat{\lambda}_{sJ}^{(k)}\right)$ with $\widehat{\lambda}_{sj}^{(k)} = n_k^{-1}\sum_{i=1}^{n_k}\left(\zeta_{ij}^{(k)} - \bar{\zeta}_{\cdot j}^{(k)}\right)^2$, $\bar{\zeta}_{\cdot j}^{(k)} = n_k^{-1}\sum_{i=1}^{n_k}\zeta_{ij}^{(k)}$, and $\zeta_{ij}^{(k)}$'s are components of $\boldsymbol{\zeta}_i^{(k)}$. We then propose the following classification rule, called as

sampling FQDA (sFQDA):

$$\widehat{G}_J^{sFQDA}(Z) = \begin{cases} 1, & \widehat{Q}_s(\boldsymbol{z}, \boldsymbol{\theta}) \geq 0, \\ 2, & \widehat{Q}_s(\boldsymbol{z}, \boldsymbol{\theta}) < 0, \end{cases} \tag{4.8}$$

where

$$\widehat{Q}_s(\boldsymbol{z}, \boldsymbol{\theta}) := (\boldsymbol{z} - \widehat{\boldsymbol{\mu}}_{s1})^\top \widehat{\mathbf{D}}_s (\boldsymbol{z} - \widehat{\boldsymbol{\mu}}_{s1}) - 2\widehat{\boldsymbol{\beta}}_s^\top (\boldsymbol{z} - \widehat{\bar{\boldsymbol{\mu}}}_s) - \log\left(|\widehat{\mathbf{D}}_s \widehat{\boldsymbol{\Sigma}}_{s1} + \mathbf{I}_J|\right) + 2\log\left(\widehat{\pi}_1/\widehat{\pi}_2\right),$$

with $\widehat{\bar{\boldsymbol{\mu}}}_s = (\widehat{\boldsymbol{\mu}}_{s1} + \widehat{\boldsymbol{\mu}}_{s2})/2$. In the subsequent sections, we shall show that (4.8) has desirable theoretical properties and finite-sample performance.

4.3 Theoretical properties

In this section, we derive sharp convergence rate for MER when data curves are either fully observed or discretely observed. We also show that the proposed FQDA and sFQDA classifiers are able to achieve the minimax optimal rates. All results are derived in imperfect classification scenario.

With a slight abuse of notation, let $\boldsymbol{\theta} = (\pi_1, \pi_2, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2)$ in which $\boldsymbol{\mu}_k = (\mu_{k1}, \mu_{k2}, \ldots)$ is the infinite sequence of mean projection scores and $\boldsymbol{\Sigma}_k$ is a diagonal linear operator from $L^2(\mathcal{T})$ to $L^2(\mathcal{T})$ satisfying $\boldsymbol{\Sigma}_k \psi_j = \lambda_j^{(k)} \psi_j$ for $j \geq 1$ and $k = 1, 2$. Given $\boldsymbol{\theta}$, it follows by [12] and [106] that the optimal Bayes classification rule for classifying a new data function $Z \in L^2(\mathcal{T})$ has an expression

$$G_{\boldsymbol{\theta}}^*(Z) = \begin{cases} 1, & Q^*(Z, \boldsymbol{\theta}) \geq 0, \\ 2, & Q^*(Z, \boldsymbol{\theta}) < 0, \end{cases}$$

where

$$Q^*(Z, \boldsymbol{\theta}) = \langle \mathbf{D}(Z - \eta_1), Z - \eta_1 \rangle - 2\langle \boldsymbol{\Sigma}_2^{-1}(\eta_2 - \eta_1), Z - \bar{\eta} \rangle - \log\det\left(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\right) + \log\left(\frac{\pi_1}{\pi_2}\right),$$

with $\bar{\eta} = (\eta_1 + \eta_2)/2$, $\mathbf{D} = \boldsymbol{\Sigma}_2^{-1} - \boldsymbol{\Sigma}_1^{-1}$ (difference of inverse operators), $\langle \cdot, \cdot \rangle$ being the usual L^2 inner product, and

$$\det\left(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\right) = \exp\left\{\sum_{s=1}^{\infty} \frac{(-1)^{s-1}}{s} \mathrm{Tr}\left(\left[\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1 - \mathrm{id}\right]^s\right)\right\}$$

being the infinite determinant of $\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1$, often called the Plemelj's formula (see [97]), and is convergent if $\mathrm{Tr}(|\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1 - \mathrm{id}|) < \infty$. Indeed, $Q^*(Z, \boldsymbol{\theta})$ is well-defined as long as both series $\sum_{j=1}^{\infty} (\mu_{1j} - \mu_{2j})^2/\lambda_j^{(2)}$ and $\sum_{j=1}^{\infty} \left(\lambda_j^{(1)}/\lambda_j^{(2)} - 1\right)^2$ are convergent (see Lemma 3 of [30]).

4.3.1 Fully observed case

Suppose that the data functions are fully observed as in Section 4.2.2. Consider the following parameter space:

$$\begin{aligned}\Theta \;=\; & \left\{\boldsymbol{\theta} : \sum_{j=1}^{\infty} \lambda_j^{(k)} \le C_0, \sum_{j=1}^{\infty} \mu_{kj}^2 \le C_0, \sum_{j=1}^{\infty} (\mu_{1j} - \mu_{2j})^2/\lambda_j^{(2)} \le C_1, \right. \\ & \left. \sum_{j=1}^{\infty} \left(\lambda_j^{(1)}/\lambda_j^{(2)} - 1\right)^2 \le C_2, C_3 \le \pi_1, \pi_2 \le 1 - C_3 \right\}, \end{aligned} \tag{4.9}$$

where C_0, C_1, C_2, C_3 are absolute constants with $C_3 \in (0, 1/2)$. Let us provide some insights on Θ. Assumption $\sum_{j=1}^{\infty} \lambda_j^{(k)} \le C_0$ implies that the covariance function Ω_k is uniformly bounded. Assumption $\sum_{j=1}^{\infty} \mu_{kj}^2 \le C_0$ implies $\eta_k \in L^2(\mathcal{T})$. Assumption $\sum_{j=1}^{\infty} (\mu_{1j} - \mu_{2j})^2/\lambda_j^{(2)} \le C_1$ and $\sum_{j=1}^{\infty} \left(\lambda_j^{(1)}/\lambda_j^{(2)} - 1\right)^2 \le C_2$ characterizes the closeness of the two populations. Note that in [30, 33], either of the two series is required being divergent so that the optimal Bayes risk is asymptotically vanishing and perfect classification is achieved. They proposed classifiers which are proven to achieve perfect classification. In our case, both series are convergent implying that the two populations are much closer than the ones in [30, 33], therefore, optimal Bayes risk does not go to zero and perfect classification is impossible. This imposes additional challenge in finding correct classification.

For $\boldsymbol{\theta} \in \Theta$ and a generic classifier $\widehat{G}$, let $R_{\boldsymbol{\theta}}(\widehat{G})$ be the classification risk for $\widehat{G}$ under $\boldsymbol{\theta}$:

$$R_{\boldsymbol{\theta}}(\widehat{G}) = E_{\boldsymbol{\theta}}[\mathbb{I}\{\widehat{G}(Z) \neq L(Z)\}].$$

For $J \geq 0$, define

$$g(J;\Theta) = \sup_{\boldsymbol{\theta}\in\Theta} \left\{ \sum_{j=J+1}^{\infty} \frac{(\mu_{1j}-\mu_{2j})^2}{\lambda_j^{(2)}} + \sum_{j=J+1}^{\infty} \left(\lambda_j^{(1)}/\lambda_j^{(2)} - 1\right)^2 \right\}.$$

Without loss of generality, assume $g(J;\Theta) \leq 1$ for any $J \geq 0$; otherwise one can scale $g(J;\Theta)$ by $g(0;\Theta)$. Let J^* be the unique solution to $\frac{J^*\log n}{n} = g(J^*;\Theta)$ with $n := \min\{n_1, n_2\}$. Theorem 4.1 provides an upper bound for excess risk of $\widehat{G}_J^{FQDA}$ when $J = J^*$.

Theorem 4.1. *Consider the parameter space Θ. Then the proposed FQDA classifier (4.5) satisfies*

$$\sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}_{J^*}^{FQDA}) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*)\right] \lesssim \frac{J^*\log n}{n}.$$

In the following theorem, we provide a lower bound for the MER which matches the above upper bound, thus, the optimality of FQDA is verified.

Theorem 4.2. *Consider the parameter space Θ. Then the MER over Θ satisfies*

$$\inf_{\widehat{G}} \sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*)\right] \gtrsim \frac{J^*\log n}{n},$$

where the infimum is taken over all possible classifiers.

The results in Theorems 4.1 and 4.2 together garantee that $\widehat{G}_{J^*}^{FQDA}$ is able to mimic $G_{\boldsymbol{\theta}}^*$ consistently over the parameter space Θ. Both of the lower and upper bounds show that the proposed FQDA rule is optimal for classifying Gaussian functional data under mild regularity conditions. No other method can achieve a faster MER rate than the proposed FQDA classifier. To the best of our knowledge, they are the first theorems providing sharp convergence rate for MER in imperfect classification.

In general, J^* has no explicit expression so the rates in Theorems 4.1 and 4.2 are abstract. To gain more insights, we consider a particular parameter space that enables us to derive a more

concrete result. Specifically, consider the following parameter space,

$$\Theta(\alpha) = \left\{ \boldsymbol{\theta} : \sum_{j=1}^{\infty} \lambda_j^{(k)} \leq C_0, \sum_{j=1}^{\infty} \mu_{kj}^2 \leq C_0, C_4 j^{-a} \leq (\lambda_j^{(1)}/\lambda_j^{(2)} - 1)^2 \leq C_5 j^{-a}, \right.$$
$$\left. C_6 j^{-b} \leq (\mu_{j1} - \mu_{j2})^2/\lambda_j^{(2)} \leq C_7 j^{-b}, a, b > \alpha, C_3 \leq \pi_1, \pi_2 \leq 1 - C_3 \right\},$$

where $\alpha \geq 1$ characterizes the closeness of the two populations and C_0, C_4, C_5, C_6, C_7 and $0 < C_3 < 1/2$ are absolute constants. Note that $\Theta(\alpha)$ is a subset of Θ, and Proposition 4.1 below provides a concrete result under $\Theta(\alpha)$.

Proposition 4.1. *Consider the parameter space* $\Theta(\alpha)$.

i) The proposed FQDA classification rule in (4.5) satisfies

$$\sup_{\boldsymbol{\theta} \in \Theta(\alpha)} E\left[R_{\boldsymbol{\theta}}(\widehat{G}_{J^*}^{FQDA}) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*) \right] \lesssim \left(\frac{\log n}{n} \right)^{1-1/\alpha},$$

where $J^* = \left(\frac{n}{\log n} \right)^{1/\alpha}$.

ii) The MER over $\Theta(\alpha)$ *satisfies*

$$\inf_{\widehat{G}} \sup_{\boldsymbol{\theta} \in \Theta(\alpha)} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*) \right] \gtrsim \left(\frac{\log n}{n} \right)^{1-1/\alpha},$$

where the infimum is taken over all possible classifiers.

4.3.2 Discretely observed case

Suppose that the data functions are discretely observed as in Section 4.2.3. For $M \geq 0$, define

$$f_1(M) = \sup_{\boldsymbol{\theta} \in \Theta} \left[\sum_{j=M+1}^{\infty} (\mu_{1j}^2 \vee \mu_{2j}^2) \right]^{1/2} \quad \text{and} \quad f_2(M) = \sup_{\boldsymbol{\theta} \in \Theta} \sum_{j=M+1}^{\infty} (\lambda_j^{(1)} \vee \lambda_j^{(2)}),$$

where Θ is provided in (4.9). Without loss of generality, assume $f_1(M) \leq 1$ and $f_2(M) \leq 1$ for any $M \geq 0$; otherwise one can scale them by $f_1(0)$ and $f_2(0)$, respectively. Both $f_1(M)$ and $f_2(M)$ are decreasing in M which depict the decay rate of μ_{kj} and $\lambda_j^{(k)}$.

We now show that when data are discretely observed, the proposed sFQDA classification rule in (4.8) can mimic $G^*_{\boldsymbol{\theta}}$ consistently over the parameter space Θ. Define M^* as the unique solution to $\frac{\log n}{n} = f(M)$, where $f(M) = f_1^2(M) \vee f_2^2(M)$ for each $M \geq 1$. When $M \leq M^*$, define $J_1^* \equiv J_1^*(M,n) \leq M$ to be the unique solution to $f(M) = g(J;\Theta)/J$, and when $M > M^*$, define $J_2^* \equiv J_2^*(M,n) \leq M$ to be the unique solution to $\log n/n = g(J;\Theta)/J$. We shall show the existence and uniqueness of such J_1^* and J_2^* in Lemma C.16 of Appendix.

Theorem 4.3. *Consider the parameter space Θ. The sFQDA in (4.8) satisfies the following.*

i) When $M \leq M^*$,

$$\sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}^{sFQDA}_{J_1^*}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})\right] \lesssim J_1^* f(M);$$

ii) When $M > M^*$,

$$\sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}^{sFQDA}_{J_2^*}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})\right] \lesssim \frac{J_2^* \log n}{n}.$$

Theorem 4.3 provides upper bounds for the MER of $\widehat{G}^{sFQDA}_J$ when $J = J_1^*$ and $J = J_2^*$ corresponding to $M \leq M^*$ and $M > M^*$, respectively. The upper bounds are dramatically different in the two regimes. In the following theorem, we obtain lower bounds for the MER in both regimes which match the upper bounds given in Theorem 4.3. Therefore, the proposed sFQDA classifier is proven minimax optimal.

Theorem 4.4. *Consider the parameter space Θ. The MER over Θ satisfies*

i) when $M \leq M^*$,

$$\inf_{\widehat{G}} \sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})\right] \gtrsim J_1^* f(M);$$

ii) when $M > M^*$,

$$\inf_{\widehat{G}} \sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})\right] \gtrsim \frac{J_2^* \log n}{n},$$

where the infimum is taken over all possible classifiers $\widehat{G}$.

To provide an explicit rate for the MER, we consider the following parameter space

$$\begin{aligned}\Theta(c,d,\alpha) \;=\; & \Big\{\boldsymbol{\theta} : \sum_{j=1}^{\infty}\lambda_j^{(k)} \le C_0, \sum_{j=1}^{\infty}\mu_{kj}^2 \le C_0, C_4 j^{-a} \le (\lambda_j^{(1)}/\lambda_j^{(2)} - 1)^2 \le C_5 j^{-a}, \\ & C_6 j^{-b} \le (\mu_{j1}-\mu_{j2})^2/\lambda_j^{(2)} \le C_7 j^{-b}, a, b > \alpha, \mu_{1j} \vee \mu_{2j} \asymp j^{-c'}, \\ & \lambda_j^{(1)} \vee \lambda_j^{(2)} \asymp j^{-d'}, c' > c, d' > d, C_3 \le \pi_1, \pi_2 \le 1 - C_3\Big\},\end{aligned}$$

where $c \ge 1/2$, $d \ge 1$, $\alpha \ge \varsigma$, such that $\varsigma = \min(2c-1, 2d-2)$, $C_0, C_4, C_5, C_6, C_7 > 0$ are constants, $C_3 \in (0, 1/2)$. Note that $\Theta(c,d,\alpha)$ is a subset of Θ. When restricted to the space $\Theta(c,d,\alpha)$, we have the following more specific rate for the MER of sFQDA. Define $J^* = M^{\varsigma/\alpha}\mathbb{I}(M < M^*) + (n/\log n)^{1/\alpha}\,\mathbb{I}(M \ge M^*)$, where $M^* = (n/\log n)^{1/\varsigma}$ and $\mathbb{I}(\cdot)$ is the indicator function.

Proposition 4.2. *Consider the parameter space $\Theta(c,d,\alpha)$. Then sFQDA in (4.8) satisfies*

$$\sup_{\boldsymbol{\theta}\in\Theta(c,d,\alpha)} E\left[R_{\boldsymbol{\theta}}(\widehat{G}_{J^*}^{sFQDA}) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*)\right] \lesssim \left(\frac{\log n}{n} + \frac{1}{M^{\varsigma}}\right)^{1-1/\alpha},$$

and

$$\inf_{\widehat{G}}\sup_{\boldsymbol{\theta}\in\Theta(c,d,\alpha)} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*)\right] \gtrsim \left(\frac{\log n}{n} + \frac{1}{M^{\varsigma}}\right)^{1-1/\alpha},$$

where the infimum is taken over all possible classifiers.

By Proposition 4.2, the critical sampling frequency for the MER over the space $\Theta(c,d,\alpha)$ is M^*. When $M > M^*$, the MER is of rate $(\log n/n)^{1-1/\alpha}$ which is free of the sampling frequency and is consistent with the rate derived in Proposition 4.1. In other words, when $M > M^*$, increasing sampling frequency will not boost the performance of sFQDA, which performs equally well as FQDA when data functions are fully observed. When $M \le M^*$, the MER is of rate $(M^{\varsigma})^{1/\alpha-1}$ which solely relies on the sampling frequency. Another interesting finding is that the rate of MER relies on the smoothness degree c, d of the mean functions and covariance functions, as well as the closeness degree α between the two populations. This differs from the estimation problems in which the rate of MER only relies on the smoothness degree of the mean function ([17]).

4.4 Simulation

In this section, we provide extensive numerical evidence to show the empirical performance of FQDA by comparing with other functional classifiers, including quadratic discriminant method (QD) proposed in [33] and the nonparametric Bayes classifier (NB) proposed in [30]. We evaluate all methods via four synthetic datasets. In all simulations, we generate $n = n_1 = n_2 = 50, 100$ training samples for each class, which indicates $\pi_1 = \pi_2 = 0.5$. The functional data are generated from $\mathcal{GP}(\eta_k, \Omega_k)$, where $k = 1, 2$. The mean for class 1 is $\eta_1(t) = \gamma t$, $\gamma = 1$ or 3, and the mean for class 2 is set to be $\eta_2(t) = 0$, $t \in [0, 1]$. We use the following two models to generate covariance functions Ω_1 and Ω_2.

Model 1: Exponential covariance function: Assign $\Omega_1(t, t') = \exp(-|t - t'|)$, and $\Omega_2(t, t') = \exp(-|t - t'|/\rho)$, where $\rho = 0.75, 1.5$.

Model 2: Matérn covariance function: Let $\Omega_1(t, t') = \frac{\sqrt{2}}{\Gamma(1/2)}|t - t'|^{1/2} K_{1/2}(|t - t'|)$, and $\Omega_2(t, t') = \frac{\sqrt{2}}{\Gamma(1/2)} \left(\frac{|t-t'|}{\rho}\right)^{1/2} K_{1/2}\left(\frac{|t-t'|}{\rho}\right)$, where $K_{1/2}(\cdot)$ is the modified Bessel function and $\rho = 0.75, 1.5$.

The random functions are sampled at M equally spaced sampling points from 0 to 1. We choose M from $\{3, 5, 10, 20, 30, 40, 50\}$ to detect how sampling frequency effects the classification error, where we regard $M = 50$ as the full observation. In each scenario, the number of repetition is set to be 100, and the classification errors are evaluated with 500 samples.

Tables 4.1 to 4.4 present the misclassification rates for three methods given the combinations of two different covariance models and two mean functions. Nevertheless the proposed FQDA classifiers have comparable performance with two competitors given different sample sizes. As expected, when the discrepancy between mean and covariance functions are smaller, e.g. $\gamma = 1$, $\rho = 1.5$, the misclassification rates for all three methods are larger. When the number of sampling points M and sample size n are increasing, misclassification rates of FQDA are decreasing, which echoes the theoretical results in Section 4.3. Similar patterns can be found for the other two methods. When the sampling points are extremely sparse, $M = 3$, FQDA has significantly superior performance compared with QD and NB methods, about 10% smaller

than the other two classifiers. Given moderate sparse sampling points, $M = 5, 10, 20$, all methods have fairly comparable performance. When the sampling points are dense, $M > 20$, FQDA outperforms two counterparts, although these two competitors also work reasonable well.

Table 4.1: Misclassification rates (%) with standard errors in brackets for Model 1 with $\eta_1(t) = 3t$.

		$\rho = 1.5$			$\rho = 0.75$		
M	n	FQDA	QD	NB	FQDA	QD	NB
50	50	6.35(0.01)	7.10(0.01)	7.37(0.02)	7.52(0.01)	7.91(0.01)	8.48(0.02)
	100	5.70(0.01)	6.15(0.01)	6.36(0.02)	6.86(0.01)	7.38(0.01)	7.33(0.01)
40	50	6.30(0.01)	7.00(0.01)	7.32(0.02)	7.53(0.01)	7.96(0.01)	8.28(0.02)
	100	6.02(0.01)	6.13(0.01)	6.46(0.01)	7.23(0.01)	7.20(0.01)	7.53(0.01)
30	50	6.64(0.01)	7.00(0.02)	6.98(0.02)	7.31(0.01)	7.94(0.01)	8.49(0.02)
	100	6.10(0.01)	5.95(0.01)	6.13(0.01)	7.18(0.01)	7.02(0.01)	7.30(0.02)
20	50	6.66(0.01)	6.65(0.01)	6.91(0.02)	8.16(0.01)	7.77(0.01)	8.63(0.02)
	100	6.40(0.01)	5.98(0.01)	5.97(0.01)	7.50(0.01)	6.93(0.01)	7.12(0.01)
10	50	8.31(0.01)	6.43(0.01)	6.68(0.02)	9.36(0.01)	7.67(0.02)	7.85(0.02)
	100	7.99(0.01)	5.89(0.01)	5.71(0.01)	9.05(0.01)	7.15(0.01)	6.88(0.01)
5	50	9.14(0.01)	5.96(0.01)	6.33(0.01)	9.77(0.01)	7.36(0.01)	7.42(0.01)
	100	9.13(0.01)	5.75(0.01)	5.92(0.01)	9.68(0.01)	7.00(0.01)	6.71(0.01)
3	50	9.60(0.01)	19.22(0.02)	20.10(0.03)	9.60(0.01)	18.28(0.03)	19.07(0.03)
	100	9.36(0.01)	19.13(0.02)	19.71(0.02)	9.26(0.01)	17.60(0.01)	18.12(0.02)

4.5 Real data analysis

This benchmark data example was extracted from the TIMIT database (TIMIT Acoustic-Phonetic Continuous Speech Corpus, NTIS, US Dept of Commerce) which is a widely used resource for research in speech recognition and functional data classification [40]. The data set we use was formed by selecting five phonemes for classification based on digitized speech from this database. From each speech frame, a log-periodogram was used as transformation for casting speech data in a form suitable for speech recognition. The five phonemes in this data set are transcribed as follows: "sh" as in "she", "dcl" as in "dark", "iy" as the vowel in "she", "aa" as the vowel in "dark", and "ao" as the first vowel in "water". For illustration purpose, we focus on the "aa", "iy" and "ao" phoneme classes. Each speech frame is represented by $n = 400$ samples at a 16-kHz sampling rate; the first $M = 150$ frequencies from each subject are retained. Figure 4.1 displays 10 log-periodograms for each class phoneme.

Table 4.2: Misclassification rates (%) with standard errors in brackets for Model 1 with $\eta_1(t) = t$.

		$\rho = 1.5$			$\rho = 0.75$		
M	n	FQDA	QD	NB	FQDA	QD	NB
50	50	29.93(0.02)	33.26(0.03)	30.09(0.03)	31.99(0.02)	35.79(0.04)	35.25(0.03)
	100	29.37(0.02)	32.10(0.02)	26.37(0.02)	30.49(0.02)	32.49(0.02)	31.54(0.03)
40	50	29.89(0.02)	33.40(0.03)	30.82(0.03)	31.68(0.02)	35.27(0.03)	34.47(0.03)
	100	29.25(0.02)	31.36(0.02)	27.73(0.03)	30.66(0.02)	33.28(0.03)	32.56(0.03)
30	50	30.30(0.02)	33.49(0.03)	31.42(0.03)	32.19(0.02)	35.27(0.04)	35.07(0.03)
	100	29.55(0.02)	31.99(0.03)	28.23(0.02)	31.39(0.02)	33.03(0.02)	32.00(0.02)
20	50	30.57(0.02)	32.90(0.03)	31.95(0.04)	32.56(0.01)	35.02(0.04)	35.13(0.04)
	100	30.17(0.02)	31.24(0.02)	29.32(0.02)	31.61(0.02)	32.77(0.02)	32.93(0.03)
10	50	32.39(0.02)	32.59(0.03)	33.96(0.03)	33.72(0.02)	34.47(0.03)	35.59(0.03)
	100	31.89(0.02)	31.09(0.02)	31.05(0.03)	32.96(0.02)	32.49(0.02)	33.12(0.03)
5	50	32.98(0.02)	31.81(0.03)	33.38(0.04)	33.61(0.02)	33.44(0.03)	34.42(0.03)
	100	32.78(0.02)	30.15(0.02)	31.12(0.02)	33.10(0.02)	31.94(0.02)	32.24(0.02)
3	50	33.13(0.01)	39.10(0.02)	41.48(0.03)	33.54(0.02)	38.43(0.02)	40.80(0.04)
	100	32.96(0.01)	38.68(0.02)	40.09(0.03)	33.43(0.02)	38.64(0.02)	39.41(0.03)

We randomly select training sample size $n_1 = n_2 = 300$ to train the classifiers of three methods and the rest of 100 samples remained as the test samples. Table 4.5 report the mean percentage (averaged over the 100 repetitions) of misclassified test curves. It can be seen that NB classifier slightly outperforms FQDA when classifying "aa" and "iy" and FQDA remains excellent when classifying "ao" and "iy".

4.6 Discussion

We present a new minimax optimality viewpoint for solving functional classification problems in imperfect classification scenario. In contrast to literature on perfect classification, our results are able to deal with the more practical scenarios where the optimal Bayes risk is asymptotically nonvanishing and the two populations are relatively close. Our contributions are twofold. First, we provide sharp convergence rates for MER when data are either fully or discretely observed, as well as a critical sampling frequency that governs the rate in the latter case. Second, we propose classifiers based on FQDA which are proven to achieve minimax optimality.

Table 4.3: Misclassification rates (%) with standard errors in brackets for Model 2 with $\eta_1(t) = 3t$.

		$\rho = 1.5$			$\rho = 0.75$		
M	n	FQDA	QD	NB	FQDA	QD	NB
50	50	6.20(0.01)	6.88(0.01)	7.08(0.02)	7.52(0.01)	8.28(0.02)	8.69(0.02)
	100	5.68(0.01)	6.02(0.01)	6.12(0.01)	6.98(0.01)	7.49(0.01)	7.43(0.01)
40	50	6.23(0.01)	6.80(0.01)	6.78(0.02)	7.54(0.01)	8.03(0.01)	8.76(0.02)
	100	5.83(0.01)	6.22(0.01)	6.21(0.01)	7.07(0.01)	7.07(0.01)	7.23(0.02)
30	50	6.40(0.01)	6.60(0.01)	6.97(0.02)	7.57(0.01)	7.91(0.01)	8.47(0.02)
	100	6.15(0.01)	6.23(0.01)	5.94(0.01)	7.34(0.01)	7.38(0.01)	7.58(0.02)
20	50	6.81(0.01)	6.97(0.02)	7.29(0.02)	8.15(0.01)	7.96(0.01)	8.31(0.02)
	100	6.50(0.01)	5.96(0.01)	5.97(0.01)	7.60(0.01)	7.12(0.01)	7.09(0.01)
10	50	8.71(0.01)	6.50(0.01)	6.79(0.02)	9.27(0.01)	7.38(0.01)	8.02(0.02)
	100	7.93(0.01)	6.02(0.01)	6.03(0.01)	9.15(0.01)	7.06(0.01)	6.94(0.01)
5	50	9.31(0.01)	5.93(0.01)	6.42(0.02)	9.84(0.01)	7.41(0.01)	7.68(0.02)
	100	9.03(0.01)	5.80(0.01)	6.07(0.01)	9.40(0.01)	6.97(0.01)	6.81(0.01)
3	50	9.76(0.01)	19.06(0.02)	20.29(0.04)	9.51(0.01)	18.05(0.02)	18.91(0.04)
	100	9.50(0.01)	19.07(0.02)	19.94(0.02)	9.27(0.01)	18.21(0.02)	18.72(0.02)

Table 4.4: Misclassification rates (%) with standard errors in brackets for Model 2 with $\eta_1(t) = t$.

		$\rho = 1.5$			$\rho = 0.75$		
M	n	FQDA	QD	NB	FQDA	QD	NB
50	50	29.90(0.02)	33.71(0.03)	29.88(0.04)	31.62(0.02)	35.03(0.03)	34.90(0.02)
	100	28.99(0.02)	31.96(0.02)	26.41(0.03)	30.66(0.02)	33.47(0.03)	31.92(0.02)
40	50	30.21(0.02)	33.64(0.04)	30.59(0.03)	31.55(0.02)	35.15(0.04)	34.90(0.03)
	100	29.47(0.02)	31.62(0.02)	27.43(0.02)	30.82(0.02)	33.52(0.03)	32.31(0.03)
30	50	30.22(0.02)	33.80(0.04)	31.56(0.04)	31.78(0.02)	34.05(0.03)	34.82(0.03)
	100	29.59(0.02)	32.14(0.03)	29.12(0.02)	31.35(0.02)	32.97(0.03)	32.06(0.02)
20	50	31.14(0.02)	33.45(0.03)	32.67(0.03)	32.74(0.02)	34.40(0.03)	35.24(0.03)
	100	30.08(0.02)	31.52(0.02)	29.45(0.02)	31.90(0.02)	33.30(0.03)	33.16(0.03)
10	50	32.12(0.02)	32.47(0.03)	32.86(0.04)	34.06(0.02)	34.34(0.03)	35.99(0.04)
	100	31.88(0.02)	31.55(0.02)	30.63(0.02)	32.73(0.02)	31.97(0.02)	32.94(0.02)
5	50	33.22(0.02)	31.77(0.03)	32.87(0.04)	33.90(0.02)	32.98(0.03)	35.10(0.04)
	100	32.57(0.02)	30.50(0.02)	31.52(0.03)	33.35(0.02)	32.27(0.02)	32.49(0.03)
3	50	33.35(0.02)	39.11(0.02)	41.34(0.03)	33.40(0.02)	38.89(0.02)	40.39(0.04)
	100	33.30(0.02)	39.08(0.02)	40.32(0.03)	33.13(0.02)	28.18(0.02)	39.22(0.03)

Table 4.5: Misclassification rates (%) with standard errors in brackets for Speech Recognition data ("aa" vs "iy").

Classes	FQDA	QD	NB
"aa" vs "iy"	0.090(0.003)	0.185(0.003)	0.040(0.002)
"ao" vs "iy"	0.040(0.001)	0.330(0.005)	0.130(0.003)

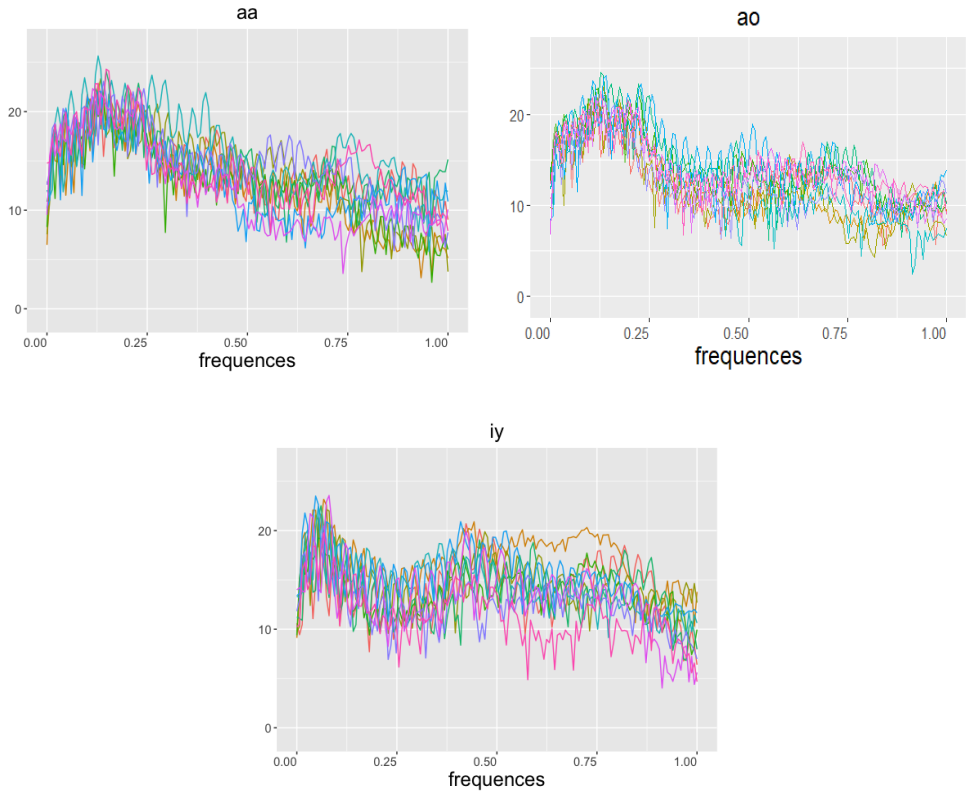

Figure 4.1: A sample of 10 log-periodograms per class

Chapter 5

Functional Classification via Deep Neural Networks

5.1 Introduction

Due to modern advanced technology, complex functional data are ubiquitous. A fundamental problem in functional data analysis is to classify a data function based on training samples. A typical 1D example is the speech recognition data extracted from the TIMIT database, in which the training samples are digitized speech curves of American English speakers from different phoneme groups, and the task is to predict the phoneme of a new speech curve. Typical 2D and 3D examples include the brain imaging data extracted from Early Mild Cognitive Impairment (EMCI) or Alzeheimer's Disease (AD), in which the training samples are digitized brain images, and the task is to predict the stage of a new patient. Besides above examples, functional data classification has wide applications in various fields such as machine learning, genetics, agriculture, chemimetrics and artificial intelligence [98, 60, 90, 23]. Recent monographs [46, 54] provide comprehensive and general discussions on this field.

Classical multivariate analysis techniques such as logistic regression or discriminant analysis no longer work for functional data due to its intrinsically infinite dimensionality [115]. A mainstream technique in functional data classification is based on functional principle component analysis (FPCA) such as functional discriminant analysis [95, 32, 31, 33, 42, 30, 12, 85, 1, 119]. Functional discriminant analysis requires data function being Gaussian process, under which the decision boundary is characterized by a linear or quadratic polynomial so that classic discriminant analysis approach can accurately recover the decision boundary. Gaussian assumption is restrictive and often violated in practice. When data distributions are general

non-Gaussian, the resulting decision boundary is often complicated which cannot be accurately recovered by existing approaches. Our aim is to construct a new functional classifier to overcome this challenge.

In this paper, we propose a new approach, called as functional deep neural network (FDNN), for multi-dimensional functional data classification. We start from FPCA to extract the functional principle components of the data functions, and then train a DNN based on these FPCs as well as their corresponding class labels. As demonstrated through numerical studies, our FDNN approach performs well in classifying complex curve or imaging data. Moreover, our FDNN has desirable theoretical properties. Intuitively, when the network architectures are suitably selected, DNN shall have large expressive power (see [86, 128]) so that functional Bayes classifier can be accurately recovered, even though data distributions are complex. Specifically, we show that, when the log-ratio of the population densities demonstrates a locally connected functional modular structure, our FDNN is proven minimax optimal. The proposed functional modular structure is useful to overcome the infinite dimensionality of functional data, and is meaningful as demonstrated in various examples (see Section 5.5). Relevant modular structures have been recently adopted by researchers in nonparametric regression and classification to characterize the local behavior of the multivariate input variables, based on which DNN approaches are proven to overcome the "curse of dimensionality." See [92, 11, 70, 73, 118, 61, 47, 52, 15].

The rest of this article is organized as follows. In Section 5.2 we review functional Bayes classifier in general setting. In Section 5.3, we propose FDNN classifier. In Section 5.4, we establish theoretical properties of FDNN under suitable technical assumptions. Section 5.5 provides three progressive examples to demonstrate the validity of these technical assumptions. In Section 5.6, performances of FDNN and its competitors are demonstrated through simulation studies. In Section 5.7, we apply FDNN to speech recognition data and Alzeheimer's Disease data. Section 5.8 summarizes the conclusions. Technical proofs are provided in Appendix and a supplement document.

5.2 Functional Bayes classifier under non-Gaussianity

In this section, we review functional Bayes classifier for binary classification. Let $X(s), s \in \mathcal{S} := [0,1]^d$ be a random process with $\int_{\mathcal{S}} \mathbb{E}X(s)^2 ds < \infty$, and $Y \in \{-1, 1\}$ be a uniform random class label such that, under $Y = k$, $X(s)$ has unknown mean function $\mu_k(s)$ and unknown covariance function $\Omega_k(s, s')$, for $s, s' \in \mathcal{S}$. Suppose that Ω_k satisfies a Karhunen–Loéve decomposition:

$$\Omega_k(s, s') = \sum_{j=1}^{\infty} \lambda_{kj} \psi_{kj}(s) \psi_{kj}(s'), s, s' \in \mathcal{S}, \tag{5.1}$$

where $\psi_{kj}, j \geq 1$ is an orthonormal basis of $L^2(\mathcal{S})$ with respect to the usual L^2 inner product, and $\lambda_{k1} \geq \lambda_{k2} \geq \cdots > 0$ are nonincreasing positive eigenvalues. Notably, (5.1) requires the covariance functions being decomposed in terms of the same eigenfunctions, which is a common assumption in functional classification literature; see [31] and [30]. Further relaxation of this assumption is discussed in Section .

Under $Y = k$, write $X(s) = \sum_{j=1}^{\infty} \xi_j \psi_{kj}(s)$, where ξ_j's are pairwise uncorrelated random coefficients. Let $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots)$ and $h_k(\cdot)$ be the unknown conditional density of $\boldsymbol{\xi}$ under $Y = k$. Define $Q^*(\cdot)$ as the log density ratio functional between the two classes:

$$Q^*(\boldsymbol{\xi}) = \log \left(\frac{h_1(\boldsymbol{\xi})}{h_{-1}(\boldsymbol{\xi})} \right).$$

The functional Bayes rule for classifying a data function $X \in L^2(\mathcal{S})$ thus has an expression

$$G^*(X) = \begin{cases} 1, & Q^*(\boldsymbol{\xi}) \geq 0, \\ -1, & Q^*(\boldsymbol{\xi}) < 0. \end{cases} \tag{5.2}$$

Direct estimation of Q^* is infeasible due to the infinite dimensionality of the input. A common practice is to estimate its finite-dimensional truncation. For $J \geq 1$, let $\boldsymbol{\xi}_J = (\xi_1, \ldots, \xi_J)^\top$ be the leading J components of $\boldsymbol{\xi}$ and $h_k^{(J)}(\cdot)$ be the marginal density of $\boldsymbol{\xi}_J$ under $Y = k$, for

$k = \pm 1$. Define the truncated log density ratio

$$Q_J^*(\boldsymbol{\xi}_J) = \log\left(\frac{h_1^{(J)}(\boldsymbol{\xi}_J)}{h_{-1}^{(J)}(\boldsymbol{\xi}_J)}\right),$$

which is the log density ratio of $h_1^{(J)}$ to $h_{-1}^{(J)}$. The intuition is that, when J is large, $h_k^{(J)}$ approaches h_k so that Q_J^* is an accurate approximation of Q^*. Our aim is to design an efficient method to estimate Q_J^*, which will in turn estimate Q^*.

5.3 Functional deep neural network classifier

Suppose we observe n i.i.d. training samples $\{(X_i(s), Y_i) : 1 \leq i \leq n, s \in \mathcal{S}\}$, which are independent of $X(s), s \in \mathcal{S}$ to be classified. For $k = \pm 1$, define sample covariance function

$$\widehat{\Omega}_k(s, s') = \frac{1}{n_k}\sum_{i \in I_k}(X_i(s) - \bar{X}_k(s))(X_i(s') - \bar{X}_k(s')), \quad s, s' \in \mathcal{S},$$

where I_k is the collection of i such that $Y_i = k$, $n_k := |I_k|$ and $\bar{X}_k(s) = \frac{1}{n_k}\sum_{i \in I_k} X_i(s)$ is the sample mean function of class k. Perform Karhunen–Loéve decomposition for $\widehat{\Omega}_k$:

$$\widehat{\Omega}_k(s, s') = \sum_{j=1}^{\infty} \widehat{\lambda}_{kj}\widehat{\psi}_{kj}(s)\widehat{\psi}_{kj}(s'), s, s' \in \mathcal{S},$$

and write the sample data function X_i, under $Y_i = k$, as

$$X_i(s) = \sum_{j=1}^{\infty} \widehat{\xi}_{ij}\widehat{\psi}_{kj}(s), i = 1, \ldots, n.$$

Intuitively, $\widehat{\boldsymbol{\xi}}^{(i)} := (\widehat{\xi}_{i1}, \widehat{\xi}_{i2}, \ldots)$ is an estimator of $\boldsymbol{\xi}^{(i)} := (\xi_{i1}, \xi_{i2}, \ldots)$, in which ξ_{ij} are unobservable random coefficients of X_i with respect to the population basis ψ_{kj}. Hence, it is natural to design classifiers based on $\widehat{\boldsymbol{\xi}}^{(i)}$'s.

Let $\widehat{\boldsymbol{\xi}}_J^{(i)} = (\widehat{\xi}_{i1}, \ldots, \widehat{\xi}_{iJ})^\top$ be the J-dimensional truncation of $\widehat{\boldsymbol{\xi}}^{(i)}$ for $i = 1, \ldots, n$. When X_i's are Gaussian processes, various classifiers have been proposed such as centroid classifier ([31]), QDA ([33]) and nonparametric Bayes classifier ([30]). When X_i's are non-Gaussian,

one major challenge is the underlying complicated form of the conditional densities h_1 and h_{-1} so that estimation of Q_J^* is typically difficult. Inspired by the rich approximation power of DNN, in this section, we propose a new classifier called FDNN (functional+DNN) that can accurately estimate Bayes classifiers even when h_1 and h_{-1} are non-Gaussian complicated.

We will train a DNN to estimate Q_J^* based on $\widehat{\boldsymbol{\xi}}_J^{(i)}$'s. In what follows, we will describe our method in details. Let σ denote the rectifier linear unit (ReLU) activation function, i.e., $\sigma(x) = (x)_+$ for $x \in \mathbb{R}$. For any real vectors $\boldsymbol{V} = (v_1, \ldots, v_w)^\top$ and $\boldsymbol{y} = (y_1, \ldots, y_w)^\top$, define the shift activation function $\sigma_{\boldsymbol{V}}(\boldsymbol{y}) = (\sigma(y_1 - v_1), \ldots, \sigma(y_w - v_w))^\top$. For $L \geq 1$, $\boldsymbol{p} = (p_1, \ldots, p_L) \in \mathbb{N}^L$, let $\mathcal{F}(L, J, \boldsymbol{p})$ denote the class of fully connected feedforward DNN with J inputs, L hidden layers and, for $l = 1, \ldots, L$, p_l nodes on the lth hidden layer. Equivalently, any $f \in \mathcal{F}(L, J, \boldsymbol{p})$ has an expression

$$f(\boldsymbol{x}) = \mathbf{W}_L \sigma_{\boldsymbol{V}_L} \mathbf{W}_{L-1} \sigma_{\boldsymbol{V}_{L-1}} \ldots \mathbf{W}_1 \sigma_{\boldsymbol{V}_1} \mathbf{W}_0 \boldsymbol{x}, \quad \boldsymbol{x} \in \mathbb{R}^J, \tag{5.3}$$

where $\mathbf{W}_l \in \mathbb{R}^{p_{l+1} \times p_l}$, for $l = 0, \ldots, L$, are weight matrices, $\boldsymbol{V}_l \in \mathbb{R}^{p_l}$, for $l = 1, \ldots, L$, are shift vectors. Here we adopt the convention that $p_0 = J$ and $p_{L+1} = 1$.

Due to the large capacity of $\mathcal{F}(L, J, \boldsymbol{p})$, training a DNN typically overfits the data, therefore, proper regularization is necessary. A common practice is to sparsify the network parameters by methods such as dropout; see [49]. Our approach follows [92, 15] to train a sparse DNN. Specifically, consider the following class of sparse DNN:

$$\mathcal{F}(L, J, \boldsymbol{p}, B) = \left\{ f \in \mathcal{F}(L, J, \boldsymbol{p}) : \max_{0 \leq l \leq L} \|\mathbf{W}_l\|_\infty \leq B, \max_{1 \leq l \leq L} \|\mathbf{v}_l\|_\infty \leq B \right\},$$

where $\|\cdot\|_\infty$ denotes the maximum-entry norm of a matrix/vector or supnorm of a function, and $B > 0$ controls the largest weights and shifts.

Given the training data $(\boldsymbol{\xi}_J^{(1)}, Y_1), \ldots, (\boldsymbol{\xi}_J^{(n)}, Y_n)$, let

$$\widehat{f}_\phi(\cdot) = \operatorname*{arg\,min}_{f \in \mathcal{F}(L, J, \boldsymbol{p}, B)} \frac{1}{n} \sum_{i=1}^n \phi(f(\widehat{\boldsymbol{\xi}}_J^{(i)}) Y_i), \tag{5.4}$$

where $\phi(x) = \max(1-x, 0)$ denotes the hinge loss. We then propose the following FDNN classifier: for $X \in L^2(\mathcal{S})$,

$$\widehat{G}^{FDNN}(X) = \begin{cases} 1, & \widehat{f}_\phi(\boldsymbol{\xi}_J) \geq 0, \\ -1, & \widehat{f}_\phi(\boldsymbol{\xi}_J) < 0. \end{cases} \tag{5.5}$$

In practice, we suggest the following data-splitting method for selecting $(L, J, \boldsymbol{p}, B)$:

- Step 1. Randomly divide the whole sample $(\widehat{\boldsymbol{\xi}}_J^{(i)}, Y_i)$'s into two subsets indexed by $\mathcal{I}_1$ and $\mathcal{I}_2$, respectively, with about $|\mathcal{I}_1| = 0.8n$ and $|\mathcal{I}_2| = 0.2n$.

- Step 2. For each $(L, J, \boldsymbol{p}, B)$, we train a DNN $\widehat{f}_{L,J,\boldsymbol{p},B}$ using (5.4) based on subset $\mathcal{I}_1$, and then calculate the testing error based on subset $\mathcal{I}_2$ as

$$\text{err}(L, J, \boldsymbol{p}, B) = \frac{1}{|\mathcal{I}_2|} \sum_{i \in \mathcal{I}_2} I(\widehat{f}_{L,J,\boldsymbol{p},B}(\widehat{\boldsymbol{\xi}}_J^{(i)}) Y_i < 0). \tag{5.6}$$

- Step 3. Choose $(L, J, \boldsymbol{p}, B)$, possibly from a preselected set, to minimize $\text{err}(L, J, \boldsymbol{p}, B)$.

5.4 Minimax optimality of FDNN

For a generic functional classifier $\widehat{G}$, its excess misclassification risk is defined as $\mathcal{E}_h(\widehat{G}) := \mathbb{E}[R_h(\widehat{G}) - R_h(G^*)]$, where $R_h(\widehat{G}) := \mathbb{E}_h[\mathbb{I}\{\widehat{G}(X) \neq Y\}]$ is the misclassification risk of $\widehat{G}$ taken with respect to (X, Y) under $h := \{h_1, h_{-1}\}$, with Y the true class label of X. A central task is to design $\widehat{G}$ that achieves minimax excess misclassification risk (MEMR), i.e.,

$$\max_{h \in \mathcal{H}} \mathcal{E}_h(\widehat{G}) \asymp \inf_{\widehat{G}} \max_{h \in \mathcal{H}} \mathcal{E}_h(\widehat{G}), \tag{5.7}$$

where $\mathcal{H}$ is a proper class of h to be described later and the infimum is taken over all classifiers based on training samples. Classifiers satisfying (5.7) are called as minimax optimal.

There is a rich literature on construction of minimax optimal classifiers when data dimension is fixed or diverging. For instance, classic nonparametric approaches, such as ones directly estimating Bayes classifier nonparametrically, are proven minimax optimal in fixed-dimension

regime; see [75, 107, 108, 58, 42, 39, 78, 47]. When data are high-dimensional Gaussian, discriminant analysis approaches are proven minimax optimal; see [19, 18].On the other hand, under functional Gaussian data, researchers have proposed various functional classifiers, including functional quadratic discriminant analysis (FQDA); see [95, 32, 31, 33, 42, 30, 12, 85, 19, 18]. Gaussianity leads to a quadratic polynomial Q^* which can be effectively estimated by FQDA, based on which [119] showed that FQDA is minimax optimal. It is still unclear how to design optimal functional classifiers when data are non-Gaussian, a gap that the present article attempts to close.

In this section, we will establish minimax optimality of FDNN classifier under non-Gaussian functional data. For technical convenience, assume that the two populations have common known basis, i.e., $\psi_{+1j}(\cdot) = \psi_{-1j}(\cdot)$. Therefore, we can train FDNN classifier based on $\boldsymbol{\xi}_J^{(i)} := (\xi_{i1}, \ldots, \xi_{iJ})^\top$, for $i = 1, \ldots, n$. We will first derive an upper bound for the excess misclassification risk of our FDNN classifier, and then derive a lower bound for the MEMR which matches the above upper bound. Therefore, our FDNN is able to achieve sharp rate of MEMR. Extensions to general basis are possible but need more tedious technical arguments.

Before proceeding further, we introduce some technical assumptions. At high levels, our assumptions are different from those proposed under Gaussian case. For instance, in either high- or infinite-dimensional Gaussian data, it is well known that density ratio between two Gaussian population densities has an explicit expression in terms of mean difference and variance ratio, which impacts the sharp rate of MEMR. More precisely, in high-dimensional Gaussian data classification, the rate depends on the number of nonzero components of mean difference vector [19, 18]; in Gaussian functional data classification, the rate depends on the decay orders of both mean difference series and variance ratio series [119]. Nonetheless, in general non-Gaussian case, likelihood ratio doesn't have an explicit expression, therefore, one cannot simply use mean or variance discrepancy to characterize the sharp rate of MEMR.

In traditional non-Gaussian multivariate data classification, a common strategy is to assume smooth density ratio and controllable noise, under which minimax optimal classifiers were proposed; see [75], [107], [7], [52] and references therein. In functional data, the input variable of Q^* is infinite-dimensional, hence, the above strategy no longer works. We instead

propose a set of functional conditions on Q^* under which minimax optimality shall be established. Such conditions are viewed as infinite-dimensional extensions of [7, 92].

For $t \geq 1$, a measurable subset $D \subset \mathbb{R}^t$ and constants $\beta, K > 0$, define

$$\begin{aligned} &\mathcal{C}^\beta(D, K) \\ = &\left\{ f : D \mapsto \mathbb{R} \Big| \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|<\beta} \|\partial^{\boldsymbol{\alpha}} f\|_\infty + \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|=\lfloor\beta\rfloor} \sup_{\boldsymbol{x},\boldsymbol{x}'\in D,\boldsymbol{x}\neq\boldsymbol{x}'} \frac{|\partial^{\boldsymbol{\alpha}} f(\boldsymbol{x}) - \partial^{\boldsymbol{\alpha}} f(\boldsymbol{x}')|}{\|\boldsymbol{x} - \boldsymbol{x}'\|_\infty^{\beta-\lfloor\beta\rfloor}} \leq K \right\}, \end{aligned}$$

where $\partial^{\boldsymbol{\alpha}} = \partial^{\alpha_1} \dots \partial^{\alpha_t}$ denotes the partial differential operator with multi-index $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_t) \in \mathbb{N}^t$, $|\boldsymbol{\alpha}| = \alpha_1 + \cdots + \alpha_t$. Equivalently, $\mathcal{C}^\beta(D, K)$ is the ball of β-Hölder smooth functions on D with radius K. A function $f : \mathbb{R}^t \to \mathbb{R}$ is said to be locally β-Hölder smooth if for any $a, b \in \mathbb{R}$, there exists a constant K (possibly depending on a, b) such that $f \in \mathcal{C}^\beta([a, b]^t, K)$.

For $q \geq 0, J \geq 1$, let $d_0 = J$ and $d_{q+1} = 1$. For $\boldsymbol{d} = (d_1, \dots, d_q) \in \mathbb{N}_+^q$, $\boldsymbol{t} = (t_0, \dots, t_q) \in \mathbb{N}_+^{q+1}$ with $t_u \leq d_u$ for $u = 0, \dots, q$, $\boldsymbol{\beta} := (\beta_0, \dots, \beta_q) \in \mathbb{R}_+^{q+1}$, let $\mathcal{G}(q, J, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta})$ be the class of functions g satisfying a modular expression

$$g(\boldsymbol{x}) = g_q \circ \cdots \circ g_0(\boldsymbol{x}), \ \forall \boldsymbol{x} \in \mathbb{R}^{d_0}, \tag{5.8}$$

where $g_u = (g_{u1}, \dots, g_{ud_{u+1}}) : \mathbb{R}^{d_u} \mapsto \mathbb{R}^{d_{u+1}}$ and $g_{uv} : \mathbb{R}^{t_u} \mapsto \mathbb{R}$ are locally β_u-Hölder smooth. The d_u arguments of g_u are locally connected in the sense that each component g_{uv} only relies on $t_u (\leq d_u)$ arguments. Similar structures have been considered by [92] and [70] in multivariate regression to overcome high-dimensionality. Generalized additive model [44] and tensor product space ANOVA model [67] are special cases; see [70].

Let $\mathcal{H}^* \equiv \mathcal{H}^*(q, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta})$ be the class of population densities $h = \{h_1, h_{-1}\}$ of $\boldsymbol{\xi}$ such that, for any $J \geq 1$, $Q_J^* \in \mathcal{G}(q, J, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta})$. Equivalently, for any $h \in \mathcal{H}^*$ and $J \geq 1$, the corresponding truncated log density ratio Q_J^* has a modular structure (5.8) with certain smoothness. Although Q_J^* has J arguments, it involves at most $t_0 d_1$ effective arguments, implying that the two population densities differ by a small number of variables. Relevant conditions are necessary for high-dimensional classification. For instance, in high-dimensional Gaussian data classification, [19, 18] show that, to consistently estimate Bayes classifier, it is necessary that

the mean vectors differ at a small number of components. The modular structure holds for arbitrary J, which may be viewed as a functional extension of [92]. Note that the density class $\mathcal{H}^*$ covers many popular models studied in literature, either Gaussian or non-Gaussian; see Section 5.5. Moreover, we introduce the following regularity conditions on Q^*.

Assumption 1. *(Functional Tsybakov noise condition) There exist constants $C > 0$ and $\alpha \geq 0$ such that*

$$\mathbb{P}\left(\left|\frac{1-\exp\left\{-Q^*\left(\boldsymbol{\xi}\right)\right\}}{1+\exp\left\{-Q^*\left(\boldsymbol{\xi}\right)\right\}}\right| \leq x\right) \leq Cx^{\alpha}, \qquad \forall x > 0. \tag{5.9}$$

Assumption 2. *(Approximation error of Q_J^*) There exist a constant $J_0 \geq 1$ and decreasing functions $\epsilon(\cdot) : [1, \infty) \to \mathbb{R}_+$ and $\Gamma(\cdot) : [0, \infty) \to \mathbb{R}_+$, with $\sup_{J \geq 1} J^{\varrho}\epsilon(J) < \infty$ for some $\varrho > 0$ and $\int_0^{\infty} \Gamma(x)dx < \infty$, such that for any $J \geq J_0$ and $x > 0$,*

$$\mathbb{P}\left(|Q^*(\boldsymbol{\xi}) - Q_J^*(\boldsymbol{\xi}_J)| \geq x\right) \leq \epsilon(J)\Gamma(x). \tag{5.10}$$

Assumption 1 characterizes the discrepancy between Q^* and random guess. Specifically, it requires that the probability of Q^* close to 0 by x is upper bounded by an order x^{α}. Assumption 1 is a functional extension of the classic *Tsybakov noise condition*, which is necessary in establishing minimax classification in multivariate case (see [75] and [107]). Assumption 2 provides an upper bound on the probability of Q^* differing from Q_J^* by at least x, which approaches zero if either J or x tends to infinity, implying that Q_J^* is an accurate approximation of Q^*. Both assumptions can be verified in concrete examples; see Section 5.5.

Our MEMR results will be based on the following class of population densities of $\boldsymbol{\xi}$:

$$\mathcal{H} \equiv \mathcal{H}\left(q, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, \alpha, C, \epsilon(\cdot), \Gamma(\cdot)\right) = \left\{h \in \mathcal{H}^* : Q^* \text{ satisfies both Assumptions 1 and 2}\right\}.$$

Finally, we introduce an assumption on the orders of $L, J, \boldsymbol{p}, B$, under which the exact rate of MEMR shall be established. Let

$$S_0 = \min_{0 \leq u \leq q} \frac{\beta_u^*(\alpha+1)}{\beta_u^*(\alpha+2)+t_u}, S_1 = \max_{0 \leq u \leq q} \frac{t_u}{\beta_u^*(\alpha+2)+t_u}, S_2 = \min_{0 \leq u \leq q} \frac{1}{\beta_u^*(\alpha+2)+t_u},$$

where $\beta_u^* := \beta_u \prod_{k=u+1}^{q} (\beta_k \wedge 1)$.

Assumption 3. *The DNN class $\mathcal{F}(L, J, \boldsymbol{p}, B)$ satisfies*

(a) $L \asymp \log n$;

(b) $\left(n \log^{-3} n\right)^{S_0/\rho} \lesssim J \lesssim (n \log^{-3} n)^{S_1}$;

(c) $\max_{1 \le \ell \le L} p_\ell \asymp (n \log^{-3} n)^{S_1}$;

(d) $B \asymp (n \log^{-3} n)^{S_2}$.

Assumption 3 (a), (c) and (d) provide exact orders on $L, \boldsymbol{p}, B$, respectively. Assumption 3 (b) provides a range on J. Notably, this condition implies $\varrho \ge S_0/S_1$, i.e., the function $\epsilon(J)$ rapidly converges to zero when $J \to \infty$.

Theorem 5.1. *There exist positive constants C_1, C_2, depending on $q, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, \alpha, C, \epsilon(\cdot), \Gamma(\cdot)$, such that the following results hold:*

(i) $\inf_{\widehat{G}} \sup_{h \in \mathcal{H}} \mathcal{E}_h(\widehat{G}) \ge C_1 n^{-S_0}$, *where the infimum is taken over all classifiers $\widehat{G}$ based on training samples;*

(ii) under Assumption 3, it holds that

$$\sup_{h \in \mathcal{H}} \mathcal{E}_h(\widehat{G}^{FDNN}) \le C_2 \left(\frac{\log^3 n}{n} \right)^{S_0}.$$

Theorem 5.1 establishes a nonasymptotic rate for the MEMR which is of order n^{-S_0}. Moreover, the proposed FDNN classifier is able to achieve this rate up to a logarithmic factor, and hence, is minimax optimal. Since S_0 involves the intrinsic dimensions t_u's rather than the original dimensions d_u's, the rate of MEMR is typically fast, demonstrating the theoretical advantage of our FDNN classifier.

5.5 Examples

The minimax results in Section 5.4 are based on parameter space $\mathcal{H}$. In this section, we provide some concrete examples to demonstrate the validity of such space.

5.5.1 Gaussian functional data with independent coefficients

Suppose that, under $Y = k$, the random coefficients ξ_j are independent Gaussian with mean μ_{kj} and variance λ_{kj}. Define $M = \{j : \mu_{1j} \neq \mu_{-1j}\}$ and $N = \{j : \lambda_{1j} \neq \lambda_{-1j}\}$. Assume that M, N are mutually disjoint with common cardinality ω. It can be shown that, for any $J \geq J_0 := \max M \cup N$, $Q_J^*(\boldsymbol{\xi}_J) = g_1(g_0(\boldsymbol{\xi}_J))$, where g_0 has components $g_{0j}(\xi_j) = a_j\xi_j^2 + b_j\xi_j + c_j$ for some constants a_j, b_j, c_j depending on μ_{1j}, μ_{-1j}, λ_{1j}, λ_{-1j}, and $g_1(g_0(\boldsymbol{\xi}_J)) = \sum_{j\in M\cup N} g_{0j}(\xi_j)$. Clearly, $d_0 = J$ and $t_0 = 1$, and $M \cup N$ has cardinality 2ω, $d_1 = t_1 = 2\omega$. So $Q_J^* \in \mathcal{G}(1, J, 2\omega, (1, 2\omega), \beta)$ for any $\beta > 0$. Meanwhile, Assumption 1 holds for $\alpha = 1$, and Assumption 2 holds for J_0, since $Q_J^* = 0$ for all $J \geq J_0$, and for any function $\epsilon(\cdot)$ with exponential tails and any density $\Gamma(\cdot)$.

5.5.2 Student's t functional data with independent coefficients

Suppose that, under $Y = k$, ξ_j are independent Student's t variables $t_{\nu_{kj}}$, where $\nu_{kj} \geq 1$ are degrees of freedom of the t variables. Define $M = \{j : \mu_{1j} \neq \mu_{-1j}\}$ whose cardinality is ω. It can be shown that, for any $J \geq J_0 := J_0 := \max M \cup N$, $Q_J^*(\boldsymbol{\xi}_J) = g_1(g_0(\boldsymbol{\xi}_J))$, where g_0 has components

$$g_{0j}(\xi_j) = \log e_j - \frac{\nu_{1j}+1}{2}\log\left(1 + \frac{\xi_j^2}{\nu_{1j}}\right) + \frac{\nu_{-1j}+1}{2}\log\left(1 + \frac{\xi_j^2}{\nu_{-1j}}\right),$$

for some constant e_j depending on ν_{kj}, and $g_1(g_0(\boldsymbol{\xi}_J)) = \sum_{j\in M\cup N} g_{0j}(\xi_j)$. Similar to Section 5.5.1, we have $Q_J^* \in \mathcal{G}(1, J, 2\omega, (1, 2\omega), \beta)$ for any $\beta > 0$. Assumptions 1 and 2 can be similarly verified as well.

5.5.3 Student's t functional data with dependent coefficients

We consider an extension of Section 5.5.2 which involves dependent coefficients. Let $p \geq 1$ and $\nu \geq 2$ be integers. Suppose that, under $Y = k$, $\boldsymbol{\zeta}_j := (\xi_j, \xi_{j+1}, \ldots, \xi_{j+p-1})^\top$, $j = 1, p+1, 2p+1, \ldots$ are independent multivariate Student's t vectors following $t_\nu(\boldsymbol{\mu}_{kj}, \boldsymbol{\Sigma}_{kj})$, where $\boldsymbol{\mu}_{kj} \in \mathbb{R}^p$ and positive definite $\boldsymbol{\Sigma}_{kj}$ is $p \times p$ positive definite. Define $M = \{j : \boldsymbol{\mu}_{1j} \neq \boldsymbol{\mu}_{-1j}\}$ and $N = \{j : \boldsymbol{\Sigma}_{1j} \neq \boldsymbol{\Sigma}_{-1j}\}$. Assume M, N are mutually disjoint with common cardinality ω.

For any $J \geq J_0 := J_0 := \max M \cup N + p - 1$, then it can be shown that

$$
\begin{aligned}
& Q_J^*(\boldsymbol{\xi}_J) \\
= & \sum_{j \in M \cup N} \left\{ \frac{1}{2} \log \left(\frac{|\boldsymbol{\Sigma}_{-1j}|}{|\boldsymbol{\Sigma}_{1j}|} \right)^{1/2} + \frac{\nu + p}{2} \log \left(\frac{1 + \nu^{-1} (\boldsymbol{\zeta}_j - \boldsymbol{\mu}_{-1j})^\top \boldsymbol{\Sigma}_{-1j}^{-1} (\boldsymbol{\zeta}_j - \boldsymbol{\mu}_{-1j})}{1 + \nu^{-1} (\boldsymbol{\zeta}_j - \boldsymbol{\mu}_{1j})^\top \boldsymbol{\Sigma}_{1j}^{-1} (\boldsymbol{\zeta}_j - \boldsymbol{\mu}_{1j})} \right) \right\}.
\end{aligned}
$$

Note that there are 2ω terms in the above sum. Similar to Sectoions 5.5.1, we have $Q_J^* \in \mathcal{G}(1, J, 2\omega, (1, 2\omega), \beta)$ for any $\beta > 0$. Assumptions 1 and 2 can be similarly verified as well.

5.6 Simulation

In this section, we examine the performances of FDNN and two competitors, quadratic discriminant method (QD) proposed in [33] and the nonparametric Bayes classifier (NB) proposed in [30], through simulation studies. Our studies involve both $d = 1$ and $d = 2$, corresponding to 1D and 2D functional data, respectively. All experiments are conducted in `R`. We summarize R codes and examples for the proposed FDNN algorithms on `GitHub` (`https://github.com/FDASTATAUBURN/fdnn-classification`).

For 1D functional data, we considered two data generation processes (DGP).

- *DGP1:* Generate $X(t) = \sum_{j=1}^{3} \xi_j \psi_j(t)$, $t \in [0, 1]$, where $\psi_1(t) = \log(t + 2)$, $\psi_2(t) = t$ and $\psi_3(t) = t^3$. Under class k, generate independently $\boldsymbol{\xi}_j \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, $j = 1, 2, 3$, where $\boldsymbol{\mu}_1 = (-1, 2, -3)^\top$, $\boldsymbol{\Sigma}_1 = \text{diag}\left(\frac{3}{5}, \frac{2}{5}, \frac{1}{5}\right)$, $\boldsymbol{\mu}_{-1} = \left(-\frac{1}{2}, \frac{5}{2}, -\frac{5}{2}\right)^\top$, $\boldsymbol{\Sigma}_{-1} = \text{diag}\left(\frac{9}{10}, \frac{1}{2}, \frac{3}{10}\right)$.

- *DGP2:* Generate $X(t) = \sum_{j=1}^{3} \xi_j \psi_j(t)$, $t \in [0, 1]$, where $\psi_j(t)$'s are the same as in DGP1. Under class 1, generate independently $\boldsymbol{\xi}_{ij} \sim N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, where $\boldsymbol{\mu}_1 = (-1, 2, -3)^\top$, $\boldsymbol{\Sigma}_1 = \text{diag}\,(3, 2, 1)$; under class -1, generate independently $\xi_{ij} \sim t_{7-2j}$.

For 2D functional data, we considered two DGPs:

- *DGP3:* Generate $X(s_1, s_2) = \sum_{j=1}^{4} \xi_{ij} \psi_j(s_1, s_2)$, $0 \leq s_1, s_2 \leq 1$, where $\psi_1(s_1, s_2) = s_1 s_2$, $\psi_2(s_1, s_2) = s_1 s_2^2$, $\psi_3(s_1, s_2) = s_1^2 s_2$, $\psi_4(s_1, s_2) = s_1^2 s_2^2$. Under class k, generate independently $\boldsymbol{\xi}_j \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, $j = 1, 2, 3, 4$, where $\boldsymbol{\mu}_1 = (8, -6, 4, -2)^\top$, $\boldsymbol{\Sigma}_1 = \text{diag}\,(8, 6, 4, 2)$, $\boldsymbol{\mu}_{-1} = \left(-\frac{7}{2}, -\frac{5}{2}, \frac{3}{2}, -\frac{1}{2}\right)^\top$, $\boldsymbol{\Sigma}_{-1} = \text{diag}\left(\frac{9}{2}, \frac{7}{2}, \frac{5}{2}, \frac{3}{2}\right)$.

n	DGP1			DGP2		
	FDNN	QD	NB	FDNN	QD	NB
40	31.76(0.10)	38.58(0.02)	38.33(0.02)	16.69(0.04)	39.99(0.01)	39.26(0.03)
100	18.82(0.10)	37.91(0.02)	41.03(0.02)	13.20(0.01)	38.42(0.09)	40.27(0.03)
200	13.19(0.10)	37.35(0.02)	39.92(0.02)	12.29(0.01)	42.63(0.02)	39.84(0.04)
400	9.62(0.04)	36.75(0.02)	38.54(0.02)	12.40(0.01)	43.98(0.09)	38.51(0.04)

Table 5.1: Misclassification rates (%) with standard errors in brackets for DGP1 and DGP2

	n			
	40	100	200	400
DGP 3	0.170(0.066)	0.148(0.055)	0.139(0.054)	0.127(0.040)
DGP 4	0.139(0.055)	0.127(0.014)	0.127(0.040)	0.123(0.011)

Table 5.2: Misclassification rates (%) with standard errors in brackets for DGP3 and DGP4

- *DGP4:* Generate $X(s_1, s_2) = \sum_{j=1}^{4} \xi_{ij}\psi_j(s_1, s_2)$, $0 \leq s_1, s_2 \leq 1$, where $\psi_j(s_1, s_2)$'s are the same as in DGP3. Under class 1, generate independently $\xi_j \sim t_{2j}(\boldsymbol{\mu})$; under class -1, generate independently $\xi_j \sim t_{2j+1}(\boldsymbol{\mu})$, with non-central parameter $\boldsymbol{\mu} = (2, \frac{3}{2}, 1, \frac{1}{2})$.

In each DGP, we generated n training data functions and 500 testing data functions, with $n = 40, 100, 200, 400$. Each data function was sampled over 50 grid points in the respective domain. Misclassification errors were evaluated based on 100 replicated datasets. Network parameters were selected based on training data using Steps 1-3 in Section 5.3. Tables 5.1 summarizes the misclassification rates when the functional data are either Gaussian or non-Gaussian . The discrepancy is increasing with the sample size. Especially, when sample size is large (over 100), the proposed FDNN still outperforms two counterparts. Table 5.2 summarizes the misclassification rates for 2D-functional data classification. Only our proposed FDNN method is applied, as there are no existing methods to compare with. From the table, we observe that both misiclassification rates and the standard errors decrease as the sample size increases, which supports the theoretical findings in Section 5.4.

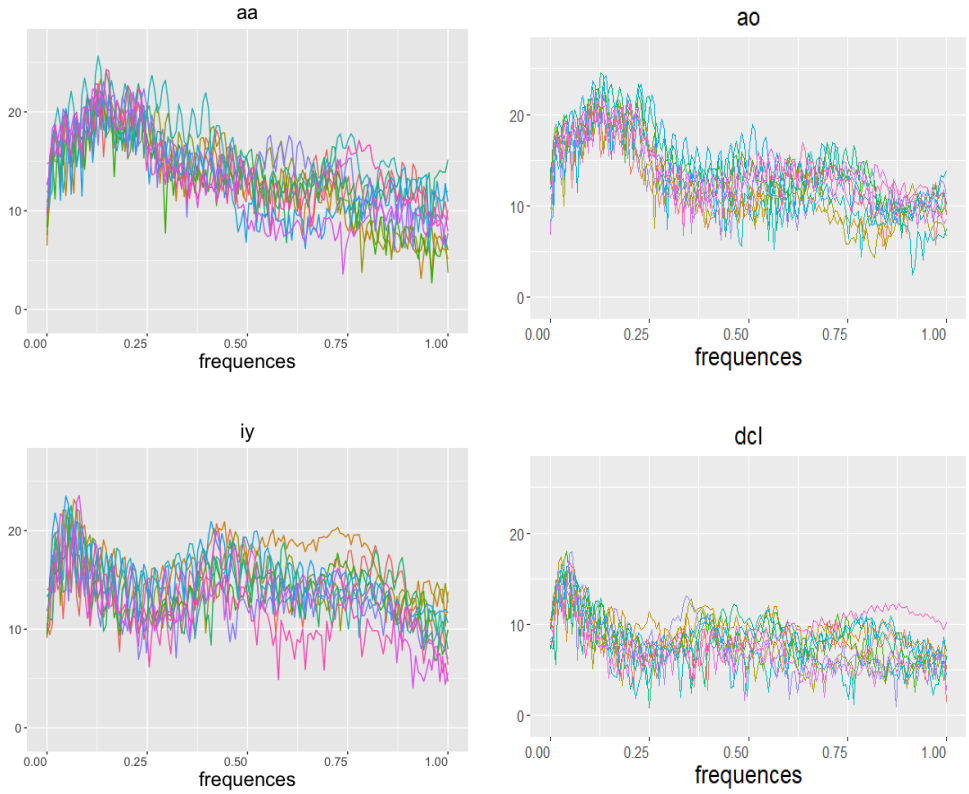

Figure 5.1: A sample of 10 log-periodograms per class

5.7 Real data analysis

5.7.1 TIMIT database

This benchmark data example was extracted from the TIMIT database (`https://catalog.ldc.upenn.edu/LDC93s1`), which is a widely used resource for research in speech recognition and functional data classification. The data set we used was constructed by selecting four phonemes for classification based on digitized speech from this database. From each speech frame, a log-periodogram transformation is applied so as to cast the speech data in a form suitable for speech recognition. The five phonemes in this data set are transcribed as follows: "sh" as in "she", "dcl" as in "dark", "iy" as the vowel in "she", "aa" as the vowel in "dark", and "ao" as the first vowel in "water". For illustration purpose, we focus on the "aa", "ao", "iy" and "dcl" phoneme classes. Each speech frame is represented by $n = 400$ samples at a 16-kHz sampling rate; the first $M = 150$ frequencies from each subject are retained. Figure 5.1 displays 10 log-periodograms for each class phoneme.

"aa" vs "ao"	'aa" vs "iy"	"ao" vs "iy"	"ao" vs "dcl"
20.278(0.014)	0.196(0.001)	0.153(0.004)	0.270(0.003)

Table 5.3: Misclassification rates (%) with standard errors in brackets for Speech Recognition data.

We randomly select training sample size $n_1 = n_2 = 300$ to train the classifiers of three methods and the rest of 100 samples remained as the test samples. Network parameters were selected based on training data using Steps 1-3 in Section 5.3. Table 5.3 reports the mean percentage (averaged over the 100 repetitions) of misclassified test curves.

The dataset used in the preparation of this article were obtained from the ADNI database (`adni.loni.usc.edu`). The ADNI is a longitudinal multicenter study designed to develop clinical, imaging, genetic, and biochemical biomarkers for the early detection and tracking of AD. From this database, we collect PET data from 79 patients in AD group, and 45 patients in EMCI group. This PET dataset has been spatially normalized and post-processed. These AD patients have three to six times doctor visits and we select the PET scans obtained in the third visits. People in EMCI group only have the second visit, and we select the PET scans obtained in the second visits. For AD group, patients' age ranges from 59 to 88 and average age is 76.49, and there are 33 females and 46 males among these 79 subjects. For EMCI group, patients' age ranges from 57 to 89 and average age is 72.33, and there are 26 females and 19 males among these 45 subjects. All scans were reoriented into $79 \times 95 \times 68$ voxels, which means each patient has 68 sliced 2D images with 79×95 pixels. For 2D case, it means each subject has $N = 79 \times 95 = 7,505$ observed pixels for each selected image slice. For 3D case, the observed number of voxels for each patient's brain sample is $N = 79 \times 95 \times 68$.

It is well known that Alzheimer's disease destroys neurons and their connections in hippocampus, the entorhinal cortex, and the cerebral cortex. These parts are corresponding to the first 25 slices. Therefore, for our 2D case study, we specifically select the 5-th, 10-th, 15-th, 20-th and 25-th slices from 68 slices for each patient. We aim to conduct classification based on the information of those slices respectively; see [81]. Figure 5.2 shows the averaged 2D images for two groups at each slice. For 3D case, we focus on the total 25 slices, so the 3D data is observed on $79 \times 95 \times 25$ points. Figure 5.3 demonstrates the misclassification rates for both

2D and 3D brain imaging data. There are several interesting finds. First, given a single slice 2D imaging data, the misclassification rates tend to be larger than using total 25 slices data (3D data). It indicates that 3D data contains more helpful information to decrease the misclassification risk. Second, the 20-th slice provides the lowest one among all 2D data. It is a promising finding for neurologists, as this smallest risk indicates this particular slice presents useful information to distinguish the EMCI and AD groups. Further medical checkups are meaningful for this special location in the brain.

5.8 Discussion

We present a novel FDNN classifier for solving functional classification problems. In comparison with the existing literature, our results are able to deal with non-Gaussian multi-dimensional functional data. Our contributions are threefold. First, we provide sharp convergence rates for EMR via DNNs when data are of functional type, and the result can be applied to a large scope of functional data with complex density functions. Second, not only can our novel FDNN classifier handle one dimensional functional data, but also can classify multi-dimensional functional data. Third, we demonstrate via extensive simulations and real-data examples that the proposed FDNN classifier has outstanding performances under either Gaussian or non-Gaussian assumption.

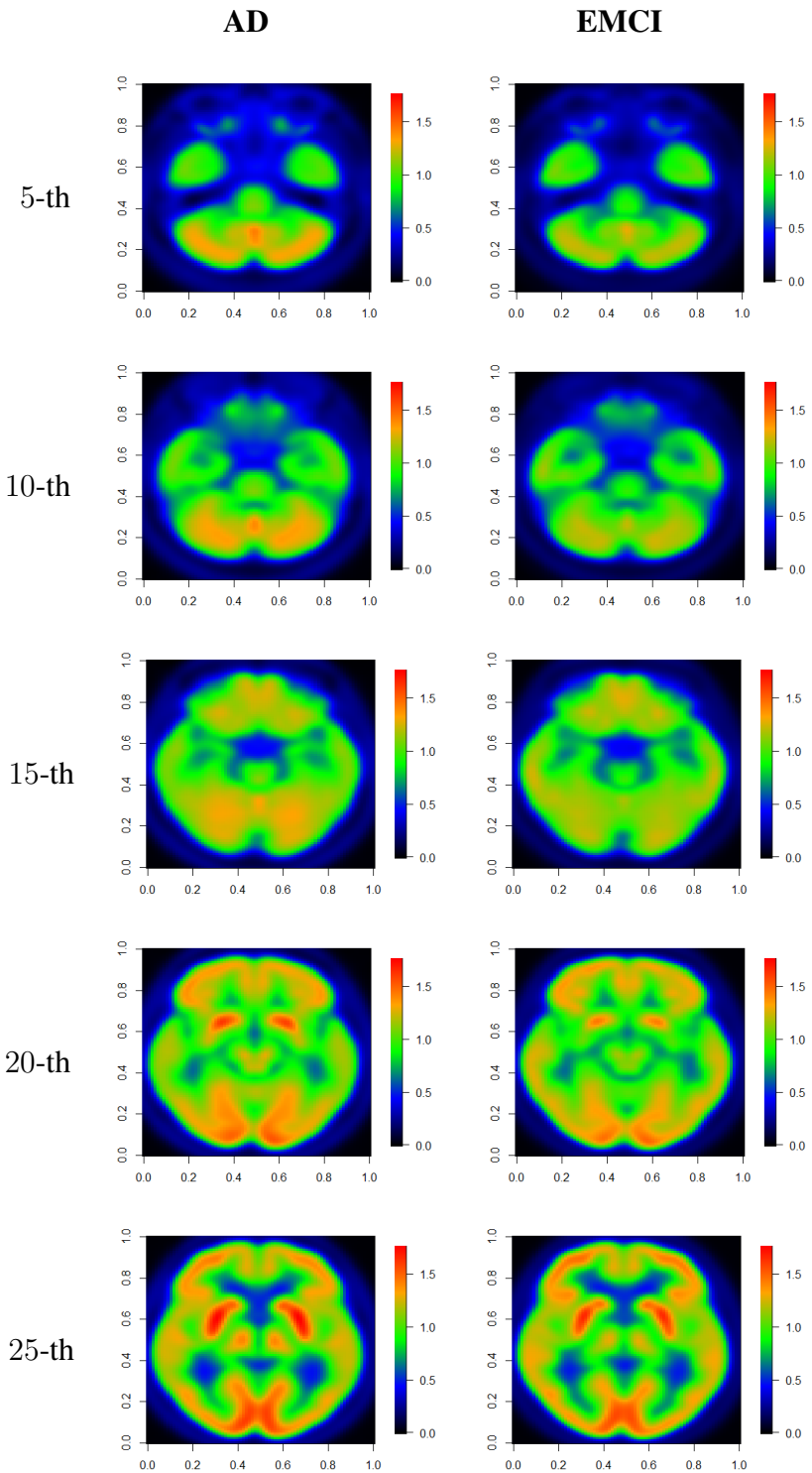

Figure 5.2: Averaged images of the 5-th, the 10-th, the 15-th, the 20-th and the 25-th slices of EMCI (left column) group and AD group (right column).

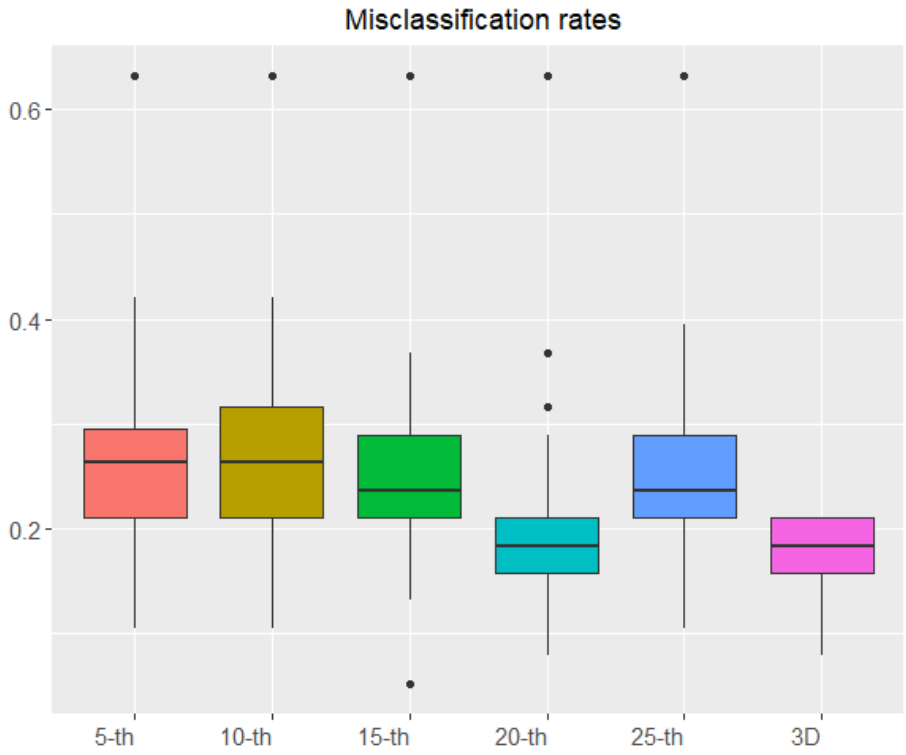

Figure 5.3: Grouped boxplot of misclassification rates for the 5-th, the 10-th, the 15-th, the 20-th , the 25-th slices and 3D data of the first 25 slices between EMCI and AD groups.

Chapter 6

Empirical Likelihood Ratio Tests for Varying Coefficient Geo Models

6.1 Introduction

Varying-coefficient models (VCMs)'s introduced by [45], are regression models commonly applied to examine the interactive associations between a response and predictors. These models are appealing because the regression coefficients are allowed to vary as a smooth function of some variables of interest to detect nonlinear interactions. Because of their flexibility, VCMs have been widely applied in many scientific areas. See [38] for a selective overview of the major methodological and theoretical developments on VCMs. This study focuses on VCMs for spatial data randomly distributed over an arbitrary geographical region.

Our work is motivated by inference problems examining the effects of the county-level food retail environment on obesity rates in United States, with the effect varying over median household income. County food retail environments are measured by the availability and healthfulness of their food retail stores. More detailed information of this data set is provided in Section 6.6. Based on this data set, socioeconomists attempt to disentangle how county-level associations between the food environment and obesity rates change with median household income levels. This leads to modeling the effect of food retail environments as functions of household income levels. However, owing to the geographic dependence, the classical VCM is not sufficient.

In this work, we propose the varying-coefficient geo model (VCGM) to solve the motivating application. Specifically, assume $\boldsymbol{S}_i = (S_{i1}, S_{i2})^\top$ is the location of the ith subject, for $i = 1, \ldots, n$. The location $\boldsymbol{S}$ ranges over a two-dimensional bounded domain $\Omega \in \mathbb{R}^2$ of any

arbitrary shape. We observe data of the form $\{Y_i, Z_i, \mathbf{X}_i, \boldsymbol{S}_i\}$, where Y_i is a response variable, $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip})^\top$ is a vector of scalar covariates, and Z_i is a scalar predictor. Furthermore, $\{(Y_i, Z_i, \mathbf{X}_i)\}_{i=1}^n$ are observed at location $\boldsymbol{S}_i$. Suppose that $\{(Y_i, Z_i, \mathbf{X}_i, \boldsymbol{S}_i)\}_{i=1}^n$ satisfies the following VCGM:

$$Y_i = \mathbf{X}_i^\top \boldsymbol{\beta}(Z_i) + \alpha(\boldsymbol{S}_i) + \varepsilon_i, \quad \boldsymbol{S}_i \in \Omega, i = 1, \ldots, n, \tag{6.1}$$

where $\boldsymbol{\beta}(Z) = (\beta_1(Z), \ldots, \beta_p(Z))^\top$, with each $\beta_k(\cdot)$ as an unknown varying-coefficient function, $\alpha(\boldsymbol{S}_i)$ is an unknown smoothing bivariate function representing the spatial component, and ε_i denotes independent and identically distributed (i.i.d.) random noise, with $E(\varepsilon_i) = 0$ and $Var(\varepsilon_i) = \sigma^2$ independent of $(Z_i, \mathbf{X}_i, \boldsymbol{S}_i)$. Our primary interest is to estimate and conduct an inference for $\boldsymbol{\beta}(\cdot)$ and $\alpha(\cdot)$ based on the given observations $\{(Y_i, Z_i, \mathbf{X}_i, \boldsymbol{S}_i)\}_{i=1}^n$.

In the proposed VCGM, when the spatial component $\alpha(\cdot)$ is ignored, the model becomes the traditional VCM. Numerous studies have proposed methods for fitting the VCM, for example, the local linear method [37], spline method [48], and two-stage methods [117, 71]. There are also several methods for estimating bivariate functions defined over 2D domains. Within the nonparametric framework, these include bivariate P-splines [77], thin plate splines [122], and bivariate splines [116, 129]. Here, we apply bivariate splines over triangulations [56], because they can handle irregular 2D domains with complex boundaries and they are computationally efficient.

This study focuses on proposing pointwise (at a specific z) and simultaneous (for all $z \in [a, b]$) testing procedures for the following hypothesis under model (6.1):

$$H_0 : H\{\boldsymbol{\beta}_0(z)\} = 0 \text{ v.s. } H_1 : H\{\boldsymbol{\beta}_0(z)\} \neq 0, \tag{6.2}$$

where $H(\mathbf{b})$ is a q-dimensional function of $\mathbf{b} = (b_1, \ldots, b_p) \in \mathbb{R}^p$, such that $\mathbf{C}(\mathbf{b}) := \partial H(\mathbf{b})/\partial \mathbf{b}^\top$ is a $q \times p$ full-rank matrix ($q \leq p$), for all $\mathbf{b}$. The above hypothesis is very general, owing to the choice flexibility of $H(\mathbf{b})$. It includes many interesting hypotheses as special cases, for instance, $H_0 : \beta_{0,k}(z) = 0$ for all k if $H(\mathbf{b}) = \mathbf{b}$, a test for any arbitrary linear

constraints on $\boldsymbol{\beta}_0$ if $H(\mathbf{b}) = \boldsymbol{\Lambda}\mathbf{b} - \mathbf{c}_0$ for a $q \times p$ known matrix $\boldsymbol{\Lambda}$ and a known vector $\mathbf{c}_0$, and even tests with nonlinear constraints. See [6] for explicit examples of nonlinear hypotheses.

In contrast to estimation, few studies have examined inferences of varying-coefficient functions. [48] proposed a goodness-of-fit test based on a comparison of the weighted residual sum of squares. This is a specific example of the generalized likelihood ratio studied by [36]. More recently, [129] proposed a spline backfitted local polynomial to estimate and make simultaneous inferences of the univariate components in a geo-additive model. Although the above-mentioned methods seem useful, they are not applicable to the general hypothesis in (6.2). Furthermore, the testing procedure involves a plug-in variance estimate, which leads to an unstable asymptotic distribution of the test statistics.

In this chapter, we propose both pointwise and simultaneous tests for the hypothesis in (6.2) based on the empirical likelihood (EL). The EL is a nonparametric likelihood, introduced by [82, 83]. In spite of its nonparametric construction based on observed data points, the EL shares some convenient merits of the parametric likelihood, and has many desirable advantages in deriving confidence sets for unknown parameters. [84] and [26] provide an overview of the EL method. The EL method has been extended to VCMs for various data types; see, for example, [124], [123],[125] and [72]. Recently, [114] considered test procedures based on the EL to conduct inferences for a class of functional concurrent linear models. However, when they applied the method to Google flu trend data, they ignored the spatial information contained in the data set. [10] and [111] considered the EL method for inference over a broad class of spatial data exhibiting stochastic spatial patterns. However, they did not consider the flexible VCGM, or the spatial information.

In contrast to existing VCMs, our proposed VCGM properly accounts for all covariates and spatial information, which improves the model flexibility. The proposed EL-based inference has many advantages over normal approximation-based methods. First, it does not involve a plug-in estimate for the limiting variance. Owing to the necessity of estimating the standard errors, which is a typical challenge in nonparametric models, the Wald-type simultaneous inference is not stable in [72]. Second, as [34] proved, the EL is Bartlett correctable and, thus, has

an advantage over the bootstrap method. To the best of our knowledge, this is the first work to propose a VCGM and conduct an EL ratio test for spatial data, which is a nontrivial extension.

The rest of the chapter is organized as follows. We propose spline estimators for both univariate and bivariate functions and develop their asymptotic consistency in Section 6.2. The pointwise and simultaneous EL tests are presented in Section 6.3, where we investigate the asymptotic distributions of the test statistics under both the null hypothesis and local alternatives. In Section 6.4, we address implementation issues such as triangulation, the number of univariate spline knots, and the kernel bandwidth selection. Simulation studies are presented in Section 6.5, followed by an analysis of a real-data example in Section 6.6. We summarize the proposed methodology and discuss future work in Section 6.7. Major technical details are included in the Supplementary Material.

6.2 Univariate and bivariate spline estimations

In the estimation stage, we approximate each varying coefficient using univariate polynomial splines. The geographical function $\alpha(\cdot)$ is approximated using bivariate penalized splines over triangulation. First, we introduce some notation for univariate and bivariate splines.

6.2.1 Setup

Suppose that the covariate Z is distributed on a compact interval $[a, b]$. Owing to the simplicity of the computation, we approximate the univariate components $\beta_k(z)$ in (6.1) using polynomial splines. Define a partition of $[a, b]$ with J_n interior knots as $v = \{a = v_0 \leq v_1 \leq \ldots \leq v_{J_{n+1}} = b\}$. For some $\varrho \geq 1$, the polynomial splines of order $\varrho + 1$ are polynomial functions with ϱ-degree on intervals $[v_j, v_{j+1})$, for $j = 0, \ldots J_n - 1$, and $[v_{J_n}, v_{J_{n+1}}]$, and have $\varrho - 1$ continuous derivatives globally. Let $\mathcal{U} = \mathcal{U}([a, b])$ be the space of such polynomial splines. Let $U_j(z)$, for $j = 1, \ldots, J_n + \varrho + 1$, be the original B-spline basis functions for the coefficient functions. Suppose for $z \in [a, b]$, $\beta_k(z) \approx \sum_{j=1}^{J_n+\varrho+1} \eta_{kj} U_j(z) = \mathbf{U}(z)^\top \boldsymbol{\eta}_k$, where $\mathbf{U}(z) = (U_1(z), \ldots, U_{J_n+\varrho+1}(z))^\top$ and $\boldsymbol{\eta}_k = (\eta_{1k}, \ldots, \eta_{J_n+\varrho+1,k})^\top$.

It has been proved that the bivariate penalized splines method is efficient in dealing with data distributed on irregular domains with complicated boundaries [129, 116]. In the following, we briefly introduce the triangulation techniques and describe the bivariate penalized spline smoothing method for the VCGM. See [56] and [116] for a detailed introduction of the triangulation technique and how to construct the bivariate spline basis functions over triangulation.

According to [56], let $\tau = \langle \mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3 \rangle$ be a nonempty-area triangle with three vertices, $\mathbf{s}_1$, $\mathbf{s}_2$, and $\mathbf{s}_3$. There is a unique representation in the form for any point $\boldsymbol{s} \in \mathbb{R}^2$, $\boldsymbol{s} = b_1\mathbf{s}_1 + b_2\mathbf{s}_2 + b_3\mathbf{s}_3$, with $b_1 + b_2 + b_3 = 1$, and b_1, b_2, and b_3 are the barycentric coordinates of the point $\boldsymbol{s}$ relative to the triangle τ. We define the Bernstein polynomials of degree d relative to triangle τ as $B_{ijk}^{\tau,d}(\boldsymbol{s}) = \frac{d}{i!j!k!} b_1^i b_2^j b_3^k$. The spatial domain Ω is a polygon of arbitrary shape, which can be partitioned into finitely many triangles. Let a collection $\triangle = \{\tau_1, \ldots, \tau_N\}$ of N triangles be a triangulation of $\Omega = \cup_{i=1}^N \tau_i$, provided that any nonempty intersection between a pair of triangles in $\triangle$ is either a shared vertex or a shared edge. For any triangle $\tau \in \triangle$, denote T_τ as the radius of the largest disk contained in τ. Let $|\tau|$ be the length of the longest edge. Denote the size of $\triangle$ as $|\triangle| = \max\{|\tau| : \tau \in \triangle\}$. For any integer $d \geq 1$ and triangle τ, let $\mathbb{P}_d(\tau)$ be the space of all polynomials of degree less than or equal to d on τ. Then, any polynomial $\zeta \in \mathbb{P}_d(\tau)$ can be uniquely written as $\zeta|_\tau = \sum_{i+j+k=d} \gamma_{ijk}^\tau B_{ijk}^{\tau,d}$, where the coefficients $\boldsymbol{\gamma}_\tau = \{\gamma_{ijk}^\tau, i + j + k = d\}$ are called B-coefficients of ζ. For any integer $r \geq 0$, let $\mathbb{C}^r(\Omega)$ be the collection of all rth continuously differentiable functions over Ω. Given a triangulation $\triangle$, define the spline space of degree d and smoothness r over $\triangle$ as $\mathbb{S}_d^r(\triangle) = \{\zeta \in \mathbb{C}^r(\Omega) : \zeta|_\tau \in \mathbb{P}_d(\tau), \tau \in \triangle\}$. Let $\{B_m\}_{m\in\mathcal{M}}$ be the set of bivariate Bernstein basis polynomials for $\mathbb{S}_d^r(\triangle)$, where $\mathcal{M}$ is an index set with cardinality $|\mathcal{M}| = N(d + 1)(d + 2)/2$. Then, we rewrite any function $\zeta \in \mathbb{S}_d^r(\triangle)$ using the basis expansion $\zeta(\boldsymbol{s}) = \sum_{m\in\mathcal{M}} B_m(\boldsymbol{s})\gamma_m = \mathbf{B}(\boldsymbol{s})^\top \boldsymbol{\gamma}$, where $\boldsymbol{s} \in \Omega$ and $\boldsymbol{\gamma} = (\gamma_m, m \in \mathcal{M})^\top$ is the bivariate spline coefficient vector.

6.2.2 Penalized least-squares estimators

In general, there are three approaches to conduct a spline estimation: smoothing splines, regression splines, and penalized splines. Smoothing splines request as many parameters as the

number of observations. Regression splines need only a small number of knots, placed judiciously, but appropriate algorithms are needed to select the knots. Penalized splines combine the features of smoothing splines and regression splines. A roughness penalty is incorporated with relatively large number of knots. [116] and [129] discuss the advantages and necessity of penalized bivariate spline smoothing. Note that, given some suitable smoothness conditions, $\beta_k(\cdot)$ and $\alpha(\cdot)$ can be well represented by a univariate spline basis expansion and the Bernstein basis polynomials introduced in Section 6.2.1. It is well known that increasing the number of triangles may overfit the data and increase the variance, while decreasing the number of triangles may result in a rigid and restrictive function that has more bias. Consequently, to improve the data fitting efficiency, reduce the computation complexity, and avoid over fitting, we consider the following penalized least-squares problem:

$$\sum_{i=1}^{n}\left\{Y_i-\sum_{k=1}^{p}\sum_{j=1}^{J_n+\varrho+1}\eta_{jk}U_j(Z_i)X_{ik}-\sum_{m\in\mathcal{M}}B_m(\boldsymbol{s})\gamma_m\right\}^2+\frac{\lambda_n}{2}\mathcal{E}(\alpha), \tag{6.3}$$

where

$$\mathcal{E}(\alpha)=\sum_{\tau\in\triangle}\int_{\tau}\sum_{i+j=2}\binom{2}{i}(\nabla_{s_1}^{i}\nabla_{s_2}^{j}\alpha)^2ds_1ds_2$$

is the roughness penalty for $\alpha(\cdot)$, λ_n is the roughness penalty parameter, and $\nabla_{s_q}^{v}$ is the vth order derivative in the direction s_q at the point $\boldsymbol{s}$, for $q=1,2$.

For a smooth join between two polynomials on adjoining triangles, we impose some linear constraints on the spline coefficients $\boldsymbol{\gamma}$: $\boldsymbol{\Psi}\boldsymbol{\gamma}=0$, where $\boldsymbol{\Psi}$ is the matrix that collects the smoothness conditions across all the shared edges of triangles. An example of $\boldsymbol{\Psi}$ can be found in [129]. Thus, the penalized least-squares problem (6.3) becomes

$$\sum_{i=1}^{n}\left\{Y_i-\sum_{k=1}^{p}\sum_{j=1}^{J_n+\varrho+1}\eta_{j,k}U_j(Z_i)X_{ik}-\sum_{m\in\mathcal{M}}B_m(\boldsymbol{s})\gamma_m\right\}^2+\frac{1}{2}\lambda_n\boldsymbol{\gamma}^{\top}\mathbf{P}\boldsymbol{\gamma}, \tag{6.4}$$

subject to $\mathbf{\Psi}\boldsymbol{\gamma} = 0$, where $\mathbf{P}$ is the block diagonal penalty matrix satisfying $\boldsymbol{\gamma}^\top \mathbf{P}\boldsymbol{\gamma} = \mathcal{E}(\mathbf{B}\boldsymbol{\gamma})$. In the following, let $\mathbf{Y} = (Y_1, \ldots, Y_n)^\top$ be collections of Y_i. Denote

$$\mathbf{W} = \begin{pmatrix} \mathbf{U}(Z_1)^\top(X_{11}) & \ldots & \mathbf{U}(Z_1)^\top(X_{1p}) \\ \ldots & \ldots & \ldots \\ \mathbf{U}(Z_n)^\top(X_{n1}) & \ldots & \mathbf{U}(Z_n)^\top(X_{np}) \end{pmatrix}$$

as an $n \times p(J_n + \varrho + 1)$ matrix. To solve the constrained minimization problem (6.4), we first remove the constraint using a QR decomposition of the transpose of the constraint matrix $\mathbf{\Psi}$. Specifically, we have $\mathbf{\Psi}^\top = \mathbf{QR} = \begin{pmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{pmatrix} \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{pmatrix}$, where $\mathbf{Q}$ is an orthogonal matrix, $\mathbf{R}$ is an upper-triangle matrix, the submatrix $\mathbf{Q}_1$ is the first r columns of $\mathbf{Q}$, where r is the rank of the matrix $\mathbf{\Psi}$, and $\mathbf{0}$ is a matrix of zeros. According to Lemma 1 in [116], the problem (6.4) is now converted to the following conventional penalized regression problem without any constraints:

$$\min_{\boldsymbol{\eta},\boldsymbol{\theta}} \left\{ \|Y - \mathbf{W}\boldsymbol{\eta} - \mathbf{B}\mathbf{Q}_2\boldsymbol{\theta}\|^2 + \lambda_n (\mathbf{Q}_2\boldsymbol{\theta})^\top \mathbf{P}(\mathbf{Q}_2\boldsymbol{\theta}) \right\},$$

where $\boldsymbol{\eta} = \left(\eta_{11}, \ldots, \eta_{p(J_n+\varrho+1)}\right)$ and $\mathbf{Q}_2\boldsymbol{\theta} = \boldsymbol{\gamma}$. For a fixed penalty parameter λ_n, we have

$$\begin{pmatrix} \widehat{\boldsymbol{\eta}} \\ \widehat{\boldsymbol{\theta}} \end{pmatrix} = \left\{ \begin{pmatrix} \mathbf{W}^\top\mathbf{W} & \mathbf{W}^\top\mathbf{B}\mathbf{Q}_2 \\ \mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{W} & \mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{B}\mathbf{Q}_2 \end{pmatrix} + \begin{pmatrix} \mathbf{0} & \\ & \lambda_n\mathbf{Q}_2^\top\mathbf{P}\mathbf{Q}_2 \end{pmatrix} \right\}^{-1} \begin{pmatrix} \mathbf{W}^\top\mathbf{Y} \\ \mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{Y} \end{pmatrix}.$$

Define

$$\mathbf{V} = \begin{pmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{pmatrix} = \begin{pmatrix} \mathbf{W}^\top\mathbf{W} & \mathbf{W}^\top\mathbf{B}\mathbf{Q}_2 \\ \mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{W} & \mathbf{Q}_2^\top(\mathbf{B}^\top\mathbf{B} + \lambda_n\mathbf{P})\mathbf{Q}_2 \end{pmatrix}.$$

It follows from well-known block matrix forms of a matrix inverse that

$$\mathbf{V}^{-1} := \mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix} = \begin{pmatrix} \mathbf{A}_{11} & -\mathbf{A}_{11}\mathbf{V}_{12}\mathbf{V}_{22}^{-1} \\ -\mathbf{A}_{22}^{-1}\mathbf{V}_{21}\mathbf{V}_{11}^{-1} & \mathbf{A}_{22} \end{pmatrix},$$

where

$$\mathbf{A}_{11}^{-1} = \mathbf{V}_{11} - \mathbf{V}_{12}\mathbf{V}_{22}^{-1}\mathbf{V}_{21} = \mathbf{W}^\top[\mathbf{I} - \mathbf{B}\mathbf{Q}_2\{\mathbf{Q}_2^\top(\mathbf{B}^\top\mathbf{B} + \lambda_n\mathbf{P})\mathbf{Q}_2\}^{-1}\mathbf{Q}_2^\top\mathbf{B}^\top]\mathbf{W}$$
$$\mathbf{A}_{22}^{-1} = \mathbf{V}_{22} - \mathbf{V}_{21}\mathbf{V}_{11}^{-1}\mathbf{V}_{12} = \mathbf{Q}_2^\top[\mathbf{B}^\top\{\mathbf{I} - \mathbf{W}(\mathbf{W}^\top\mathbf{W})^{-1}\mathbf{W}^\top\}\mathbf{B} + \lambda_n\mathbf{P}]\mathbf{Q}_2.$$

Hence, $\widehat{\boldsymbol{\eta}} = \mathbf{A}_{11}\mathbf{W}^\top\left\{\mathbf{I} - \mathbf{B}\mathbf{Q}_2\{\mathbf{Q}_2^\top(\mathbf{B}^\top\mathbf{B} + \lambda_n\mathbf{P})\mathbf{Q}_2\}^{-1}\mathbf{Q}_2^\top\mathbf{B}^\top\right\}\mathbf{Y}$ and $\widehat{\boldsymbol{\theta}} = \mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\{\mathbf{I} - \mathbf{W}(\mathbf{W}^\top\mathbf{W})^{-1}\mathbf{W}^\top\}\mathbf{Y}$. Thus, the estimators of $\beta_k(\cdot)$ and $\alpha(\cdot)$ are

$$\widehat{\beta}_k(z) = \mathbf{U}(z)^\top\widehat{\boldsymbol{\eta}}_k \quad \text{and} \quad \widehat{\alpha}(\boldsymbol{s}) = \mathbf{B}(\boldsymbol{s})^\top\widehat{\boldsymbol{\gamma}}, \text{ respectively, where } \widehat{\boldsymbol{\gamma}} = \boldsymbol{Q}_2\widehat{\boldsymbol{\theta}}. \tag{6.5}$$

We now investigate the asymptotic properties of the spline estimates $\widehat{\beta}_k(z)$ and $\widehat{\alpha}(\boldsymbol{s})$. To avoid confusion, let $\beta_{0,k}(\cdot)$ and $\alpha_0(\cdot)$ be the true functions of $\beta_k(\cdot)$ and $\alpha(\cdot)$ in model (6.5). For any Lebesgue measurable function $\phi(\boldsymbol{s})$ on a domain $\mathcal{D}$, where $\mathcal{D} = [a, b]$ or $\Omega \subseteq \mathbb{R}^2$, let $\|\phi\|_{L_2}^2 = \int_{\mathcal{D}} \phi^2(\boldsymbol{s})d\boldsymbol{s}$.

Theorem 6.1 (Rate of Convergence). *Suppose that Assumptions (A1)–(A6) in the Supplementary Material hold. Then the spline estimators $\widehat{\beta}_k$ and $\widehat{\alpha}$ satisfy*

$$\begin{aligned}
&\|\widehat{\alpha} - \alpha_0\|_{L_2} \\
&= O_p\left\{J_n^{-\varrho-1}|\triangle| + n^{-1/2}|\triangle|^{-1} + \frac{\lambda_n}{n|\triangle|^3} + \left(1 + \frac{\lambda_n}{n|\triangle|^5}\right)|\triangle|^{d+1}\right\}, \\
&\sum_{k=1}^{p}\|\widehat{\beta}_k - \beta_{0,k}\|_{L_2} = O_p\left(n^{-1/2}J_n^{1/2} + n^{-1}|\triangle|^{-1} + J_n^{-\varrho-1}\right).
\end{aligned}$$

REMARK **1.** *This consistency result echoes similar phenomena discovered by other nonparametric regression literature. In fact, when only spatial information is available and no other scale covariates are included, the model (6.1) reduces to the same model in [57]. When the varying coefficients reduce to linear coefficients, model (6.1) reduces to the same model in [116]. In these two reduced models, the above convergence rate of $\widehat{\alpha}$ is the same as those given in [57] and [116], that is, $O_p\left\{n^{-1/2}|\triangle|^{-1} + \frac{\lambda_n}{n|\triangle|^3} + \left(1 + \frac{\lambda_n}{n|\triangle|^5}\right)|\triangle|^{d+1}\right\}$. When the geo function $\alpha(\cdot)$ is excluded from model (6.1), the convergence rate of $\widehat{\beta}_k$ reduces to*

$O_p(n^{-1/2}J_n^{1/2} + J_n^{-\varrho-1})$. *If* $\beta_{0,k}$ *have bounded second-order derivatives* ($\varrho = 1$) *and* $J_n \asymp n^{1/5}$, *we have* $\|\widehat{\beta}_k - \beta_{0,k}\|_{L_2} = O_p(n^{-2/5})$, *achieving the optimal nonparametric rate [101].*

Given these consistency results of the proposed univariate and bivariate spline estimators, we can now build hypothesis testing statistics based on these estimators.

6.3 Empirical likelihood ratio tests for varying coefficients

It is challenging to derive the asymptotic distribution and the measure of variability for the spline estimators introduced in Section 6.2. Similar findings have been discussed in [71] and [129]. To investigate the uncertainty in the estimation of the varying effect of the covariates, we propose an inference for hypothesis (6.2) using the EL method, with bivariate penalized spline estimators plugged in for the geo function.

To test (6.2) and construct an EL ratio function for $\boldsymbol{\beta}(z)$, we first introduce an auxiliary random vector

$$g_i\{\boldsymbol{\beta}(z), \alpha_0\} = \left(Y_i - \boldsymbol{\beta}(z)^\top \boldsymbol{X}_i - \alpha_0(\boldsymbol{S}_i)\right) \boldsymbol{X}_i K_h(Z_i - z),$$

where $K(\cdot)$ stands for a continuous kernel function, h is a bandwidth, and $K_h(\cdot) = K(\cdot/h)/h$ is a rescaling of K. Note that $Eg_i\{\boldsymbol{\beta}(z), \alpha_0\}$ is close to zero if $\boldsymbol{\beta}(z) = \boldsymbol{\beta}_0(z)$. Hence, the problem of testing whether $\boldsymbol{\beta}(z)$ is the true function $\boldsymbol{\beta}_0(z)$ is equivalent to testing whether $Eg_i\{\boldsymbol{\beta}(z), \alpha_0\}$ is close to zero, for $i = 1, 2, \ldots, n$. According to [84], this can be done by using the EL; that is, we can define the profile EL ratio function

$$R\{\boldsymbol{\beta}(z), \alpha_0\} = \max_{p_i: 1 \le i \le n} \left\{ \prod_{i=1}^{n} np_i : 0 \le p_i \le 1, \sum_{i=1}^{n} p_i = 1, \sum_{i=1}^{n} p_i g_i\{\boldsymbol{\beta}(z), \alpha_0\} = 0 \right\}.$$

The rich EL literature has shown that $-2 \log R\{\boldsymbol{\beta}_0(z), \alpha_0\}$ is asymptotically chi-squared with p degrees freedom. However, $R\{\boldsymbol{\beta}(z), \alpha_0\}$ cannot be used directly to make a statistical inference on $\boldsymbol{\beta}(z)$, because $R\{\boldsymbol{\beta}(z), \alpha_0\}$ contains the unknown function $\alpha_0(\cdot)$. A natural way

is to replace $\alpha_0(\cdot)$ with the estimator $\widehat{\alpha}(\boldsymbol{S}_i)$ given in (6.5), that is,

$$g_i\{\boldsymbol{\beta}(z)\} := g_i\{\boldsymbol{\beta}(z), \widehat{\alpha}\} = \left(Y_i - \boldsymbol{\beta}^\top(z)\boldsymbol{X}_i - \widehat{\alpha}(\boldsymbol{S}_i)\right)\boldsymbol{X}_i K_h(Z_i - z).$$

Note that the solution to $\sum_{i=1}^n g_i\{\boldsymbol{\beta}(z)\} = 0$ corresponds to the local constant estimator

$$\check{\boldsymbol{\beta}}(z) = \left\{\sum_{i=1}^n \boldsymbol{X}_i\boldsymbol{X}_i^\top K_h(Z_i - z)\right\}^{-1}\left\{\sum_{i=1}^n (Y_i - \hat{\alpha}(\boldsymbol{S}_i))\boldsymbol{X}_i K_h(Z_i - z)\right\}. \tag{6.6}$$

After replacing the true function $\alpha_0(\cdot)$, we show that the discrepancy between $g_i\{\boldsymbol{\beta}_0(z)\}$ and $g_i\{\boldsymbol{\beta}_0(z), \alpha_0\}$ is asymptotically negligible in the following proposition. Let $\mu_{jj'} = \int u^{j'}K^j(u)du$ and $\boldsymbol{\Omega}(z) = E(\mathbf{X}_1\mathbf{X}_1^\top|Z = z)$.

Proposition 6.1. *Under Assumptions (A1)—(A5), (A6′), (A7), and (A8) in the Supplementary Material, we have*

$$E[g_i\{\boldsymbol{\beta}_0(z)\}] = O(h^2)$$

and

$$Var[g_i\{\boldsymbol{\beta}_0(z)\}] = \sigma^2\boldsymbol{\Omega}(z)f(z)\mu_{20}h^{-1}\left\{1 + o(1)\right\},$$

where $f(z)$ is the probability density function of Z.

REMARK **2.** *To investigate the EL tests for the geo spatial model, the key point is to check the asymptotic property of $g_i\{\boldsymbol{\beta}_0(z)\}$. More specifically, if the first and second moments of $g_i\{\boldsymbol{\beta}_0(z)\}$ have the same orders as those of $g_i\{\boldsymbol{\beta}_0(z), \alpha_0\}$, the asymptotic distribution of $-2\log R\{\boldsymbol{\beta}(z)\}$ is similar to the common VCM cases. According to Theorem 6.1, we establish the orders of the first two moments for $g_i\{\boldsymbol{\beta}_0(z)\}$ as in Proposition 6.1 by bounding $E\left\{\mathbf{X}_i\mathbf{X}_i^\top\left(\boldsymbol{\beta}_0(Z_i) - \boldsymbol{\beta}_0(z)\right)K_h(Z_i - z)\right\}$ and $E\left\{\mathbf{X}_i K_h(Z_i - z)(\alpha_0(\boldsymbol{S}_i) - \widehat{\alpha}(\boldsymbol{S}_i))\right\}$, with a careful choice of the lower bound of J_n and the upper bound of $|\triangle|$. The details can be found in the proof of Proposition 6.1 in the Supplementary Material.*

With a slight abuse of notation, we define the EL function

$$L\{\boldsymbol{\beta}(z)\} = \max_{p_i:1\le i\le n}\left\{\prod_{i=1}^n p_i : 0 \le p_i \le 1, \sum_{i=1}^n p_i = 1, \sum_{i=1}^n p_i g_i\left\{\boldsymbol{\beta}(z)\right\} = 0\right\}. \tag{6.7}$$

We can maximize (6.7) using the Lagrange multiplier technique, which leads to the following log-EL:

$$\log L\{\boldsymbol{\beta}(z)\} = -\sum_{i=1}^{n} \log\left\{1 + \boldsymbol{\delta}^\top(z) g_i\{\boldsymbol{\beta}(z)\}\right\} - n\log n,$$

where $\boldsymbol{\delta}(z)$ is determined by the equation: $\sum_{i=1}^{n} g_i\{\boldsymbol{\beta}(z)\}[1 + \boldsymbol{\delta}^\top(z) g_i\{\boldsymbol{\beta}(z)\}]^{-1} = 0$. Therefore, the negative log-EL ratio statistic for testing $H_0 : H\{\boldsymbol{\beta}_0(z)\} = 0$ is

$$\ell(z) := \min_{H\{\boldsymbol{\beta}(z)\}=0} \sum_{i=1}^{n} \log\left\{1 + \boldsymbol{\delta}^\top(z) g_i\{\boldsymbol{\beta}(z)\}\right\}. \tag{6.8}$$

To investigate the power of the tests, we consider the local alternatives $H_1 : H\{\boldsymbol{\beta}_0(z)\} = b_n \boldsymbol{d}(z)$, where b_n is a sequence of numbers converging to zero and $\boldsymbol{d}(z) \neq 0$ is a q-dimensional function. For any fixed nonzero function $\boldsymbol{d}(z)$, b_n depicts the order of signals that a test can detect. The smallest order of b_n is given in [27], who show that the EL method can detect alternatives of order $(nh)^{-1/2}$ for pointwise tests and order $n^{-1/2}h^{-1/4}$ for simultaneous tests. Both orders are larger than the parametric rate $n^{-1/2}$.

The following theorem summarizes the asymptotic distribution of $2\ell(z)$ under both the local alternative and the null hypothesis H_0 for each fixed z.

Theorem 6.2. *Under Assumptions (A1)—(A5), (A6′), (A7), and (A8) in the Supplementary Material, and for each $z \in [a, b]$ under the null hypothesis: $H\{\boldsymbol{\beta}_0(z)\} = 0$, we have $2\ell(z) \xrightarrow{d} \chi^2_q$. For each $z \in [a, b]$ and any fixed real vector of function $\mathbf{d}(z)$, under the alternative hypothesis $H_1 : H\{\boldsymbol{\beta}_0(z)\} = (nh)^{-1/2}\mathbf{d}(z)$, we have*

$$2\ell(z) \xrightarrow{d} \chi^2_q\left(\boldsymbol{d}^\top(z)\boldsymbol{R}(z)\boldsymbol{d}(z)\right),$$

where $\mathbf{R}(z) = \sigma^2\mu_{20} f(z)\left\{\mathbf{C}(z)\boldsymbol{\Omega}(z)\mathbf{C}^\top(z)\right\}^{-1}$ and

$$\mathbf{C}(z) = \mathbf{C}\left(\boldsymbol{\beta}(z)\right) = \frac{\partial H\left(\boldsymbol{\beta}(z)\right)}{\partial \boldsymbol{\beta}(z)^\top}.$$

According to the Theorem 6.2, we can construct a pointwise confidence interval for each $\beta_j(z)$. The construction of the confidence interval is based on an asymptotic α-level test when

$H\{\boldsymbol{\beta}(z)\} = \beta_j(z)$. We reject H_0 at a fixed point z if $2\ell(z) > \chi^2_{1,\alpha}$, where $\chi^2_{1,\alpha}$ is the upper α-quantile of χ^2_1, and a $100(1-\alpha)\%$ confidence interval for $\beta_j(z)$ is given by $\{\beta_j(z) : 2\ell(z) \leq \chi^2_{1,\alpha}\}$.

For the simultaneous test on H_0 in (6.2), for all $z \in [a, b]$, we consider the Cramér—von Mises type test statistic. Because $2\ell(z)$ can be viewed as the distance between $H\{\boldsymbol{\beta}(z)\}$ and zero, we propose the following test statistic for H_0:

$$D_n = \int_a^b 2\ell(z)w(z)dz, \tag{6.9}$$

where $w(z)$ is some probability weight function.

Theorem 6.3. *Under Assumptions (A1)—(A5), (A6′), (A7), and (A8) in the Supplementary Material, with the null hypothesis* $H_0 : H\{\boldsymbol{\beta}_0(\cdot)\} = 0$, *as* $n \to \infty$, *we have*

$$h^{-1/2}\{D_n - q\} \xrightarrow{d} N\left(0, q\sigma_0^2\right),$$

where $\sigma_0^2 = 2\mu_{20}^{-2}\int_a^b w^2(t)dt\int_{-2}^2\{K^{(2)}(u)\}^2du$. *When the alternative hypothesis* $H_1 : H\{\boldsymbol{\beta}_0(z)\} = n^{-1/2}h^{-1/4}\boldsymbol{d}(z)$ *holds, we have*

$$h^{-1/2}\{D_n - q\} \xrightarrow{d} N\left(\mu_0, q\sigma_0^2\right),$$

where $\mu_0 = \int_a^b \mathbf{d}^\top(z)\mathbf{R}(z)\mathbf{d}(z)w(z)dz$.

Although the above theorem guarantees the asymptotic normality of D_n, the convergence rate is $h^{-1/2}$. According to Assumption (A6′), the rate is $o(n^{1/10})$, which is much slower than the classical nonparametric rate $n^{2/5}$. To obtain accurate type-I and type-II error probabilities in practice, we suggest a bootstrap procedure to generate the empirical quantile and perform the simultaneous testing. The distribution consistency of this method is discussed in [114]. The proposed bootstrap procedure consists of the following steps:

Step 1. For each subject, calculate the residual $\widetilde{e}_i = Y_i - \check{\boldsymbol{\beta}}(Z_i)^\top \boldsymbol{X}_i - \widehat{\alpha}_i(\boldsymbol{S}_i)$, with the local constant estimator $\check{\boldsymbol{\beta}}(z)$ in (6.6). Compute the sample variance of $\widetilde{e}_i$, and denote it as $\widetilde{\sigma}^2$;

Step 2. For the bth bootstrapping, for $b = 1, \ldots, B$, construct observation $Y_i^{(b)} = \check{\boldsymbol{\beta}}(Z_i)^\top \boldsymbol{X}_i + \widehat{\alpha}_i(\boldsymbol{S}_i) + \epsilon_i^{(b)}$, where $\epsilon_i^{(b)}$ are independently generated from a normal distribution satisfying $E\left(\epsilon_i^{(b)}\right) = 0$ and $Var\left(\epsilon_i^{(b)}\right) = \widetilde{\sigma}^2$. Apply $\left\{Y_i^{(b)}\right\}_{i=1}^n$ as new observations, and compute the bootstrapped version of D_n, denoted by $D_n^{(b)}$;

Step 3. Calculate the $100(1-\alpha)\%$ quantile of the bootstrap samples $\left\{D_n^{(b)}\right\}_{b=1}^B$, and denote it as $\widehat{d}_\alpha$. Reject the null hypothesis if $D_n > \widehat{d}_\alpha$.

REMARK **3.** *In step 1, $\check{\boldsymbol{\beta}}(z)$ is the solution to $n^{-1}\sum_{i=1}^n g_i\left(\boldsymbol{\beta}(z), \widehat{\alpha}\right) = 0$. We use $\check{\boldsymbol{\beta}}(z)$ instead of the spline estimator $\widehat{\boldsymbol{\beta}}(z)$ to generate residuals, because $\check{\beta}(z)$ is the maximum empirical likelihood estimator involved in the construction of $\ell(z)$ and D_n.*

The following proposition justifies the bootstrap procedure. The proof is similar to Theorem 4 in [114]. Thus it is omitted.

Proposition 6.2. *Let $\mathcal{X}_n = \{(Y_i, Z_i, \boldsymbol{X}_i, \boldsymbol{S}_i)\}_{i=1}^n$ be the original data, and $\mathcal{L}(D_n)$ be the asymptotic distribution of D_n under the null hypothesis. Under Assumptions (A1)—(A6), (A6′), (A7) and (A8), the conditional distribution of $D_n^{(b)}$ given $\mathcal{X}_n$, $\mathcal{L}\left(D_n^{(b)}|\mathcal{X}_n\right)$ converges to $\mathcal{L}\left(D_n\right)$ almost surely.*

6.4 Implementation

In extensive numerical studies, we find that the selections of the knots for the univariate spline, triangulation, and the choice of bandwidth are crucial, especially for simultaneous tests. In the following, we discuss the selection procedures one by one.

6.4.1 Selection of the tuning parameters in univariate and bivariate spline smoothing

In this work, we do not directly need the spline estimator $\widehat{\boldsymbol{\beta}}(z)$ for the inference of $\boldsymbol{\beta}(z)$. However, $\widehat{\alpha}(\boldsymbol{s})$ is essential for constructing the EL ratio tests (6.8), and its estimating procedure involves $\widehat{\boldsymbol{\beta}}(z)$. Hence, we need to make sure that $\boldsymbol{\beta}(z)$ is estimated efficiently. For univariate spline smoothing, we suggest applying knots on a grid of equally spaced sample quantiles. Assumption (A6′) in the Supplementary Material suggests that the number of

knots J_n needs to satisfy $|\triangle|^{1/(\varrho+1)}\ n^{2/(5\varrho+5)} \ll J_n \ll |\triangle|^2 n\log^{-1}(n)$. Given the widely used cubic splines, in practice, we suggest the rule-of-thumb number of interior knots $J_n = \max\left\{\lfloor c_1 n^{2/(5\varrho+5)}\rfloor + 1, 3\right\}$, where the tuning parameter $c_1 \in [1,3]$. A similar is considered in [129]. We also compared the proposed knot selection method with other data-driven methods, namely, the AIC and BIC. The well-selected parameters using the AIC and BIC are similar to our proposed rule-of-thumb choices. Therefore, for the purpose of efficient computation, we recommend the rule-of-thumb choices for practical applications.

When selecting the number of triangles, we need to balance the computational burden and the approximation accuracy. According to [129] and Assumption (A6′), in practice, when the boundary of the spatial domain is not extremely complicated, we suggest taking the number of triangles as the following: $N = \min\left\{\lfloor c_2 n^{4/(5d+5)}\rfloor, n/4\right\} + 1$, for some tuning parameter c_2. Typically, $c_2 \in [1,5]$ and is chosen using cross-validation. When the boundary of the spatial domain looks complicated, we suggest N to be much larger than n, and the triangulation can approximate the complicated domain precisely. Once N is chosen, a typical triangulation method, such as Delaunay triangulation, can be used to build the triangulated meshes. From our numerical experience, when the smoothness $r = 1$, compared with the setting $d = 2$ or 3, using $d = 5$ requires too much unnecessary computational time, because its improvement in terms of accuracy is negligible. We suggest using $r = 1$ and $d = 2$ or 3 in practice, because they provide enough accuracy for smooth functions and reduce the computational cost. Similar settings are also found in [57], [129] and [51].

The generalized cross-validation (GCV) criterion is an efficient method for selecting the smoothing parameters λ_n, and also has good theoretical properties [112]. The fitted values at the n data points are $\widehat{\mathbf{Y}} = \mathbf{W}\widehat{\boldsymbol{\eta}} + \mathbf{B}\mathbf{Q}_2\widehat{\boldsymbol{\theta}}$, and the smoothing matrix is

$$\begin{aligned}\mathbf{S}(\lambda_n) \;=\; & \mathbf{W}\mathbf{A}_{11}\mathbf{W}^\top\left\{\mathbf{I} - \mathbf{B}\mathbf{Q}_2\{\mathbf{Q}_2^\top(\mathbf{B}^\top\mathbf{B} + \lambda_n\mathbf{P})\mathbf{Q}_2\}^{-1}\mathbf{Q}_2^\top\mathbf{B}^\top\right\} \\ & + \mathbf{B}\mathbf{Q}_2\mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\left\{\mathbf{I} - \mathbf{W}(\mathbf{W}^\top\mathbf{W})^{-1}\mathbf{W}^\top\right\}.\end{aligned}$$

We choose the smoothing parameter λ_n by minimizing

$$GCV(\lambda_n) = n\|\mathbf{Y} - \widehat{\mathbf{Y}}\|^2 / \left[n - tr\left\{\mathbf{S}(\lambda_n)\right\}\right]^2$$

over a grid of values of λ_n. We use a 10-point grid, where the values of $\log_{10}(\lambda_n)$ are equally spaced between -6 and 1 in our numerical studies. The aforementioned bivariate spline smoothing methods are all implemented using the R package "BPST" developed by [116].

6.4.2 Bandwidth selection

The performance of the EL pointwise and simultaneous tests depends on the choice of the bandwidth h. We apply the five-fold cross-validation criterion and choose the bandwidth h by minimizing

$$CV(h) = 5^{-1}\sum_{k=1}^{5}|\mathcal{F}_k|^{-1}\sum_{i\in\mathcal{F}_k}\left\{Y_i - \check{\boldsymbol{\beta}}^{(-k)}(Z_i)^\top\mathbf{X}_i - \widehat{\alpha}^{(-k)}(\boldsymbol{S}_i)\right\}^2,$$

where $\mathcal{F}_k$ denotes the subject index set for the kth folder and $|\mathcal{F}_k|$ denotes the cardinality of $\mathcal{F}_k$ over a grid of values of h. In our numerical studies, we select the bandwidth $h = \lfloor c_3 n^{1/5}\rfloor + 0.02$ for the pointwise tests, and $h = \lfloor c_3 n^{1/5}\rfloor$ for the simultaneous tests, where $c_3 \in \{0.1, 0.2, \ldots, 0.9, 1\}$.

6.5 Simulation

In this section, we conduct simulation studies to evaluate the finite-sample performance of the proposed methodology. We generate the data from the following VCGM:

$$Y_i = X_{i1}\beta_1(Z_i) + X_{i2}\beta_2(Z_i) + \alpha(\boldsymbol{S}_i) + \epsilon_i, i = 1, \cdots, n, \tag{6.10}$$

where X_{ij} and ϵ_i are independently generated from $N(0,1)$, and Z_i follows $Unif[0,1]$ independently. In addition, we choose the Epanechnikov kernel $K(x) = 3/4\left(1-x^2\right)_+$ for the local linear estimation, where $(a)_+ = \max(a, 0)$. The sample sizes are chosen to be

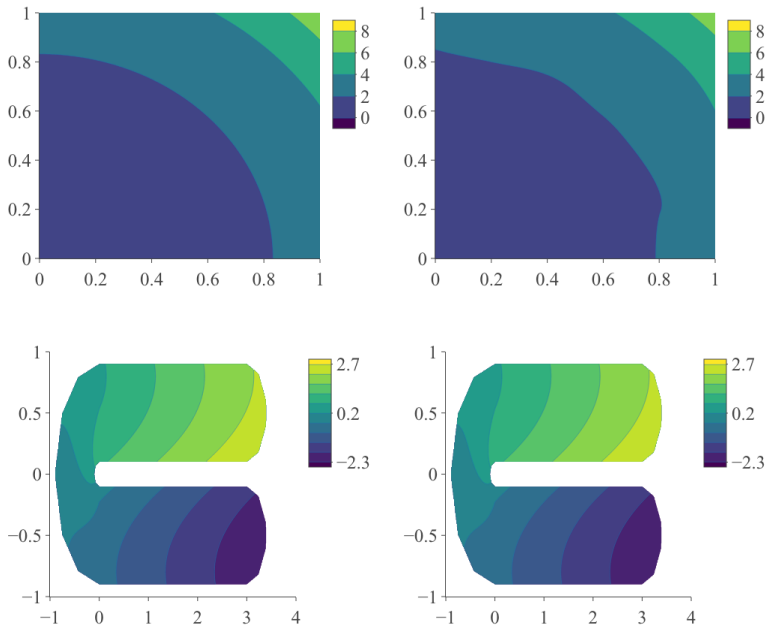

Figure 6.1: Contour maps of the true function $\alpha_0(\cdot)$ (first column) and the estimators (second colmun) over the square region (first row) and the horseshoe region (second row).

$n = 500, 1000, 2000$. We consider two spatial domains for the bivariate function $\alpha(\cdot)$: 1) a rectangular domain $[0, 1]^2$; and 2) a modified horseshoe domain used by [91] and [116]. For each Monte Carlo replication, we randomly sample n locations uniformly from the grid points inside the two spatial domains. Under all scenarios, $1,000$ Monte Carlo replicates are conducted. For all the univariate splines, we use cubic B-splines with $\varrho = 3$. For the bivariate spline smoothing, we consider $d = 3$ and $r = 1$.

To check the accuracy of the proposed spline estimators, we compute the mean squared error (MSE) for α, β_1, and β_2. Figure 6.1 shows the surface and the contour map of the true bivariate function $\alpha(\cdot)$ and the estimated one when the sample size $n = 2,000$. The proposed estimates look visually close to the true functions. Figure 6.2 shows the box plot of the MSEs of the spline estimators for both regions, showing that the MSEs and the corresponding standard deviations decrease as the sample size increases.

We first conduct pointwise hypothesis tests. Let $H\left\{(\beta_1, \beta_2)^\top\right\} = \beta_1 - \beta_2$ to test H_0 : $\beta_1(z) = \beta_2(z)$ versus H_1 : $\beta_1(z) \neq \beta_2(z)$, where we set $\beta_1(z) = (2 + a)\sin(2\pi z)$ and $\beta_2(z) = 2\sin(2\pi z)$, for some nonnegative a in model (6.10), to evaluate the empirical size (when $a = 0$) and power (when $a > 0$) at the 5% nominal level. Figure 6.3 shows the empirical

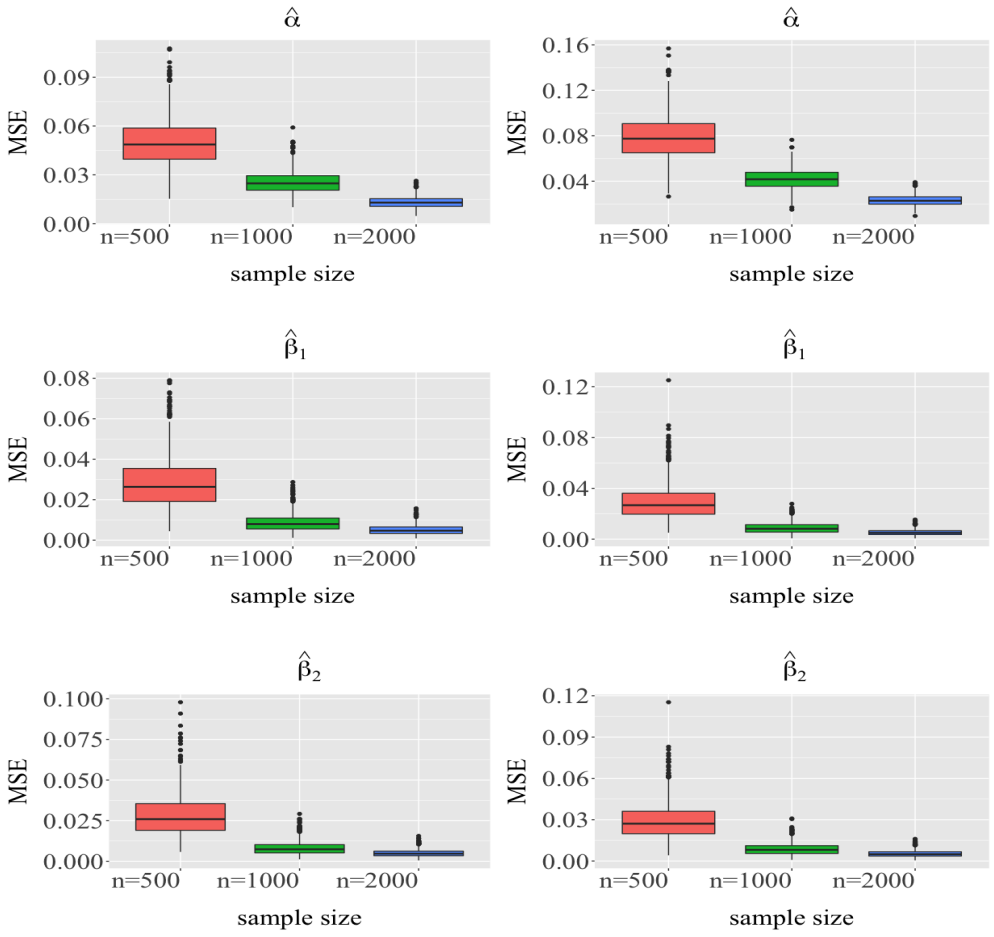

Figure 6.2: Mean squared error of the spline estimators. First column: the square region; Second column: the horseshoe region.

size and power with two different domains of $\alpha(\boldsymbol{s})$ and different $z \in \{0.3, 0.4, 0.6, 0.7\}$. Given each z, the empirical size is reasonably controlled around the nominal level of 5% for all sample sizes, and the power increases with a until reaching one. As expected, a larger sample size leads to greater power.

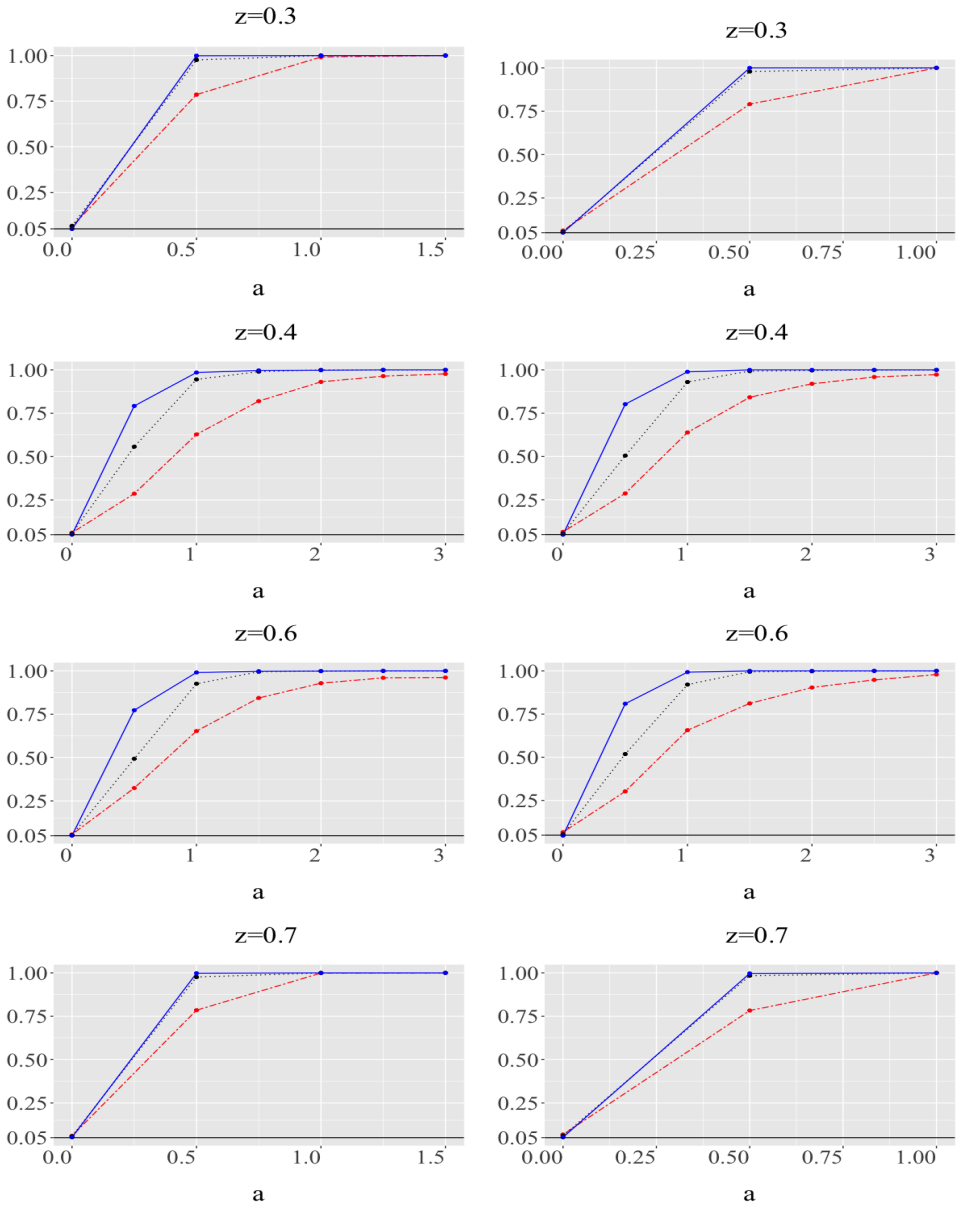

Figure 6.3: Empirical size and power for the pointwise test $H_0 : \beta_1(z) = \beta_2(z)$ at the 5% nominal level. ····: $n = 500$; - - -: $n = 1,000$; —— : $n = 2,000$. First column: square region; Second column: horseshoe region.

Next, we set $\beta_1(z) = 1/2\sin(z)$, $\beta_2(z) = 2\sin(z + 1/2)$ in model (6.10), and apply the procedure in Section 6.3 to construct pointwise confidence intervals for $\beta_1(z)$ at the 95% nominal level. Table 6.1 summarizes the empirical coverage probability (as percentages) and

the average length of the confidence intervals (in parentheses) for $\beta_1(z)$ at $z = 0.3, 0.4, 0.6, 0.7$. From the table, we see that for different z, the coverage rates increase with the sample size, and are around 95% when $n = 2,000$. Furthermore, the length of the confidence intervals decreases as the sample size increases.

Finally, we consider simultaneous inference. We test $H_0 : \beta_1(z) = \beta_2(z)$ for all $z \in [0, 1]$ versus $H_1 : \beta_1(z) \neq \beta_2(z)$, for some z, where we set $\beta_1(z) = (2 + a)\sin(2\pi z)$ and $\beta_2(z) = 2\sin(2\pi z)$ for $a \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$ in model (6.10). We evaluate the empirical size (when $a = 0$) and power (when $a > 0$); the results are presented in Table 6.2. All tests are under two scenarios of bivariate function regions. In the construction of the test statistics D_n, we choose the weight function $w(z) = 1$ for $z \in (0, 1)$, and $w(z) = 0$ otherwise. The critical value of the test is estimated using 500 bootstrap samples in each simulation run. From Table 6.2, we find that the empirical size for each n is around the nominal level of 5%, and the trend of the power is reasonably controlled.

Table 6.1: Coverage rate and average length (in parentheses) of confidence intervals.

	n	$z = 0.3$	$z = 0.4$	$z = 0.6$	$z = 0.7$
Square	500	0.920 (0.265)	0.935 (0.260)	0.934 (0.308)	0.934 (0.262)
	1000	0.931 (0.234)	0.947 (0.233)	0.959 (0.225)	0.947 (0.224)
	2000	0.949 (0.135)	0.944 (0.134)	0.950 (0.165)	0.959 (0.163)
Horseshoe	500	0.938 (0.278)	0.942 (0.272)	0.948 (0.263)	0.945 (0.263)
	1000	0.940 (0.207)	0.951 (0.208)	0.948 (0.206)	0.949 (0.199)
	2000	0.944 (0.156)	0.949 (0.154)	0.951(0.154)	0.949 (0.154)

Table 6.2: Empirical size and power for the simultaneous test $H_0 : \beta_1(\cdot) = \beta_2(\cdot)$.

	n	$a = 0$	$a = 0.1$	$a = 0.2$	$a = 0.3$	$a = 0.4$	$a = 0.5$	$a = 0.6$
Square	500	0.045	0.091	0.274	0.604	0.868	0.984	1
	1000	0.045	0.136	0.572	0.927	0.997	1	1
	2000	0.050	0.262	0.868	1	1	1	1
Horseshoe	500	0.046	0.078	0.280	0.597	0.879	0.975	1
	1000	0.049	0.140	0.561	0.937	0.999	1	1
	2000	0.052	0.256	0.889	0.999	1	1	1

6.6 Real data analysis

The unequal food retail environment (FRE) has been recognized as a critical contextual factor contributing to geographic disparities in obesity. However, there is no clear conclusion on the relationship between the FRE and obesity, owing to diverse measures of the FRE and socioeconomic disparities. In order to resolve this challenge, we include multiple types of food stores, restaurants, and Supplemental Nutrition Assistance Program (SNAP) stores to assess the FRE from two important perspectives: X_1, availability, and X_2, healthfulness. In particular, X_1 is a composite index of the densities of food stores, restaurants, and SNAP stores, and X_2 is a composite index of the ratios of healthy to unhealthy food stores, full service restaurants to fast food restaurants, and healthy to unhealthy SNAP stores. Data are collected from $3,091$ counties in the United States in 2018. For each county, $\boldsymbol{S}_i = (S_{i1}, S_{i2})^\top$ is their geographical location, and Z_i is their median household income. We model the county-level obesity rate (Y) as the following VCGM:

$$Y_i = \beta_0(Z_i) + X_{i1}\beta_1(Z_i) + X_{i2}\beta_2(Z_i) + \alpha(\boldsymbol{S}_i) + \epsilon_i, i = 1, \cdots, 3,091. \tag{6.11}$$

To check whether the two covariates X_1 and X_2 are significant in model (6.11), we first conduct two simultaneous tests $H_{01} : \beta_1(z) = 0$ and $H_{02} : \beta_2(z) = 0$, for all z. For the simultaneous test H_{01}, the test statistic is $D_n = 28.888$, and the 95% quantile of the bootstrap samples is $\widehat{d}_{0.05} = 11.666$; for the simultaneous test H_{02}, the test statistic is $D_n = 85.060$, and the 95% quantile of the bootstrap samples is $\widehat{d}_{0.05} = 11.696$. Hence, both null hypotheses are rejected, indicating that at least for some point z, $\beta_1(z) \neq 0$ and $\beta_2(z) \neq 0$. Next, we investigate the pointwise properties for these varying-coefficient functions. Figure 6.4 shows the 95% pointwise confidence bands and empirical maximum likelihood estimators for $\beta_0(\cdot)$, $\beta_1(\cdot)$, $\beta_2(\cdot)$, and the penalized bivariate spline estimator $\widehat{\alpha}(\cdot)$. From the pointwise confidence bands, we conclude that food availability (X_1) and healthfulness (X_2) have strong nonlinear effects on reducing county obesity rates, given the higher household income level, especially when the income value is larger than USD $100,000$. Interestingly, the pointwise confidence

bands and zero lines together indicate that for those counties with a median household income less than about USD $75,000$, food availability (X_1) has no significant impact on the obesity rate. However, the composite index of healthfulness (X_2) has significant negative impact on the obesity rate of counties with a median household income less than about USD $100,000$. This finding suggests that increasing the value of healthfulness can help to reduce adult obesity rates in counties with a median household income of less than about USD $100,000$. Because there are few counties with a household income greater than USD $100,000$, the confidence bands are much wider in that region. Given the relatively large variation, food availability has a negative effect , and the index of healthfulness has no significant impact on the obesity rate. As expected, Figure 6.4 also indicates that the traditional deep-south states have a large positive geo value $\alpha(\cdot)$, suggesting that these states have higher obesity rates than others with similar FRE values. This reflects that, in addition to the FRE, local food preference, culture, and other factors also influenc county obesity rates.

Because social scientists doubt the association between FRE and obesity may differ with county median household income, $z_0 = 56,516$. We perform the pointwise hypothesis test $H_{0P} : \beta_1(z_0) = \beta_2(z_0)$ versus $H_{1P} : \beta_1(z_0) \neq \beta_2(z_0)$ to test whether availability and healthfulness have the same contribution to obesity rates at z_0. We use cubic B-splines for three univariate splines, and consider $d = 2$ and $r = 1$ for the bivariate spline smoothing. The corresponding pointwise test statistic based on the data is 0.137, which accepts H_{0P}. Thus, we conclude that availability and healthfulness do not have significantly different contributions to obesity rates at the median household income point. For availability and healthfulness, we derive the pointwise confidence intervals separately, which are $[-0.552, 0.099]$ and $[-0.356, -0.235]$, respectively. This indicates that at the 95% significance level, we believe that at $z_0 = 56,516$, availability has no contribution to obesity rates; however, healthfulness has a negative contribution to obesity rates. The results reflect that, compared with availability, healthfulness is a more influential factor shaping the spatial pattern of obesity rates across counties. The associations between obesity rates and both FRE indicators vary greatly with changes in the county median household income and across space.

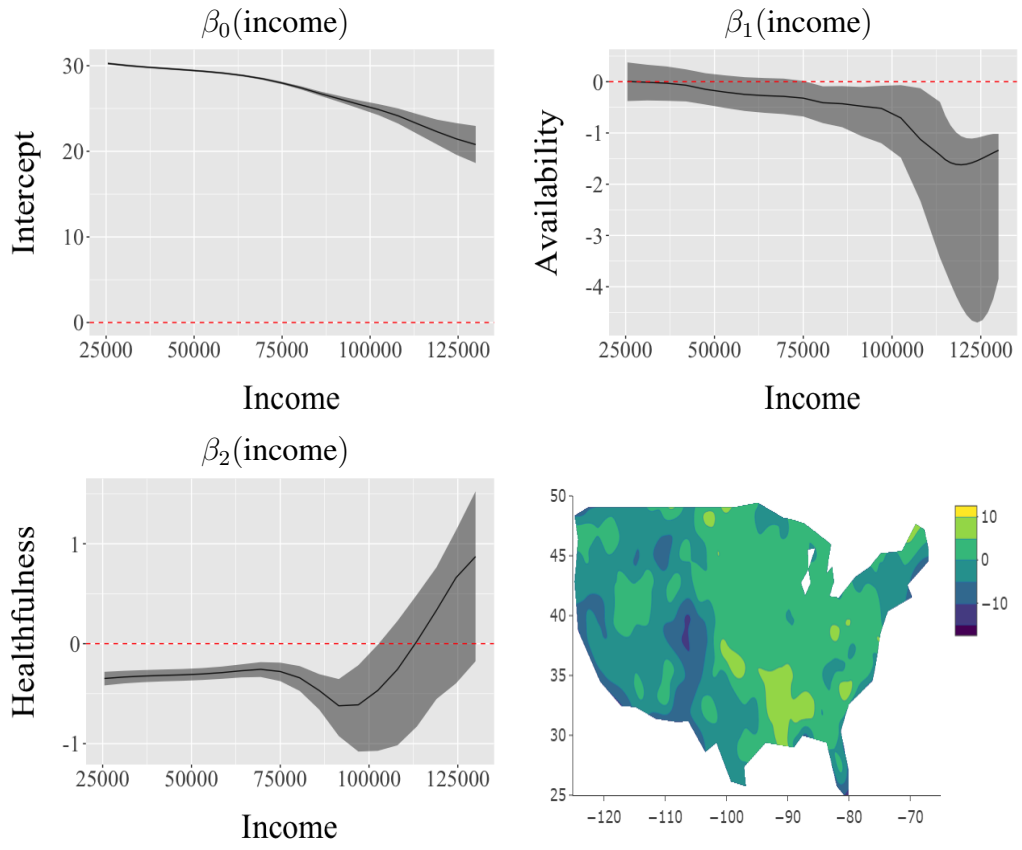

Figure 6.4: 95% pointwise confidence bands for β_0 (top left), β_1 (top right), and β_1 (bottom left) (—: maximum empirical likelihood estimator $\check{\beta}$; - - -: zero line), and the penalized bivariate spline estimator $\widehat{\alpha}$ (bottom right).

6.7 Discussion

In this work, we have proposed both pointwise and simultaneous tests for a general hypothesis in a spatial VCM. Compared with classical VCMs, the proposed VCGM is able to handle spatial information in any regular or irregular 2D domains. Furthermore, regression coefficients are allowed to vary systematically and smoothly in some variables. Owing to the advantages over normal approximation-based methods, the EL method is proposed for conducting the inference. We argue that the proposed hypothesis testing method for the VCGM has attractive properties that have not been investigated.

References

[1] Jorge Adrover, Matias Salibian-Barrera, and Ruben Zamar. Globally robust inference for the location and simple linear regression models. *Journal of Statistical Planning and Inference*, 119(2):353–375, 2004.

[2] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. *Proceedings of the 36th International Conference on Machine Learning*, 97:242–252, 2019.

[3] T. W. Anderson. *An Introduction To Multivariate Statistical Analysis 3rd Ed*. Wiley-Intersceince, New York, 2003.

[4] M. Anthony and P. Bartlett. *Neural Network Learning*. Cambridge University Press, Cambridge, 2009.

[5] Yuko Araki, Sadanori Konishi, Shuichi Kawano, and Hidetoshi Matsui. Functional logistic discrimination via regularized basis expansions. *Communications in Statistics. Theory and Methods*, 38(16-17):2944–2957, 2009.

[6] F. Gregory Ashby. *Statistical Analysis of fMRI Data*. MIT Press, Cambridge, 2011.

[7] Jean-Yves Audibert and Alexandre B. Tsybakov. Fast learning rates for plug-in classifiers. *The Annals of Statistics*, 35(2):608–633, 2007.

[8] Martin Azizyan, Aarti Singh, and Larry Wasserman. Minimax theory for high-dimensional gaussian mixtures with sparse mean separation. *In Advances in Neural Information Processing Systems*, page 2139–2147, 2013.

[9] Juan Lucas Bali, Graciela Boente, David E Tyler, and Jane Ling Wang. Robust functional principal components: A projection-pursuit approach. *Annals of Statistics*, 39(6):2852–2882, 2011.

[10] Soutir Bandyopadhyay, Soumendra Lahiri, and Daniel Nordman. A frequency domain empirical likelihood method for irregularly spaced spatial data. *The Annals of Statistics*, 25(2):519–545, 2015.

[11] B. Bauer and M. Kohler. On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *The Annals of Statistics*, 47:2261–2285, 2019.

[12] J. R. Berrendero, A. Cuevas, and J. L. Torrecilla. On the use of reproducing kernel hilbert spaces in functional classification. *Journal of the American Statistical Association*, 113(523):1210–1218, 2018.

[13] G. Biau, F. Bunea, and M. Wegkamp. Functional classification in hilbert spaces. *IEEE Trans. Info. Theory*, 51:2163–2172, 2005.

[14] G. Biau, F. Cérou, and A. Guyader. Rates of convergence of the functional k-nearest neighbor estimate. *IEEE Trans. Info. Theory*, 56:2034–2040, 2010.

[15] Thijs Bos and Johannes Schmidt-Hieber. Convergence rates of deep relu networks for multiclass classification. *arXiv:2108.00969*, 2021.

[16] M. L. Braun. Accurate error bounds for the eigenvalues of the kernel matrix. *Journal of Machine Learning Research*, 7:2303–2328, 2006.

[17] T. T. Cai and M. Yuan. Optimal estimation of the mean function based on discretely sampled functional data: Phase transition. *The Annals of Statistics*, 39:2330–2355, 2011.

[18] T. Tony Cai and Linjun Zhang. A convex optimization approach to high-dimensional sparse quadratic discriminant analysis. *arXiv:1912.02872*, 2019.

[19] T. Tony Cai and Linjun Zhang. High dimensional linear discriminant analysis: optimality, adaptive algorithm and missing data. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, 81(4):675–705, 2019.

[20] G. Cao, L. Wang, Y. Li, and L. Yang. Oracle-efficient confidence envelopes for covariance functions in dense functional data. *Statistica Sinica*, 26:359–383, 2016.

[21] G. Cao, L. Yang, and D. Todem. Simultaneous inference for the mean function of dense functional data. *Journal of Nonparametric Statistics*, 24:359–377, 2012.

[22] Hervé Cardot. Nonparametric estimation of smoothed principal components analysis of sampled noisy functions. *Journal of Nonparametric Statistics*, 12:503–538, 2000.

[23] F. Chamroukhi and H. Glotin. Mixture model-based functional discriminant analysis for curve classification. *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2012.

[24] K. Chen and H. G. Müller. Modeling repeated functional observations. *Journal of the American Statistical Association*, 107:1599–1609, 2012.

[25] Lu-Hung Chen and Ci-Ren Jiang. Multi-dimensional functional principal component analysis. *Statistics and Computing*, 27(5):1181–1192, 2017.

[26] Songxi Chen and Ingrid Van Keilegom. A review on empirical likelihood methods for regression. *TEST*, 18(3):415—-447, 2009.

[27] Songxi Chen and Ping-shou Zhong. Anova for longitudinal data with missing values. *The Annals of Statistics*, 36(6):3630–3659, 2010.

[28] Jeng-Min Chiou and Pai-Ling Li. Correlation-based functional clustering via subspace projection. *Journal of the American Statistical Association*, 103(484):1684–1692, 2008.

[29] Antonio Cuevas, Manuel Febrero, and Ricardo Fraiman. Robust estimation and classification for functional data via projection-based depth notions. *Computational Statistics*, 22(3):481–496, 2007.

[30] Xiongtao Dai, Hans-Georg Müller, and Fang Yao. Optimal Bayes classifiers for functional data and density ratios. *Biometrika*, 104(3):545–560, 2017.

[31] A. Delaigle and P. Hall. Achieving near-perfect classification for functional data. *Journal of the Royal Statistical Society, Series B*, 74:267–286, 2012.

[32] A. Delaigle, P. Hall, and N. Bathia. Componentwise classification and clustering of functional data. *Biometrika*, 99(2):299–313, 2012.

[33] Aurore Delaigle and Peter Hall. Classification using censored functional data. *Journal of the American Statistical Association*, 108(504):1269–1283, 2013.

[34] Thomas DiCiccio, Peter Hall, and Joseph Romano. Empirical likelihood is Bartlett-correctable. *The Annals of Statistics*, 19(2):1053–1061, 1991.

[35] Jianqing Fan, Cong Ma, and Yiqiao Zhong. A selective overview of deep learning. *arXiv preprint arXiv:1904.05526*, 2019.

[36] Jianqing Fan, Chunming Zhang, and Jian Zhang. Generalized likelihood ratio statistics and Wilks phenomenon. *The Annals of Statistics*, 29(1):153–193, 2001.

[37] Jianqing Fan and Wenyang Zhang. Statistical estimation in varying coefficient models. *The Annals of Statistics*, 27(5):1491–1518, 1999.

[38] Jianqing Fan and Wenyang Zhang. Statistical methods with varying coefficient models. *Statistics and its Interface*, 1(1):179–195, 2008.

[39] Farzan Farnia and David Tse. A minimax approach to supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 4240–4248, Red Hook, NY, USA, 2016.

[40] F. Ferraty and P. Vieu. Curves discrimination: a nonparametric functional approach. *Computational Statistics & Data Analysis*, 44(1-2):161–173, 2003.

[41] Frédéric Ferraty and Philippe Vieu. *Nonparametric functional data analysis: Theory and practice*. Springer Series in Statistics. Springer, New York, 2006.

[42] Pedro Galeano, Esdras Joseph, and Rosa E. Lillo. The Mahalanobis distance for functional data with applications to classification. *Technometrics*, 57(2):281–291, 2015.

[43] P. Hall, H. G. Müller, and J. L. Wang. Properties of principal component methods for functional and longitudinal data analysis. *The Annals of Statistics*, 34:1493–1517, 2006.

[44] Travor J. Hastie and Robert J. Tibshirani. *Generalized Additive Models*. Chapman & Hall/CRC, 1990.

[45] Trevor Hastie and Robert Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society. Series B.*, 55(4):757–796, 1993.

[46] T. Hsing and R. Eubank. *Theoretical foundations of functional data analysis, with an introduction to linear operators*. Wiley Series in Probability and Statistics. John Wiley & Sons, Ltd., Chichester, 2015.

[47] Tianyang Hu, Zuofeng Shang, and Guang Cheng. Sharp rate of convergence for deep neural network classifiers under the teacher-student setting. *arXiv:2001.06892*, 2020.

[48] Jianhua Z. Huang, Colin O. Wu, and Lan Zhou. Varying-coefficient models and basis function approximations for the analysis of repeated measurements. *Biometrika*, 89(1):111–128, 2002.

[49] Goodfellow Ian, Bengio Yoshua, and Courville Aaron. *Deep learning*. MIT Press, 2016.

[50] Ioannis Kalogrids. Asymptotics for m-type smoothing splines with non-smooth objective functions. *arXiv*, page https://arxiv.org/abs/2002.04898, 2020.

[51] Myungjin Kim, Li Wang, and Yuyu Zhou. Spatially varying coefficient models with sign preservation of the coefficient functions. *Journal of Agricultural, Biological and Environmental Statistics*, 26:367–386, 2021.

[52] Yongdai Kim, Ohn Ilsang, and Kim Dongha. Fast convergence rates of deep neural networks for classification. *Neural Networks*, 138:179–197, 2021.

[53] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *In the 3rd International Conference on Learning Representations (ICLR)*, 2015.

[54] P. Kokoszka and M. Reimherr. *Introduction to functional data analysis*. Texts in Statistical Science Series. CRC Press, Boca Raton, FL, 2017.

[55] M. Kosorok. *Introduction to empirical processes and semiparametric Inference*. Springer-Verlag, New York, 2008.

[56] Ming-Jun Lai and Larry L. Schumaker. *Spline functions on triangulations*, volume 110 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2007.

[57] Minjun Lai and Li. Wang. Bivariate penalized splines for regression. *Statistica Sinica*, 23:1399–1417, 2013.

[58] Guillaume Lecué. Classification with minimax fast rates for classes of Bayes rules with sparse representation. *Electronic Journal of Statistics*, 2:741–773, 2008.

[59] Seokho Lee, Hyejin Shin, and Nedret Billor. M-type smoothing spline estimators for principal functions. *Computational Statistics & Data Analysis*, 66:89–100, 2013.

[60] X. Leng and H.G. Müller. Classification using functional data analysis for temporal gene expression data. *Bioinformatics*, 22:68–76, 2006.

[61] Kexuan Li, Fangfang Wang, Ruiqi Liu, Fan Yang, and Zuofeng Shang. Calibrating multidimensional complex ode from noisy data via deep neural networks. *arXiv:2106.03591*, 2021.

[62] Xiuqi Li and Subhashis Ghosal. Bayesian classification of multiclass functional data. *Electronic Journal of Statistics*, 12(2):4669–4696, 2018.

[63] Y. Li and T. Hsing. Uniform convergence rates for nonparametric regression and principal component analysis in functional/longitudinal data. *The Annals of Statistics*, 38:3321–3351, 2010.

[64] E. Lila, J. A. D. Aston, and L. Sangalli. Smooth principal component analysis over two-dimensional manifolds with an application to neuroimaging. *The Annals of Applied Statistics*, 10(4):1854–1879, 2016.

[65] Italo R. Lima, Guanqun Cao, and Nedret Billor. M-based simultaneous inference for the mean function of functional data. *Annals of the Institute of Statistical Mathematics*, 71:577–598, 2019.

[66] Italo R. Lima, Guanqun Cao, and Nedret Billor. Robust simultaneous inference for the mean function of functional data. *TEST*, 28:785—-803, 2019.

[67] Yi Lin. Tensor product space anova models. *The Annals of Statistics*, 28:734 – 755, 2000.

[68] Huan Liu, Yongqiang Tang, and Helen Zhang. A new chi-square approximation to the distribution of non-negative definite quadratic forms in non-central normal variables. *CSDA*, 53:853–856, 2009.

[69] R. Liu, B. Boukai, and Z. Shang. Optimal nonparametric inference via deep neural network. *Preprint*, 2019.

[70] R. Liu, Z. Shang, and G. Cheng. On deep instrumental variables estimate. *arXiv:2004.14954*, 2021.

[71] Rong Liu, Lijian Yang, and Wolfgang K. Härdle. Oracally efficient two-step estimation of generalized additive model. *Journal of the American Statistical Association*, 108(502):619–631, 2013.

[72] Rong Liu and Yichuan Zhao. Empirical likelihood inference for generalized additive partially linear models. *TEST*, 30(3):569–585, 2021.

[73] Ruiqi Liu, Ben Boukai, and Zuofeng Shang. Optimal nonparametric inference via deep neural network. *Journal of Mathematical Analysis and Applications*, 505:125561, 2022.

[74] Yudell L.Luke. Inequalities for generalized hypergeometric functions. *Journal of Approximation Theory*, 5:41–65, 1972.

[75] Enno Mammen and Alexandre B. Tsybakov. Smooth discrimination analysis. *The Annals of Statistics*, 27:1808–1829, 1999.

[76] Ricardo A Maronna and Victor J Yohai. Robust functional linear regression based on splines. *Computational Statistics & Data Analysis*, 65:46–55, 2013.

[77] Brian D. Marx and Paul H. C. Eilers. Multidimensional penalized signal regression. *Technometrics*, 47(1):13–22, 2005.

[78] Santiago Mazuelas, Andrea Zanoni, and Aritz Perez. Minimax classification with 0-1 loss and performance guarantees. *arXiv:2010.07964*, 2020.

[79] J. S. Morris and R. J. Carroll. Wavelet-based functional mixed models. *Journal of the Royal Statistical Society, Series B*, 68:179–199, 2006.

[80] S. J. Morris. Spline estimators for semi-functional linear model. *Annual Review of Statistics and Its Application*, 2:321–359, 2015.

[81] Yangling Mu and Fred H. Gage. Adult hippocampal neurogenesis and its role in alzheimer's disease. *Mol Neurodegener.*, 2011.

[82] Art B. Owen. Empirical likelihood ratio confidence intervals for a single functional. *Biometrika*, 75(2):237–249, 1988.

[83] Art B. Owen. Empirical likelihood ratio confidence regions. *The Annals of Statistics*, 18(1):90–120, 1990.

[84] Art B. Owen. *Empirical likelihood*. CRC Press, Boca Raton, 2001.

[85] Juhyun Park, Jeongyoun Ahn, and Yongho Jeon. Sparse functional linear discriminant analysis. *arXiv:2012.06488*, 2020.

[86] Philipp Petersen and Felix Voigtlaender. Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Networks*, 108:296–330, 2018.

[87] J. O. Ramsay and B. W. Silverman. *Functional Data Analysis, Second Edition*. Springer Series in Statistics, New York, 2005.

[88] J.A. Rice and C.O. Wu. Nonparametric mixed effects models for unequally sampled noisy curves. *Biometrics*, 57:253–259, 2001.

[89] B. D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge, 2014.

[90] Fabric Rossi, Delannay Nicolas, Brieuc Conan-Guez, and Michel Verleysen. Representation of functional data in neural networks. *Neurocomputing*, 64:183–210, 2005.

[91] Laura M. Sangalli, James O. Ramsay, and Timothy O. Ramsay. Spatial spline regression models. *Journal of the Royal Statistical Society. Series B.*, 75(4):681–703, 2013.

[92] J. Schmidt-Hieber. Nonparametric regression using deep neural networks with relu activation function. *arXiv:1708.06633*, 2019.

[93] Carlo Sguera, Pedro Galeano, and Rosa Lillo. Spatial depth-based classification for functional data. *TEST*, 23(4):725–750, 2014.

[94] Z. Shang and G. Cheng. Computational limits of a distributed algorithm for smoothing spline. *Journal of Machine Learning Research*, 18:1–37, 2017.

[95] H. Shin. An extension of fisher's discriminant analysis for stochastic processes. *Journal of Multivariate Analysis*, 99:1191—-1216, 2008.

[96] Hyejin Shin and Seokho Lee. An RKHS approach to robust functional linear regression. *Statistica Sinica*, 26:255–272, 2016.

[97] Barry Simon. Notes on infinite determinants of Hilbert space operators. *Advances in Mathematics*, 24(3):244–273, 1977.

[98] J. Song, W. Deng, H. Lee, and D. Kwon. Optimal classification for time-course gene expression data using functional data analysis. *Biometrika*, 103(1):147–159, 2016.

[99] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

[100] Ingo Steinwart and Andreas Christmann. *Support vector machines*. Springer Science and Business Media, 2008.

[101] Charles J. Stone. Optimal global rates of convergence for nonparametric regression. *The Annals of Statistics*, 10(4):1040–1053, 1982.

[102] Taiji Suzuki. Adaptivity of deep relu network for learning in besov and mixed smooth besov spaces: optimal rate and curse of dimensionality. *arXiv preprint arXiv:1810.08033*, 2018.

[103] J. Taylor. Lecture notes for stats 352: Spatial statistics. 2009.

[104] Barinder Thind, Kevin Multani, and Jiguo Cao. Deep learning with functional inputs. *arXiv preprint arXiv:2006.09590*, 2020.

[105] Barinder Thind, Kevin Multani, and Jiguo Cao. Neural networks as functional classifiers. *arXiv preprint arXiv:2010.04305*, 2020.

[106] J. L. Torrecilla, Carlos Ramos-Carreno, Manuel Sanchez-Montanes, and Suarez Alberto. Optimal classification of gaussian processes in homo- and heteroscedastic settings. *Statistics and Computing*, 30:1091–1111, 2020.

[107] Alexandre B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32:135–166, 2004.

[108] Alexandre B. Tsybakov. *Introduction to nonparametric estimation. Springer Series in Statistics*. Springer, New York, 2009.

[109] A. W. van der Vaart. *Asymptotic statistics*, volume 3 of *Cambridge Series in Statistical and Probabilistic Mathematics*. Cambridge University Press, Cambridge, 1998.

[110] Aad W. van der Vaart and Jon A. Wellner. *Weak convergence and empirical processes*. Springer Series in Statistics. Springer-Verlag, New York, 1996. With applications to statistics.

[111] Matthew Van Hala, Daniel Nordman, and Zhengyuan Zhu. Empirical likelihood for irregularly located spatial data. *Statistica Sinica*, 25(2):1399–1420, 2015.

[112] G. Wahba. Spline models for observational data. *SIAM CBMS-NSF Regional Conference Series in Applied Mathematics*, 59, 1990.

[113] B. Wang, B. Nan, J. Zhu, and R. Koeppe. Regulzarized 3d functional regression for brain image data via haar wavelets. *The Annals of Applied Statistics*, 8:1045–1064, 2014.

[114] Honglang Wang, Ping-Shou Zhong, Yuehua Cui, and Yehua Li. Unified empirical likelihood ratio tests for functional concurrent linear models and the phase transition from sparse to dense functional data. *Journal of the Royal Statistical Society. Series B.*, 80(2):343–364, 2018.

[115] J.L. Wang, J. M. Chiou, and H. G. Müller. Functional data analysis. *Annual Review of Statistics and Its Application*, 3:257–295, 2016.

[116] Li Wang, Guannan Wang, Minjun Lai, and Lei Gao. Efficient estimation of partially linear models for data on complicated domains by bivariate penalized splines over triangulations. *Statistica Sinica*, 30:347–369, 2020.

[117] Li Wang and Lijian Yang. Spline-backfitted kernel smoothing of nonlinear additive autoregression model. *The Annals of Statistics*, 35(6):2474–2503, 2007.

[118] Shuoyang Wang, Guanqun Cao, and Zuofeng Shang. Estimation of the mean function of functional data via deep neural networks. *Stat*, e393, 2021.

[119] Shuoyang Wang, Zuofeng Shang, Guanqun Cao, and Jun S. Liu. Optimal classification for functional data. *arXiv:2103.00569*, 2021.

[120] Shuoyang Wang, Honglang Wang, Yichuan Zhao, Guanqun Cao, and Yingru Li. Empirical likelihood ratio tests for varying coefficient geo models. *Statistica Sinica*, 33(4), 2021+.

[121] Y. Wang, G. Wang, L. Wang, and T. Ogden. Simultaneous confidence corridors for mean functions in functional data analysis of imaging data. *Biometrics*, page In press, 2019.

[122] Simon N. Wood. Thin plate regression splines. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 65(1):95–114, 2003.

[123] L. Xue and Q. Wang. Empirical likelihood for single-index varying-coefficient models. *Bernoulli*, 18:836–856, 2012.

[124] Liugen Xue and Lixing Zhu. Empirical likelihood for a varying coefficient model with longitudinal data. *Journal of the American Statistical Association*, 102(478):642–654, 2007.

[125] Yiping Yang, Gaorong Li, and Heng Peng. Empirical likelihood of varying coefficient errors-in-variables models with longitudinal data. *Journal of Multivariate Analysis*, 127:1–18, 2014.

[126] F. Yao, H. G. Müller, and J. L. Wang. Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association*, 100:577–590, 2005.

[127] F. Yao, H. G. Müller, and J. L. Wang. Functional linear regression analysis for longitudinal data. *The Annals of Statistics*, 33:2873–2903, 2005.

[128] Dmitry Yarotsky. Error bounds for approximations with deep relu networks. *Neural Network*, 94:103–114, 2021.

[129] Shan Yu, Guannan Wang, Li Wang, Chenhui Liu, and Lijian Yang. Estimation and inference for generalized geoadditive models. *J. Amer. Statist. Assoc.*, 115(530):761–774, 2020.

[130] Lingsong Zhang, Haipeng Shen, and Jianhua Z. Huang. Robust regularized singular value decomposition with application to mortality data. *The Annals of Applied Statistics*, 7(3):1540–1561, 2013.

[131] Lan Zhou and Huijun Pan. Principal component analysis of two-dimensional functional data. *Journal of Computational and Graphical Statistics*, 23(3):779–801, 2014.

Appendices

Appendix A

Estimation of the Mean Function of Functional Data via Deep Neural Networks

A.1 Examples

A.1.1 Example 1

Let $\mathbf{X}_j = (j_1/N_d, \ldots, j_d/N_d)$, $1 \leq j_k \leq N_d$ for $k = 1, \ldots, d$, be the evenly spaced grid points of $[0,1]^d$, where $N_d = N^{1/d}$, $d \geq 1$. Consider a Bernoulli polynomial kernel function $G_0(x, x') = 2\sum_{k=1}^{\infty} \frac{\cos(2\pi k(x-x'))}{(2\pi k)^{\varrho d}}$, $x, x' \in [0,1]$, where $\varrho > 1$. See [112] for an introduction of such kernel. For $k = 1, \ldots, d$, the kernel matrix on the k-th coordinate of $\mathbf{X}_j$ is $\mathbf{C}_{N,k} = \{N^{-1}G_0(j_k/N_d, j_k'/N_d)\}_{j_k, j_k'=1}^{N_d}$. Assume that the covariance matrix $N\mathbf{C}_N$ has an additive structure such that $\mathbf{C}_N = \sum_{k=1}^{d} \mathbf{C}_{N,k}$. We require certain order of grid points by sorting them based on the d-th coordinate values first, then by the $(d-1)$-th coordinate values, and so on, until we reach the first coordinate. Let $\mathbf{A}_{N_d}$ be an $N_d \times N_d$ matrix whose (ℓ, ℓ')-th entry is $2N_d^{-1}\sum_{k=1}^{\infty} \frac{\cos(2\pi k(\ell-\ell')/N_d)}{(2\pi k)^{\varrho d}}$, $\ell, \ell' = 1, \ldots, N_d$, and $\mathbf{1}_{N_d}$ be the all-ones $N_d \times N_d$ matrix. Then we have the following relationship:

$$
\begin{aligned}
&\mathbf{C}_{N,1} = N_d^{1-d} \times \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d} \otimes \ldots \otimes \mathbf{1}_{N_d} \otimes \mathbf{A}_{N_d}, \\
&\mathbf{C}_{N,2} = N_d^{1-d} \times \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d} \otimes \ldots \otimes \mathbf{A}_{N_d} \otimes \mathbf{1}_{N_d}, \\
&\ldots\ldots, \\
&\mathbf{C}_{N,d-1} = N_d^{1-d} \times \mathbf{1}_{N_d} \otimes \mathbf{A}_{N_d} \otimes \ldots \otimes \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d}, \\
&\mathbf{C}_{N,d} = N_d^{1-d} \times \mathbf{A}_{N_d} \otimes \mathbf{1}_{N_d} \otimes \ldots \otimes \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d},
\end{aligned}
$$

where $\otimes$ is the kronecker product operator. According to equation (20) in [94], $\mathbf{A}_{N_d}$ is a circulant matrix whose eigenvalues have an explicit expression:

$$\lambda_j^* = \begin{cases} 2\sum_{k=1}^{\infty} \frac{1}{(2\pi k N_d)^{\varrho d}}, & j = 0 \\ \sum_{k=1}^{\infty} \frac{1}{[2\pi(kN_d - j)]^{\varrho d}} + \sum_{k=0}^{\infty} \frac{1}{[2\pi(kN_d + j)]^{\varrho d}}, & 1 \le j \le N_d - 1. \end{cases}$$

In the Appendix C, we have shown that $\max_{j=1,\ldots,N_d} \lambda_j^* \lesssim N_d^{-\varrho d}$. Since the maximal eigenvalue of $\mathbf{1}_{N_d}$ is N_d, by the property of Kronecker product, the maximal eigenvalue of $\mathbf{C}_{N,k}$ is $O(N^{-\varrho})$. Consequently, the first largest eigenvalue for $\mathbf{C}_N$ is $\lambda_{1,N} \lesssim N^{-\varrho}$. According to Assumption (A3), this ensures the better convergence rate in equation (2.7). When $N \gg n^{\frac{1}{\varrho\theta}}$, the convergence rate of $\widehat{f}$ is faster than n^{-1}.

A.1.2 Example 2

Define a cosine random process $\Lambda_k(2\pi x) = \xi_k \cos(2\pi x) + \xi_k' \sin(2\pi x)$, where ξ_k and ξ_k' are identically distributed and uncorrelated, with mean zero and covariance $\mathbb{E}\xi^2$. According to [103], the covariance function for cosine process is given by

$$G_0\left(j_k/N_d, j_k'/N_d\right) = \mathbb{E}\xi^2 \cos\left(2\pi\left(j_k - j_k'\right)/N_d\right)$$

and

$$G_0(\mathbf{X}_j, \mathbf{X}_{j'}) = d^{-1}\mathbb{E}\xi^2 \sum_{k=1}^{d} \cos\left(2\pi\left(j_k - j_k'\right)/N_d\right),$$

which is the (j, j')-th entry in covariance matrix $\mathbf{C}_N$.

Therefore, $\mathbf{C}_N$ can be written as $\mathbf{C}_N = \sum_{k=1}^{d} \mathbf{C}_{N,k}$, where $\mathbf{C}_{N,k}$ is the kernel matrix for the k-th coordinate of $\mathbf{X}_j$, with (j, j')-th entry $N^{-1}\cos\left(2\pi\left(j_k - j_k'\right)/N_d\right)$. Let $\mathbf{B}_{N_d}$ be an $N_d \times N_d$ matrix whose (ℓ, ℓ')-th entry is $N_d^{-1}\cos\left(2\pi\left(\ell - \ell'\right)/N_d\right)$, $\ell, \ell' = 1, \ldots, N_d$. Similar to Example 1, we require the certain order of the grid points and thus have the following

relationship:

$$\begin{aligned}
\mathbf{C}_{N,1} &= N_d^{1-d} \times \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d} \otimes \ldots \otimes \mathbf{1}_{N_d} \otimes \mathbf{B}_{N_d}, \\
\mathbf{C}_{N,2} &= N_d^{1-d} \times \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d} \otimes \ldots \otimes \mathbf{B}_{N_d} \otimes \mathbf{1}_{N_d}, \\
&\ldots\ldots, \\
\mathbf{C}_{N,d-1} &= N_d^{1-d} \times \mathbf{1}_{N_d} \otimes \mathbf{B}_{N_d} \otimes \ldots \otimes \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d}, \\
\mathbf{C}_{N,d} &= N_d^{1-d} \times \mathbf{B}_{N_d} \otimes \mathbf{1}_{N_d} \otimes \ldots \otimes \mathbf{1}_{N_d} \otimes \mathbf{1}_{N_d},
\end{aligned}$$

Since $\mathbf{B}_{N_d}$ is a circulant matrix, its maximal eigenvalue λ_1^* can be explicitly written as

$$N_d^{-1} \sum_{k=0}^{N_d-1} \cos\left(2\pi k/N_d\right) \omega^{N_d-k},$$

where $\omega = \exp\left(2\pi\sqrt{-1}/N_d\right)$. By direct calculations, it can be shown that $\lambda_1^* = N_d/2$. Since the maximal eigenvalue of $N_d^{-1}\mathbf{1}_{N_d}$ is 1, by the property of Kronecker product, it follows that the maximal eigenvalue of $\mathbf{C}_{N,k}$ is $1/2$. Consequently, the maximal eigenvalue of $\mathbf{C}_N$ is $\lambda_{1,N} \asymp \mathbb{E}(\xi^2)/2 = O(1)$. According to the trivial case ($\varrho = 0$) in Assumption (A3), we have the usual nonparametric convergence rate for $\|\widehat{f} - f_0\|_N^2$ as $O(n^{-\frac{\theta}{\theta+1}} \log^6 n)$.

A.1.3 Implementation of Example 1

We first prove two facts regarding the eigenvalues λ_0^*, λ_1^*, …, $\lambda_{N_d-1}^*$ of circulant kernel matrix $\mathbf{A}_{N_d}$:

Fact 1: $\lambda_0^* \leq \lambda_j^* = \lambda_{N_d-j}^*$ for all $j = 1, \ldots, N_d - 1$.

Fact 2: $\lambda_j^* \leq \lambda_1^* = \lambda_{N_d-1}^*$ for all $j = 2, \ldots, N_d - 2$.

For $\varrho > 1$ and $d \geq 1$ according to equation (20) in [94], for $1 \leq j \leq N_d - 1$, we have

$$\lambda_j^* \;=\; \sum_{k=1}^{\infty} \frac{1}{[2\pi(kN_d - j)]^{\varrho d}} + \sum_{k=1}^{\infty} \frac{1}{[2\pi(kN_d - (N_d - j))]^{\varrho d}}.$$

It's trivial that $\lambda_j^* = \lambda_{N_d-j}^*$. Denote $\widetilde{\lambda}_j = \sum_{k=1}^{\infty} [2\pi(kN_d - j)]^{-\varrho d} + \sum_{k=1}^{\infty} [2\pi(kN_d + j)]^{-\varrho d}$, then $\widetilde{\lambda}_j \leq \lambda_j^*$. Let $F_k(a) = (kN_d - a)^{-\varrho d} + (kN_d + a)^{-\varrho d}$, where $k \geq 1$, $a \in [0, N_d - 1]$, then $\widetilde{\lambda}_j = (2\pi)^{-\varrho d} \sum_{k=1}^{\infty} F_k(a)$ and more specifically, $\lambda_0^* = (2\pi)^{-\varrho d} \sum_{k=1}^{\infty} F_k(0)$.

Since $F_k'(a) = \varrho d \left[(kN_d - a)^{-\varrho d-1} - (kN_d + a)^{-\varrho d-1}\right] \geq 0$, we conclude that $F_k(0) \leq F_k(j)$ for arbitrary $k \geq 1$ and $j = 0, 1, \ldots, N_d - 1$, which implies that $\lambda_0^* \leq \widetilde{\lambda}_j \leq \lambda_j^*$. Thus Fact 1 is proved.

Now denote

$$H(a) = \sum_{k=1}^{\infty} \frac{1}{[2\pi(kN_d - a)]^{\varrho d}} + \sum_{k=1}^{\infty} \frac{1}{\{2\pi[kN_d - (N_d - a)]\}^{\varrho d}},$$

where $a \in [1, N_d - 1]$. Then we have

$$\begin{aligned} H'(a) &= \frac{\varrho d}{(2\pi)^{\varrho d}} \left\{ \sum_{k=1}^{\infty} \frac{1}{(kN_d - a)^{\varrho d+1}} + \sum_{k=1}^{\infty} \frac{1}{[kN_d - (N_d - a)]^{\varrho d+1}} \right\} \\ &= \frac{\varrho d}{(2\pi)^{\varrho d}} \sum_{k=1}^{\infty} (h_k(a) - h_k(N_d - a)), \end{aligned}$$

where $h_k(a)$ is a monotone increasing function. For all $k = 1, 2, \ldots$, $h_k(a) - h_k(N_d - a) > 0$ when $a > N_d/2$, $h_k(a) - h_k(N_d - a) < 0$ when $a < N_d/2$, thus we have $a = N_d/2$ is the global maximum for $a \in [1, N_d - 1]$. By Fact 1, it's easy to conclude that λ_1^* and $\lambda_{N_d-1}^*$ are the two largest eigenvalues among all λ_j^*, $j = 1, 2, \ldots, N_d - 1$. Fact 2 is proved.

Combining the above two facts, we can conclude that λ_1^* and $\lambda_{N_d-1}^*$ are the two largest eigenvalues among all $j = 0, 1, \ldots, N_d - 1$.

Secondly, we show the upper bound for the the largest eigenvalues. Without loss generality, when $j = 1$ and $\varrho d > 0$, by equation (20) in [94], we have

$$\begin{aligned}\max_{j=1,\ldots,N_d} \lambda_j^* &= \sum_{k=1}^{\infty} \frac{1}{[2\pi(kN_d - 1)]^{\varrho d}} + \sum_{k=0}^{\infty} \frac{1}{[2\pi(kN_d + 1)]^{\varrho d}} \\ &\leq \frac{1}{[2\pi(N_d - 1)]^{\varrho d}} + \frac{1}{(2\pi)^{\varrho d}} + \sum_{k=2}^{\infty} \frac{1}{[2\pi(kN_d - N_d)]^{\varrho d}} + \sum_{k=1}^{\infty} \frac{1}{(2\pi k N_d)^{\varrho d}} \\ &= \frac{1}{[2\pi(N_d - 1)]^{\varrho d}} + \frac{1}{(2\pi)^{\varrho d}} + 2\sum_{k=1}^{\infty} \frac{1}{(2\pi k N_d)^{\varrho d}} \asymp N_d^{-\varrho d}.\end{aligned}$$

In the following two propositions, we introduce the relationship between eigenvalues of Mercer kernel and eigenvalues of the corresponding kernel matrix. We follow the same notations in [16]. Define a Mercer kernel as $K(x, x') = \sum_{k=1}^{\infty} \lambda_k \psi_k(x)\psi_k(x')$, and its corresponding kernel matrix $\mathbf{K}_N$ with (j, j')-th entry as $[\mathbf{K}_N]_{j,j'} = N^{-1}K(X_j, X_{j'})$. Denote the eigenvalues of $\mathbf{K}_N$ by $\omega_1 \geq \omega_2 \geq \ldots \geq \omega_N \geq 0$. Given a truncation point κ, define the decomposed kernel $K_{[\kappa]}$ and error function E_κ as:

$$K_{[\kappa]}(x, x') = \sum_{k=1}^{\kappa} \lambda_k \psi_k(x)\psi_k(x'), \quad E_\kappa(x, x') = K(x, x') - K_{[\kappa]}(x, x').$$

The kernel matrix induced by E_κ is denoted by $\mathbf{E}_{\kappa,N}$. Let $\boldsymbol{\psi}_{\kappa,N}$ be the $N \times \kappa$ matrix and its (j, l)-th entry $[\boldsymbol{\psi}_{\kappa,N}]_{j,l} = N^{-1/2}\psi_l(X_j)$. The following propositions provide the upper bound of the difference between ω_j and λ_j.

Proposition A.1. *(Theorem 3 in [16]) Let K be a Mercer kernel with bounded eigenfunctions* $|\psi_k(x)| \leq M < \infty$ *for all* k *and* $x \in [0, 1]$. *Then, for* $1 \leq \kappa \leq N$, *with probability larger than* $1 - \epsilon$,

$$\|\boldsymbol{C}_{\kappa,N}\| < M^2\kappa\sqrt{2N^{-1}\log[\epsilon^{-1}\kappa(\kappa + 1)]}, \quad \|\mathbf{E}_{\kappa,N}\| < M^2 \sum_{k=\kappa+1}^{\infty} \lambda_k,$$

where $\boldsymbol{C}_{\kappa,N} = {\boldsymbol{\psi}_{\kappa,N}}^{\top}\boldsymbol{\psi}_{\kappa,N} - \mathbf{I}_\kappa$ *and* $\mathbf{I}_\kappa$ *is a* $\kappa \times \kappa$ *identity matrix. Consequently,*

$$|\omega_j - \lambda_j| = O\left(\lambda_j\kappa\sqrt{\log \kappa N^{-1}} + \sum_{k=\kappa+1}^{\infty} \lambda_k\right), 1 \leq j \leq N.$$

Proposition A.2. *Covariance function $G(\cdot,\cdot)$ is a Mercer kernel. The j-th eigenvalue $\lambda_{j,N}$ of the kernel matrix $\mathbf{C}_N$ from sample $\{\mathbf{X}_j\}_{1\le j\le N}$ satisfies:*

$$\lambda_{j,N} = O\left\{\lambda_j\left(1+\kappa\sqrt{N^{-1}\log\kappa}+\sum_{k=\kappa+1}^{\infty}\lambda_k\right)\right\}.$$

Proof. The result is obtained from Proposition A.1 directly. □

Proposition A.2 establishes the relation between eigenvalues of covariance matrix and eigenvalues of its corresponding covariance function.

A.2 Technical lemmas and proofs

We first introduce some lemmas which is useful in the proof of Theorem 2.1. Without loss of generality, we assume that the radii of the Hölder balls K_i satisfy $K_i \ge 1$, $i = 0,\dots,q$. Let $h_{0j} = 2^{-1}\left(K_0^{-1}g_{0j}(\mathbf{x})+1\right)$, $h_{ij}(\mathbf{x}) = 2^{-1}\left(K_i^{-1}g_{ij}(2K_{i-1}\mathbf{x}-K_{i-1})+1\right)$, $i = 1,\dots,q-1$, and $h_{qj}(\mathbf{x}) = g_{qj}(2K_{q-1}\mathbf{x}-K_{q-1})$, where $K_{i-1}\mathbf{x} = (K_{i-1}x_1,\dots,K_{i-1}x_{d_{i+1}})^\top$. It follows that $h_{0j} \in \mathcal{C}_{t_0}^{\beta_0}([0,1]^{t_0},1)$, $h_{ij} \in \mathcal{C}_{t_i}^{\beta_i}([0,1]^{t_i},(2K_{i-1})^{\beta_i})$ for $i = 1,\dots,q-1$, and $h_{qj} \in \mathcal{C}_{t_q}^{\beta_q}([0,1]^{t_q},K_q(2K_{q-1})^{\beta_q})$.

A.2.1 Definition

Let t_i be the maximal number of variables on which each of the g_{ij} depends on, and $t_i \le d_i$. Define the ball of β-Hölder functions with radius K as

$$\begin{aligned}\mathcal{C}_d^{\beta}(D,K) = \quad & \{\, f : D\subset\mathbb{R}^d\to\mathbb{R} : \\ & \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|<\beta}\|\partial^{\boldsymbol{\alpha}}f\|_\infty + \sum_{\boldsymbol{\alpha}:|\boldsymbol{\alpha}|=\lfloor\beta\rfloor}\sup_{\mathbf{x},\mathbf{y}\in D,\mathbf{x}\ne\mathbf{y}}\frac{|\partial^{\boldsymbol{\alpha}}f(\mathbf{x})-\partial^{\alpha}f(\boldsymbol{y})|}{|\mathbf{x}-\mathbf{y}|_\infty^{\beta-\lfloor\beta\rfloor}} \le K \,\},\end{aligned}$$

where $\partial^{\boldsymbol{\alpha}} = \partial^{\alpha_1}\dots\partial^{\alpha_d}$ with $\boldsymbol{\alpha} = (\alpha_1,\dots,\alpha_d)\in\mathbb{N}^d$ and $|\boldsymbol{\alpha}| := |\boldsymbol{\alpha}|_1$. We assume each g_{ij} is β_i-Hölder function with radius K_i (the definition is given in the Appendix). Since g_{ij} is also

t_i-variate, the true underlying function space becomes

$$\mathcal{G}\left(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K}\right) := \{\ f = g_q \circ \ldots \circ g_0 : \quad g_i = (g_{ij})_j : [a_i, b_i]^{d_i} \to [a_{i+1}, b_{i+1}]^{d_{i+1}}\,,$$
$$g_{ij} \in \mathcal{C}_{t_i}^{\beta_i}\left([a_i, b_i]^{t_i}, K_i\right), |a_i|, |b_i| \leq K_i\ \} \quad \text{(A1)}$$

with $\mathbf{d} := (d_0, \ldots, d_{q+1})$, $\mathbf{t} := (t_0, \ldots, t_q)$, $\boldsymbol{\beta} := (\beta_0, \ldots, \beta_q)$, $\mathbf{K} := (K_0, \ldots, K_q)$ and $\beta_i^* := \beta_i \prod_{k=i+1}^{q} (\beta_k \wedge 1)$.

LEMMA **A.1.** *For any function $h_{ij} \in \mathcal{C}_{t_i}^{\beta_i}([0,1]^{t_i}, K_i)$, $i = 0, 1, \ldots, q$, $j = 1, \ldots, d_{i+1}$ and any constant $M \geq \max_i (\beta_i + 1)^{t_i} \vee (K_i + 1) e^{t_i}$, there exists a network*

$$\widetilde{h}_{ij} \in \mathcal{F}(L_i', (t_i, 6(t_i + \lceil \beta_i \rceil) M, 6(t_i + \lceil \beta_i \rceil) M, \ldots, 1), s_i, \infty)$$

with depth

$$L_i' = 8 + (D_0 \log M + 5)(1 + \lceil \log(t_i \vee \beta_i) \rceil),$$

and numbers of parameters

$$s_i \leq 141 (t_i + \beta_i + 1)^{3 + t_i} M (D_0 \log M + 6),$$

such that

$$\|\widetilde{h}_{ij} - h_{ij}\|_{L^\infty([0,1]^{t_i})} \leq C_i M^{-\frac{\beta_i}{t_i}}, \quad \text{(A2)}$$

where $D_0 = 2 \max_i \frac{t_i + \beta_i}{\beta_i}$ and $C_i = 2\left\{[(2K_i + 1)(1 + t_i^2 + \beta_i^2) 6^{t_i}] \vee (K_i 3^{\beta_i})\right\}$.

Proof. According to Theorem 5 in [92], as the approximation error is determined by $(2K_i + 1)(1 + t_i^2 + \beta_i^2) 6^{t_i} M 2^{-m} + K_i 3^{\beta_i} M^{-\frac{\beta_i}{t_i}}$, and we will choose the optimal m to achieve the best overall error rate. When $m = 2 \max_i \frac{t_i + \beta_i}{\beta_i} \log M$, then $M 2^{-m} \leq M^{-\frac{\beta_i}{t_i}}$ for any i. Thus, $\|\widetilde{h}_{ij} - h_{ij}\|_{L^\infty([0,1]^{t_i})} \leq C_i M^{-\frac{\beta_i}{t_i}}$ for any $i = 0, \ldots, q$ and $j = 1, \ldots, d_{i+1}$. □

Lemma A.1 establishes a network with certain numbers of hidden layers and active parameters, which approximates each h_{ij} with the error rate free of sample size and number of design points. For fixed M, when number of hidden layer and active parameters increases linearly, the

approximation error of the neural network decreases exponentially. Since M is only bounded below, choosing a large enough M can ensure a satisfied approximation error.

LEMMA **A.2.** *For any composite regression function in the class $\mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K})$ and large M, such that $M \geq \max_{i=0,\ldots,q}\{(\beta_i+1)^{t_i} \vee (K_i+1)e^{t_i}\}$, there exists an estimator $f^* \in \mathcal{F}(L, \mathbf{p}, s, F)$ such that*

$$\inf_{f^*\in\mathcal{F}} \|f^* - f_0\|_\infty^2 \leq C_I M^{-\theta}, \tag{A3}$$

where $\theta = \min_{i=0,\ldots,q} \frac{2\beta_i^*}{t_i}$ and C_I is a constant. The parameters of the neural network f^* satisfies:

(i) $L \leq D_0 D_1 \log M - 1$;

(ii) $V^2 := \{\prod_{l=0}^{L+1}(p_l+1)\}^2 \leq 4(d+1)^2(12rM)^{2D_0D_1\log M}$;

(iii) $s \leq D_0 D_2 M \log M - 1$,

where $D_0 = 2\max_i \frac{t_i+\beta_i}{\beta_i}$, $D_1 = 4\max_i(1+\lceil\log(t_i\vee\beta_i)\rceil)(q+1)$, $D_2 = 290\sum_{i=0}^q d_{i+1}(t_i+\beta_i+1)^{3+t_i}$ and $r = \max_i d_{i+1}(t_i+\lceil\beta_i\rceil)$.

Proof. Define

$$h_{ij}^* = 1 - \sigma(1-\widetilde{h}_{ij}),\ 0 \leq i < q,$$

where $\widetilde{h}_{ij}$ is defined in (A2). In this step, h_{ij}^* is obtained by adding another two hidden layers on $\widetilde{h}_{ij}$. This procedure ensures that h_{ij}^* is bounded by 1. Thus, the output $\sigma(h_{ij}^*) = (h_{ij}^* \vee 0) \wedge 1$, for all $0 \leq i < q$. Consequently, according to Lemma A.1, we have

$$\|\sigma(h_{ij}^*) - h_{ij}\|_{L^\infty([0,1]^{t_i})} \leq C_i M^{-\frac{\beta_i}{t_i}},$$

For notation simplicity, define $\sigma(h_q) = h_q$. According to Lemma 3 in [92], the composite network $f^* = h_q^* \circ \sigma(h_{q-1}^*) \circ \ldots \circ \sigma(h_0^*)$ satisfies

$$\begin{aligned} & \left\|\sigma(h_q^*)\circ\sigma(h_{q-1}^*)\circ\ldots\circ\sigma(h_0^*) - h_q\circ\ldots\circ h_0\right\|_{L^\infty([0,1]^d)} \\ \leq\ & K_q \prod_{l=0}^{q-1}(2K_i)^{\beta_{l+1}} \sum_{i=0}^{q} \left\| |\sigma(h_i^*) - \widetilde{h}_i|_\infty \right\|_{L^\infty[0,1]^{d_i}}^{\prod_{l=i+1}^q \beta_l+1}. \end{aligned}$$

Thus, f^* has the approximation error

$$\inf_{f^* \in \mathcal{F}(L,\mathbf{p},s,F)} \|f^* - f_0\|_\infty^2 \leq C_I \max_{i=0,\ldots,q} M^{-\frac{2\beta_i^*}{t_i}},$$

with some constant C_I.

To ensure the above approximation error, we need to figure out the value of L, $\mathbf{p}$ and s with respect to M. According to Lemma A.1 and equation (25) in [92], by the fact that $D_0 \log M \geq 6$, we have

$$\begin{aligned}
L+1 &= 3(q-1) + \sum_{i=0}^{q} L_i' + 1 \\
&\leq 3(q+1) + 8(q+1) + \max_i\{1 + \lceil \log(t_i + \beta_i) \rceil\}(q+1)(D_0 \log M + 5) \\
&= 3(q+1) + 8(q+1) + D_1'(q+1)(D_0 \log M + 5) \\
&= \{(D_0 \log M + 5)D_1' + 11\}(q+1) \\
&\leq 4D_1'(q+1)D_0 \log M = D_0 D_1 \log M,
\end{aligned}$$

and $s + 1 \leq 145M(D_0 \log M + 6)\sum_{i=0}^{q} d_{i+1}(t_i + \beta_i + 1)^{3+t_i} \leq D_0 D_2 M \log M$, and $V := \prod_{l=0}^{L+1}(p_l + 1) \leq 2(d+1)(6rM+1)^{L-1} \leq 2(d+1)(12rM)^{D_0 D_1 \log M}$. □

Lemma A.2 establishes the upper bound of the approximation error, which only depends on M. This bound directly controls the first part in equation (2.6). According to Lemma A.2, the approximation error between $\mathcal{F}(L, \mathbf{p}, s, F)$ and $\mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K})$ can be bounded if we choose proper L, s and $\mathbf{p}$. For fixed M, the value of L, s and V can be exactly calculated. Notice that the derived bounds for L, s and V in Lemma A.2 are just for simplicity, since we want more clear forms connecting these network parameters with M. Obviously, there is no harm to arbitrarily increase these values to achieve the same approximation error, therefore we can substitute L, s and V by the bounds given in Lemma A.2. These bounds will directly control the covering number of the network class in later proofs.

Under the normed space $(\mathcal{F}, \|\cdot\|_N)$, the relation between covering number and packing number is given by:

$$\mathcal{D}(2\delta, \mathcal{F}, \|\cdot\|_N) \leq \mathcal{N}(\delta, \mathcal{F}, \|\cdot\|_N) \leq \mathcal{D}(\delta, \mathcal{F}, \|\cdot\|_N). \tag{A4}$$

For simplicity, denote $\mathcal{N}(\delta) := \mathcal{N}(\delta, \mathcal{F}, \|\cdot\|_N)$ and $\mathcal{D}(\delta) := \mathcal{D}(\delta, \mathcal{F}, \|\cdot\|_N)$ as the δ-covering number and δ-packing number respectively, under the normed space $(\mathcal{F}, \|\cdot\|_N)$.

LEMMA **A.3.** *Under Assumptions (A2) and (A3), given some $\varpi > 0$, $\varrho \geq 0$ and $\vartheta > \ln 2$, with probability $1 - 2e^{-\vartheta}$, we have*

$$\sup_{f,g\in\mathcal{F}:\|f-g\|_N\leq\delta} \frac{1}{N}\sum_{j=1}^{N} (f(\mathbf{X}_j) - g(\mathbf{X}_j))\,\overline{\rho}_{\cdot j} \leq \vartheta\sqrt{9\varpi^2\delta^2(nN^{\varrho})^{-1}\log(\mathcal{N}(\delta))}. \tag{A5}$$

Proof. Define $Z(f) = \frac{1}{N}\sum_{j=1}^{N}\overline{\rho}_{\cdot j} f(\mathbf{X}_j)$. Let $N\mathbf{C}_N$ be the covariance matrix of $(\eta_{i1}, \ldots, \eta_{iN})$ and $\mathbf{\Omega}_N$ be the covariance matrix of $(\epsilon_{i1}, \ldots, \epsilon_{iN})$, where $i = 1, 2, \ldots, n$. Since $Z(f) - Z(g) = \frac{1}{N}\sum_{j=1}^{N}\overline{\rho}_{\cdot j}(f-g)(\mathbf{X}_j)$, and $(\rho_1, \ldots, \rho_N)^\top \sim \mathbf{N}(\mathbf{0}, N\mathbf{C}_N + \mathbf{\Omega}_N)$, for $\forall f, g \in \mathcal{F}$, define $\Delta^2(f,g) = \frac{1}{N^2}\sum_{j,j'=1}^{N}(f-g)(\mathbf{X}_j)(f-g)(\mathbf{X}_{j'})\,\mathrm{Cov}(\overline{\rho}_{\cdot j}, \overline{\rho}_{\cdot j'})$, we have

$$Z(f) - Z(g) \sim \mathbf{N}\left(\mathbf{0}, \Delta^2(f,g)\right).$$

Thus, we have

$$\begin{aligned} P(|Z(f) - Z(g)| \geq x) &= P\left(\frac{|Z(f) - Z(g)|}{\Delta(f,g)} \geq \frac{x}{\Delta(f,g)}\right) \\ &\lesssim \exp\left(-\frac{x^2}{2\Delta^2(f,g)}\right). \end{aligned}$$

Define $\boldsymbol{\gamma}^\top = ((f-g)(\mathbf{X}_1), (f-g)(\mathbf{X}_2), \ldots, (f-g)(\mathbf{X}_N))$. By Assumption (A2), $\Delta^2(f,g)$ can be written as

$$\Delta^2(f,g) = (nN^2)^{-1}\boldsymbol{\gamma}^\top(N\mathbf{C}_N + \mathbf{\Omega}_N)\boldsymbol{\gamma} \lesssim (nN)^{-1}\boldsymbol{\gamma}^\top\mathbf{C}_N\boldsymbol{\gamma} \leq \lambda_{1,N}(nN)^{-1}\boldsymbol{\gamma}^\top\boldsymbol{\gamma}.$$

By Assumption (A3), $\lambda_{1,N} = O(N^{-\varrho})$, where $\varrho \geq 0$. Therefore,

$$\Delta^2(f,g) \lesssim N^{-\varrho}(nN)^{-1}\boldsymbol{\gamma}^\top\boldsymbol{\gamma} = (nN^{\varrho})^{-1}(N^{-1}\boldsymbol{\gamma}^\top\boldsymbol{\gamma}) = (nN^{\varrho})^{-1}\|f-g\|_N^2.$$

By Lemma 8.1 in [55],

$$\left\|Z(f)-Z(g)\right\|_{\psi_2} \lesssim \sqrt{\Delta^2(f,g)} = \Delta(f,g) \lesssim (nN^{\varrho})^{-1/2}\|f-g\|_N,$$

where $\|\cdot\|_{\psi_2} = \inf\{c>0 : E[\psi_2(c^{-1}|\cdot|)] \leq 1\}$ and $\psi_2(x) = \exp(x^2)-1$. Since f and g are arbitrary, by Theorem 8.4 in [55], with semimetric $\|\cdot\|_N$, we have for any $\delta > 0$, there exits a constant $\varpi > 0$ such that,

$$\left\| \sup_{f,g\in\mathcal{F}:\|f-g\|_N\leq\delta} \sqrt{nN^{\varrho}}|Z(f)-Z(g)| \right\|_{\psi_2} \leq \varpi\left[\int_0^\delta \psi_2^{-1}(\mathcal{D}(\epsilon))d\epsilon + \delta\psi_2^{-1}(\mathcal{D}^2(\delta))\right].$$

It follows Lemma 8.1 in [55] again, we have

$$\Pr\left(\sup_{f,g\in\mathcal{F}:\|f-g\|_N\leq\delta} \sqrt{nN^{\varrho}}|Z(f)-Z(g)| > x \right) \leq 2\exp\left(-\frac{x^2}{\varpi^2 J^2(\delta)}\right),$$

where $J(\delta) = \int_0^\delta \sqrt{\log((\mathcal{D}(\epsilon)+1)}d\epsilon + \delta\sqrt{\log(\mathcal{D}^2(\delta)+1)}$, by equation (A4) and Lemma 5 in [92], let $\Delta = \frac{\delta}{2(L+1)V^2}$, then we have the following:

$$\begin{aligned}
\int_0^\delta \sqrt{\log(\mathcal{D}(\epsilon)+1)}d\epsilon &\leq \int_0^\delta \sqrt{\log(\mathcal{N}(\epsilon/2)+1)}d\epsilon \leq \sqrt{2}\int_0^\delta \sqrt{\log(\mathcal{N}(\epsilon/2))}d\epsilon \\
&\leq (L+1)V^2\sqrt{2s+2}\int_0^\Delta \sqrt{\log(\epsilon^{-1})}d\epsilon \\
&\leq (L+1)V^2\Delta\sqrt{2s+2}\sqrt{\log(\Delta^{-1})} \\
&= \delta\sqrt{2^{-1}(s+1)\log(\Delta^{-1})} = \delta 2^{-1/2}\sqrt{\log(\mathcal{N}(\delta))},
\end{aligned}$$

and

$$\begin{aligned}\delta\sqrt{\log(\mathcal{D}^2(\delta)+1)} &\leq \delta\sqrt{2\log(\mathcal{D}(\delta))} \leq \delta\sqrt{2\log(\mathcal{N}(\delta/2))} \\ &\leq \delta\sqrt{2(s+1)\log\left(2(L+1)V^2\delta^{-1}\right)} \\ &= \delta\sqrt{2(s+1)\log\left(\Delta^{-1}\right)} = \sqrt{2}\delta\sqrt{\log(\mathcal{N}(\delta))}.\end{aligned}$$

□

Define $\mathcal{F}_\delta = \mathcal{F}(L, \mathbf{p}, s, \delta)$ in equation (2.4). Lemma A.3 establishes the upper bound for the second item in equation (2.6) under network class $\mathcal{F}_\delta$, which contains the $\widehat{f} - f^*$. The upper bound depends on the sample size n, the number of design points N, covariance parameter ϱ and covering number of $\mathcal{F}_\delta$. Especially, it depends on δ. Notice that if δ is getting smaller, the upper bound is getting smaller for those functions in the space $\mathcal{F}_\delta$. We will use this property to achieve a sharper convergence rate in Lemma A.4.

LEMMA **A.4.** *Under Assumptions (A2) and (A3), after T iterations, with probability* $\left(1-2e^{-\vartheta}\right)^{T+1}$, *we have*

$$\sup_{f,g\in\mathcal{F}:\|f-g\|_N\leq\delta_T} \frac{1}{N}\sum_{j=1}^{N}\left(f(\mathbf{X}_j)-g(\mathbf{X}_j)\right)\overline{\rho}_{\cdot j} \leq C_{n,N}^{1-2^{-T-1}} F^{2^{-T}} \left\{\prod_{j=0}^{T}\log(\mathcal{N}(\delta_j))\right\}^{2^{-T-1+j}},$$

where $C_{n,N} = (nN^\varrho)^{-1}144\varpi^2\vartheta^2$. *More specifically, if* $T = \lceil\log(nN^\varrho)\rceil$, $\vartheta = \log(nN^\varrho)$, *then with probability* $(1-\frac{2}{nN^\varrho})^{\lceil\log(nN^\varrho)\rceil+1}$ *goes to 1,*

$$\sup_{f,g\in\mathcal{F}:\|f-g\|_N\leq\delta_T} \frac{1}{N}\sum_{j=1}^{N}\left(f(\mathbf{X}_j)-g(\mathbf{X}_j)\right)\overline{\rho}_{\cdot j} \leq C_{II}\frac{\prod_{j=0}^{T}\left\{\log(\mathcal{N}(\delta_j))\right\}^{2^{-T-1+j}}}{nN^\varrho}\log^3(nN^\varrho),$$

where $C_{II} = 288\varpi^2\vartheta^2F^{2^{-\lceil\log(nN^\varrho)\rceil}}$.

Proof. Set the initial norm of the difference $\|\widehat{f} - f^*\|_N \leq \delta_0 = F$, if we choose M discussed in Lemma A.2 large enough, thus the approximation error (A3) is small enough. According to equation (2.6), Lemma A.2 and equation (A5) in Lemma A.3, with probability $1-2e^{-\vartheta}$ we

have

$$\|\widehat{f} - f^*\|_N^2 \leq \{C_{n,N} \log(\mathcal{N}(\delta_0))\}^{1/2} \delta_0,$$

under the event set $\mathcal{A}_0 = \{\|f - f^*\|_N^2 \leq \delta_0^2\}$.

The iteration is based on the conclusion in Lemma A.3. Notice that M can be chosen extremely small, thus we can always assume that until the last step of iteration, the upper bound for the first item is always smaller than the second item in equation (2.6).

Step 1 Let $\delta_1^2 = \{\log(C_{n,N}\mathcal{N}(\delta_0))\}^{1/2} \delta_0$, and $\delta_1 \leq \delta_0$. Under event $\mathcal{A}_1 = \{\|f - f^*\|_N^2 \leq \delta_1^2\}$, considering $\mathcal{A}_{1|0} = \{\|f - f^*\|_N^2 \leq \delta_1^2 \mid \|f - f^*\|_N^2 \leq \delta_0^2\}$, then with probability $(1 - 2e^{-\vartheta})^2$, we have

$$\begin{aligned}
\|\widehat{f} - f^*\|_N^2 &\leq \{C_{n,N} \log(\mathcal{N}(\delta_1))\}^{1/2} \delta_1 \\
&= C_{n,N}^{3/4} \{\log(\mathcal{N}(\delta_1))\}^{1/2} \{\log(\mathcal{N}(\delta_0))\}^{1/4} \delta_0^{1/2} \\
&= C_{n,N}^{3/4} \prod_{j=0}^{1} \{\log(\mathcal{N}(\delta_j))\}^{2^{j-2}} \delta_0^{1/2}.
\end{aligned}$$

Step 2 Let $\delta_2^2 = C_{n,N}^{3/4}\{\log(\mathcal{N}(\delta_1))\}^{1/2}\{\log(\mathcal{N}(\delta_0))\}^{1/4}\delta_0^{1/2}$ and $\delta_2 \leq \delta_1$. Under event $\mathcal{A}_2 = \{\|f - f^*\|_N^2 \leq \delta_2^2\}$, considering $\mathcal{A}_{2|1} = \{\|f - f^*\|_N^2 \leq \delta_2^2 \mid \|f - f^*\|_N^2 \leq \delta_1^2\}$, then with probability $(1 - 2e^{-\vartheta})^3$, we have

$$\begin{aligned}
\|\widehat{f} - f^*\|_N^2 &\leq \{C_{n,N} \log(\mathcal{N}(\delta_2))\}^{1/2} \delta_2 \\
&= C_{n,N}^{7/8} \{\log(\mathcal{N}(\delta_2))\}^{1/2} \{\log(\mathcal{N}(\delta_1))\}^{1/4} \{\log(\mathcal{N}(\delta_0))\}^{1/8} \delta_0^{1/4} \\
&= C_{n,N}^{7/8} \prod_{j=0}^{2} \{\log(\mathcal{N}(\delta_j))\}^{2^{j-3}} \delta_0^{1/4}.
\end{aligned}$$

.....

Step t Let $\delta_t^2 = C_{n,N}^{1-2^{-t}} \prod_{j=0}^{t-1} \{\log(\mathcal{N}(\delta_j))\}^{2^{-t+j}} \delta_0^{2^{-t+1}}$, since $\delta_t \leq \delta_{t-1}$. Under the event $\mathcal{A}_t = \{\|f - f^*\|_N^2 \leq \delta_t^2\}$, considering $\mathcal{A}_{t|t-1} = \left\{\|f - f^*\|_N^2 \leq \delta_t^2 \mid \|f - f^*\|_N^2 \leq \delta_{t-1}^2\right\}$, then

with probability $(1-2e^{-\vartheta})^{t+1}$, we have

$$\|\widehat{f}-f^*\|_N^2 \;\leq\; C_{n,N}^{1-2^{-t-1}}\prod_{j=0}^{t}\left\{\log(\mathcal{N}(\delta_j))\right\}^{2^{-t-1+j}}\delta_0^{2^{-t}}.$$

.....

Step T Let $\delta_T^2 = C_{n,N}^{1-(\frac{1}{2})^T}\prod_{j=0}^{T-1}\left\{\log(\mathcal{N}(\delta_j))\right\}^{2^{-T+j}}\delta_0^{2^{-T+1}}$. As $\delta_T \leq \delta_{T-1}$, under event $\mathcal{A}_T = \{\|f-f^*\|_N^2 \leq \delta_T^2\}$, considering $\mathcal{A}_{T|T-1} = \left\{\|f-f^*\|_N^2 \leq \delta_T^2 \mid \|f-f^*\|_N^2 \leq \delta_{T-1}^2\right\}$, then with probability $(1-2e^{-\vartheta})^{T+1}$, we have

$$\|\widehat{f}-f^*\|_N^2 \;\leq\; C_{n,N}^{1-2^{-T-1}}\prod_{j=0}^{T}\left\{\log(\mathcal{N}(\delta_j))\right\}^{2^{-T-1+j}}\delta_0^{2^{-T}}.$$

If choose $T = \lceil\log(nN^{\varrho})\rceil$ and $\vartheta = \log(nN^{\varrho})$, then when $n \to \infty$ and $N \to \infty$, $(1-\frac{2}{nN^{\varrho}})^{\lceil\log(nN^{\varrho})\rceil+1} \to 1$ and $(nN^{\varrho})^{2^{T+1}} \leq \log(nN^{\varrho})$. Hence, according to equation (A5),

$$\begin{aligned}&\sup_{f,g\in\mathcal{F}:\|f-g\|_N\leq\delta_T}\frac{1}{N}\sum_{j=1}^{N}\left(f(\mathbf{X}_j)-g(\mathbf{X}_j)\right)\overline{\rho}_{\cdot j}\\ \leq\;& C_{n,N}F^{2^{-\lceil\log(nN^{\varrho})\rceil}}\prod_{j=0}^{T}\left\{\log(\mathcal{N}(\delta_j))\right\}^{2^{-T-1+j}}\log^3(nN^{\varrho}).\end{aligned}$$

□

According to Lemma A.4, as $\delta_T < F$, there exists a smaller network function class $\mathcal{F}_{\delta_T} \subset \mathcal{F}$, such that it contains $\widehat{f}-f^*$. This network rules out those functions with distance to f^* larger than δ_T and reduces the complexity of the original network class $\mathcal{F}$. In another word, it narrows down the scope of $\widehat{f}$. Therefore, we only consider $\widehat{f}$ in a smaller space, and this gives a narrower bound for the second item in (2.6).

A.3 Proof of Theorem 2.1

Proof. According to Lemma A.2 the first item on the right hand side of equation (2.6) is bounded by $C_I M^{-\theta}$. According to Lemma A.4, the upper bound of the second item of equation

(2.6) is

$$C_{II}(nN^{\varrho})^{-1}\prod_{j=0}^{T}\{\log(\mathcal{N}(\delta_j))\}^{(\frac{1}{2})^{T+1-j}}\log^3(nN^{\varrho}).$$

According to Lemma A.4, and $\log(\mathcal{N}(\delta_j))$ is increasing as δ_j decreasing, we have

$$\|\widehat{f}-f_0\|_N^2 \le C_I M^{-\theta} + C_{II}(nN^{\varrho})^{-1}\log(\mathcal{N}(\delta_{\lceil\log(nN^{\varrho})\rceil}))\log^3(nN^{\varrho}).$$

By the Lemmas 5 in [92] and the fact that $\forall g, g^* \in \mathcal{F}$, $\|g-g^*\|_N \le \|g-g^*\|_\infty$, we have $\mathcal{N}(\delta,\mathcal{F},\|\cdot\|_N) \le (s+1)\log(2\delta^{-1}(L+1)V^2)$. Thus,

$$\log(\mathcal{N}(\delta_{\lceil\log(nN^{\varrho})\rceil})) \le (s+1)\log\left(2\delta^{-1}_{\lceil\log(nN^{\varrho})\rceil}(L+1)V^2\right),$$

where $\delta_{\lceil\log(nN^{\varrho})\rceil} = C_I^{\frac{1}{2}}M^{-\frac{\theta}{2}}$. According to Lemma A.2, such that

$$\begin{aligned}
&(s+1)\log\left(2\delta^{-1}_{\lceil\log(nN^{\varrho})\rceil}(L+1)V^2\right)\\
\le\ &(D_0D_2M\log M)\log(8(d+1)^2C_I^{-\frac{1}{2}}D_0D_1)\\
&+\log(M^{\frac{\theta}{2}}\log M)+2D_0D_1\log(12r)\log M+2D_0D_1\log^2 M \le (DD_0D_2)M\log^3 M,
\end{aligned}$$

where $D = 4\left\{[\log(8(d+1)^2C_I^{-\frac{1}{2}}D_0D_1)]\vee\frac{\theta+2}{2}\vee(2D_0D_1\log(12r))\right\}$.

When $M = (nN^{\varrho})^{\frac{1}{\theta+1}}$, the convergence rate for $\|\widehat{f}-f_0\|_N^2$ is $c(nN^{\varrho})^{-\frac{\theta}{\theta+1}}\log^6(nN^{\varrho})$, where the constant $c = 2\{C_I\vee[C_{II}DD_0D_2(\theta+1)^3]\}$. □

Appendix B

Robust Deep Neural Network Estimation for Multi-dimensional Functional Data

For any vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^N$, define the scaled N-inner product $\langle \boldsymbol{a}, \boldsymbol{b} \rangle_N = N^{-1} \sum_{j=1}^N a_j b_j$ and the associated N-norm $\|\boldsymbol{a}\|_N = \sqrt{N^{-1} \sum_{j=1}^N a_j^2}$.

B.1 Technical lemmas

We first introduce some lemmas which are useful in the proof of Theorem 3.1.

LEMMA **B.5.** *(Lemma A.2 of [118]) For any composite regression function in the class $\mathcal{G}(q, \mathbf{d}, \mathbf{t}, \boldsymbol{\beta}, \mathbf{K})$ and large M, such that $M \geq \max_{\ell=0,\ldots,q} \{(\beta_\ell + 1)^{t_\ell} \vee (K_\ell + 1) e^{t_\ell}\}$, there exists an estimator $f^* \in \mathcal{F}$ such that*

$$\inf_{f^* \in \mathcal{F}} \|f^* - f_0\|_\infty^2 \leq C_I M^{-\theta}, \tag{B1}$$

where $\theta = \min_{\ell=0,\ldots,q} \frac{2\beta_\ell^*}{t_\ell}$ and C_I is a constant. The parameters of the neural network f^* satisfies:

(i) $L \leq D_0 D_1 \log M - 1$;

(ii) $V^2 := \{\prod_{l=0}^{L+1} (p_l + 1)\}^2 \leq 4(d+1)^2 (12rM)^{2D_0 D_1 \log M}$;

(iii) $s \leq D_0 D_2 M \log M - 1$,

where $D_0 = 2 \max_\ell \frac{t_\ell + \beta_\ell}{\beta_\ell}$, $D_1 = 4 \max_\ell (1 + \lceil \log(t_\ell \vee \beta_\ell) \rceil)(q+1)$, $D_2 = 290 \times \sum_{\ell=0}^q d_{\ell+1} (t_\ell + \beta_\ell + 1)^{3+t_\ell}$ and $r = \max_\ell d_{\ell+1} (t_\ell + \lceil \beta_\ell \rceil)$.

LEMMA **B.6.** *Consider Euclidean space $\mathbb{R}^N$ endowed with inner product $\langle \, , \rangle_N$ and associated norm $\| \cdot \|_N$. Let $L(\cdot) : \mathbb{R}^N \to \mathbb{R}^+$ be a convex lower semicontinuous function. If $L(\mathbf{0}) <$*

$\inf_{\|\boldsymbol{a}\|_N=1} L(\boldsymbol{a})$ *for* $\boldsymbol{a} \in \mathbb{R}^N$ *, then there exists an* $\widetilde{\boldsymbol{a}}$ *(may not be unique), such that* $\widetilde{\boldsymbol{a}} = \inf_{\boldsymbol{a}\in\mathbb{R}^N} L(\boldsymbol{a})$*, and the minimizer is attained in the unit ball* $\{\boldsymbol{a} : \|\boldsymbol{a}\|_N \leq 1\}$.

Proof. Note that the given space is a Hilbert space. Let $s = \inf_{\|\boldsymbol{a}\|_N=1} L(\boldsymbol{a})$. Since L is bounded below by 0, we have $0 \leq s < \infty$. Denote $A = \{\boldsymbol{a} : L(\boldsymbol{a}) \leq s\}$ and $B = \{\boldsymbol{a} : \|\boldsymbol{a}\|_N \leq 1\}$.

We first prove $A \subset B$. Suppose $\exists\ \bar{\boldsymbol{a}} \in A$, and $\bar{\boldsymbol{a}} \notin B$, which implies $L(\bar{\boldsymbol{a}}) \leq s$ and $\|\bar{\boldsymbol{a}}\|_N > 1$. By convexity assumption,

$$L\left(\frac{\bar{\boldsymbol{a}}}{\|\bar{\boldsymbol{a}}\|_N}\right) = L\left(\frac{1}{\|\bar{\boldsymbol{a}}\|_N}\bar{\boldsymbol{a}} + \frac{\|\bar{\boldsymbol{a}}\|_N - 1}{\|\bar{\boldsymbol{a}}\|_N}\mathbf{0}\right) \leq \frac{1}{\|\bar{\boldsymbol{a}}\|_N} L\left(\bar{\boldsymbol{a}}\right) + \frac{\|\bar{\boldsymbol{a}}\|_N - 1}{\|\bar{\boldsymbol{a}}\|_N} L(\mathbf{0}) < s,$$

where the last inequality is obtained by assumption $L(\mathbf{0}) < s$. Since $\frac{\bar{\boldsymbol{a}}}{\|\bar{\boldsymbol{a}}\|_N}$ is on the unit ball, this leads to the contradiction, which implies $A \subset B$.

We then prove A is a compact set. In particular, we only need to prove the boundedness and closedness in the Euclidean space. Indeed, boundedness is implied by that fact that $A \subset B$. For any sequence $\{\boldsymbol{a}_m\}_{m\geq 1}$, define $\boldsymbol{a}^* = \lim_{m\to\infty} \boldsymbol{a}_m$, by the lower semicontinuity of L, $L(\boldsymbol{a}^*) \leq \liminf_{m\to\infty} L(\boldsymbol{a}_m) \leq s$, which implies $\boldsymbol{a}^* \in A$. Thus, the closedness is proven.

Finally, we prove that there exists a vector $\widetilde{\boldsymbol{a}} \in A$, such that $\widetilde{\boldsymbol{a}} = \inf_{\boldsymbol{a}\in\mathbb{R}^N} L(\boldsymbol{a})$, which implies that the minimizer is attained in B. Since $0 \leq \inf_{\boldsymbol{a}\in\mathbb{R}^N} L(\boldsymbol{a}) \leq s < \infty$, there exists $\{\widetilde{\boldsymbol{a}}_m\}_{m\geq 1}$, such that $\lim_{m\to\infty} L(\widetilde{\boldsymbol{a}}_m) = \inf_{\boldsymbol{a}\in\mathbb{R}^N} L(\boldsymbol{a})$. Without loss of generality, assume $L(\widetilde{\boldsymbol{a}}_m) \leq s$ for all m, i.e. $\widetilde{\boldsymbol{a}}_m \in A$. Since A is compact, there exists a vector $\widetilde{\boldsymbol{a}} \in A$, such that $\|\boldsymbol{a}_m - \widetilde{\boldsymbol{a}}\|_N \to 0$. Thus, by the lower semicontinuity of L,

$$L(\widetilde{\boldsymbol{a}}) \leq \liminf_{m\to\infty} L(\widetilde{\boldsymbol{a}}_m) = \inf_{\boldsymbol{a}\in\mathbb{R}^N} L(\boldsymbol{a}),$$

i.e. $\widetilde{\boldsymbol{a}} \in A \subset B$ is the minimzer of L. □

B.2 Proof of Theorem 3.1

Proof. Let $L_n(f)$ denote the objective function, that is,

$$L_n(f) = \frac{1}{nN} \sum_{i=1}^{n} \sum_{j=1}^{N} \rho(Y_{ij} - f(\mathbf{X}_j)).$$

Let $g(\mathbf{X}_j) = f^*(\mathbf{X}_j) - f(\mathbf{X}_j)$ and it is easy to see that minimizing $L_n(f)$ is equivalent to minimizing

$$L_n(g) = \frac{1}{nN}\sum_{i=1}^{n}\sum_{j=1}^{N}\rho(\epsilon_{ij} + R_j + g(\mathbf{X}_j)).$$

where $R_j = f_0(\mathbf{X}_j) - f^*(\mathbf{X}_j)$.

Denoting $C_{nN} = (nN^\nu)^{-\frac{\theta}{\theta+1}}\log^6(nN^\nu)$, we aim to show that for every $\epsilon > 0$, there exists a $\gamma_\epsilon \geq 1$ (in the following we simply write γ instead of γ_ϵ), such that

$$\lim_{n\to\infty} Pr\left(\inf_{\|g\|_N=\gamma} L_n(C_{nN}^{1/2}g) > L_n(0)\right) \geq 1-\epsilon. \tag{B2}$$

We first show the convexity and lower semicontinuity of L_n on $\mathcal{F}^* = \{f - f^* : f \in \mathcal{F}\}$. The convexity follows from Assumption (A3) and the convexity of the map $f \to \|f\|_N^2$. The lower semi-continuity can be derived from the Assumption (A3) that ρ is convex and continuous, thus also lower semicontinuous.

We now establish (B2). To this end, we decompose

$$\begin{aligned}
& L_n(C_{nN}^{1/2}g) - L_n(0) \\
= \;& \frac{1}{nN}\sum_{i=1}^{n}\sum_{j=1}^{N}\rho(\epsilon_{ij} + R_j + C_{nN}^{1/2}g(\mathbf{X}_j)) - \frac{1}{nN}\sum_{i=1}^{n}\sum_{j=1}^{N}\rho(\epsilon_{ij} + R_j) \\
= \;& I_1(g) + I_2(g) + I_3(g),
\end{aligned}$$

where

$$\begin{aligned}
I_1(g) &= \frac{1}{nN}\sum_{i=1}^{n}\sum_{j=1}^{N}\int_{R_j}^{R_j + C_{nN}^{1/2}g(\mathbf{X}_j)} \mathbb{E}\{\psi(\epsilon_{ij}+u)\}du \\
I_2(g) &= \frac{1}{nN}\sum_{i=1}^{n}\sum_{j=1}^{N}\int_{R_j}^{R_j + C_{nN}^{1/2}g(\mathbf{X}_j)} [\{\psi(\epsilon_{ij}+u) - \psi(\epsilon_{ij})\} \\
&\quad -\mathbb{E}\{\psi(\epsilon_{ij}+u) - \psi(\epsilon_{ij})\}]\, du \\
I_3(g) &= C_{nN}^{1/2}\frac{1}{nN}\sum_{i=1}^{n}\sum_{j=1}^{N}\psi(\epsilon_{ij})g(\mathbf{X}_j).
\end{aligned}$$

By the superadditivity of the infimum we have the lower bound

$$\inf_{\|g\|_N=\gamma} \left\{ L_n(C_{nN}^{1/2} g) - L_n(0) \right\} \geq \inf_{\|g\|_N=\gamma} I_1(g) + \inf_{\|g\|_N=\gamma} I_2(g) + \inf_{\|g\|_N=\gamma} I_3(g).$$

We need to determine the order of each one of the three terms.

By Schwarz Inequality, we obtain

$$|I_3(g)| \leq \frac{C_{nN}^{1/2}}{n} \left\{ \frac{1}{N} \sum_{j=1}^{N} \left| \sum_{i=1}^{n} \psi(\epsilon_{ij}) \right|^2 \right\}^{1/2} \left\{ \frac{1}{N} \sum_{j=1}^{N} |g(\mathbf{X}_j)|^2 \right\}^{1/2}.$$

By Assumption (A6), we have

$$\begin{aligned} \mathbb{E} \left\{ \frac{1}{N} \sum_{j=1}^{N} \left| \sum_{i=1}^{n} \psi(\epsilon_{ij}) \right|^2 \right\} &= \frac{1}{N} \sum_{j=1}^{N} \sum_{i=1}^{n} \sum_{k=1}^{n} \mathbb{E}\{\psi(\epsilon_{ij})\psi(\epsilon_{kj})\} \\ &= \frac{n}{N} \sum_{j=1}^{N} \mathbb{E}\{|\psi(\epsilon_{ij})|^2\} = O(1) n N^{-\nu}. \end{aligned}$$

Using Markov's inequality, we thus find $\left\{ \frac{1}{N} \sum_{j=1}^{N} \left| \sum_{i=1}^{n} \psi(\epsilon_{ij}) \right|^2 \right\}^{1/2} = O_p(n^{1/2} N^{-\nu/2})$. The second factor is naturally bounded by γ. Combining these two bounds and note that $(nN^{\nu})^{-1} \ll C_{nN}$, we obtain

$$\inf_{\|g\|_N=\gamma} |I_3(g)| = C_{nN}^{1/2} O_p(1) \left(nN^{\nu}\right)^{-1/2} \gamma \leq O_p(1) \gamma C_{nN}. \tag{B3}$$

When n is large enough, for all $g \in \mathcal{F}^*$,

$$C_{nN}^{1/2} \max_{j \leq N} |g(\mathbf{X}_j)| \leq 2 C_{nN}^{1/2} F = o(1),$$

and by Lemma B.5,

$$\max_{j \leq N} |R_j| \leq \|f_0 - f^*\|_{\infty} \leq C_I M^{-\theta}.$$

When $M^{-\theta} = O(1) C_{nN}$, we have $\max_{j \leq N} |R_j| = o(1)$ when n is large.

Applying Assumption (A6) yields, for large n

$$\begin{aligned}
&\int_{R_j}^{R_j+C_{nN}^{1/2}g(\mathbf{X}_j)} \mathbb{E}\{\psi(\epsilon_{ij}+u)\}du \\
= &\int_{R_j}^{R_j+C_{nN}^{1/2}g(\mathbf{X}_j)} \{\delta_j u + o(u)\}du \\
= &\frac{C_{nN}}{2}\delta_j |g(\mathbf{X}_j)|^2\{1+o(1)\} + \delta_j C_{nN}^{1/2} R_j g(\mathbf{X}_j)\{1+o(1)\}.
\end{aligned}$$

Hence,

$$\begin{aligned}
&\frac{1}{N}\sum_{j=1}^{N}\int_{R_j}^{R_j+C_{nN}^{1/2}g(\mathbf{X}_j)} \mathbb{E}\{\psi(\epsilon_{ij}+u)\}du \\
= &\frac{C_{nN}}{2N}\sum_{j=1}^{N}\delta_j |g(\mathbf{X}_j)|^2\{1+o(1)\} + \frac{C_{nN}^{1/2}}{N}\sum_{j=1}^{N}\delta_j R_j g(\mathbf{X}_j)\{1+o(1)\} \\
\geq &I_{11}(g) + I_{12}(g),
\end{aligned}$$

where $I_{11}(g) := 1/2 \inf_j \delta_j C_{nN}\|g\|_N^2$ and $I_{12}(g) := N^{-1}C_{nN}^{1/2}\sum_{j=1}^{N}\delta_j R_j |g(\mathbf{X}_j)|^2\{1+o(1)\}$. We find an upper bound for $|I_{12}(g)|$. By the Schwarz inequality,

$$\begin{aligned}
\sup_{\|g\|_N=\gamma}|I_{12}(g)| &\leq \sup_{\|g\|_N=\gamma}\left(\sup_j \delta_j\right)\frac{C_{nN}^{1/2}}{N}\sum_{j=1}^{N}|g(\mathbf{X}_j)||R_j|\{1+o(1)\} \\
&\lesssim C_{nN}^{1/2}\|R\|_N \sup_{\|g\|_N=\gamma}\|g\|_N\{1+o(1)\} \\
&\lesssim \gamma C_{nN}\{1+o(1)\},
\end{aligned}$$

where the last inequality is derived by $\|R\|_N^2 \leq \|f_0 - f^*\|_\infty \leq M^{-\theta} \asymp C_{nN}$. Therefore,

$$\inf_{\|g\|_N=\gamma} I_{11}(g) + I_{12}(g) = O(1)C_{nN}\gamma^2\{1+O(\gamma^{-1})+o(1)\}$$

for all $g \in \mathcal{F}$.

In the following, we show that for every $\epsilon > 0$,

$$\lim_{n\to\infty} Pr\left(\sup_{\|g\|_N=\gamma}|I_2(g)| \geq \epsilon C_{nN}\right) = 0. \tag{B4}$$

Consider the class of real-valued functions $h_g : \mathbb{R}^N \to \mathbb{R}$ given by

$$h_g(\mathbf{w}) = \frac{1}{N}\sum_{j=1}^{N}\int_{R_j}^{R_j+C_{nN}^{1/2}g(\mathbf{X}_j)}\{\psi(w_j+u)-\psi(w_j)\}du.$$

Let $\mathbb{P}_n$ be the empirical measure associated with $(\epsilon_{i1},\ldots,\epsilon_{iN})$, $i=1,\ldots,n$. We rewrite $I_2(g)$ as $I_2(g) = (\mathbb{P}_n - \mathbb{P})h_g$, where $\mathbb{P}h_g$ is the expectation for h_g. Let $\mathcal{F}^* = \{f - f^* : f \in \mathcal{F}\}$, where $\mathcal{F}$ is the sparse neural network class defined in (2.4). By Markov's inequality, we have

$$Pr\left(\sup_{g\in\mathcal{F}^*}|(\mathbb{P}_n-\mathbb{P})h_g| > \epsilon C_{nN}\right) \le \frac{\mathbb{E}\{|\sup_{g\in\mathcal{F}^*} n^{1/2}|(\mathbb{P}_n-\mathbb{P})h_g|\}}{n^{1/2}C_{nN}\epsilon}. \tag{B5}$$

Thus it suffices to show that the right-hand side of the above inequality tends to zero as $n \to \infty$.

Let $\mathcal{N}_{[]}(\epsilon, \mathcal{G}, L_2(\mathbb{P}))$ be the ϵ-bracketing number for a set of functions $\mathcal{G}$ in the $L_2(\mathbb{P})$-norm and define the ϵ-bracketing integral,

$$J_{[]}(\epsilon, \mathcal{G}, L_2(\mathbb{P})) = \int_0^{\epsilon}\{\log \mathcal{N}_{[]}(u, \mathcal{G}, L_2(\mathbb{P}))\}^{1/2}du.$$

By Lemma 19.36 of [109], for any class of real-valued functions $\mathcal{H}$ such that $\mathbb{E}(|h|^2) < \delta^2$ and $\|h\|_\infty < C_3$ for all $h \in \mathcal{H}$, we have

$$\mathbb{E}\left(\sup_{h\in\mathcal{H}} n^{1/2}|(\mathbb{P}_n-\mathbb{P})h|\right) \le c_0 J_{[]}(\delta, \mathcal{H}, L_2(\mathbb{P}))\left(1 + \frac{J_{[]}(\delta, \mathcal{H}, L_2(\mathbb{P}))C_3}{\delta^2 n^{1/2}}\right).$$

Hence, in the following, we first determine a bound on the bracketing number and then we estimate δ and C_3. By our previous discussion, $\max_{j\le N}|R_j| \to 0$ and $C_{nN}^{1/2} \times \sup_{g\in\mathcal{F}^*}\|g\|_\infty \to 0$. Hence, for any $(g_1, g_2) \in \mathcal{F}^* \times \mathcal{F}^*$, by Assumption (A4),

$$\begin{aligned}
&\sup_{\mathbf{w}\in\mathbb{R}^N}|h_{g_1}(\mathbf{w}) - h_{g_2}(\mathbf{w})| && \text{(B6)}\\
=\ &\sup_{\mathbf{w}\in\mathbb{R}^N}\left|\frac{1}{N}\sum_{j=1}^{N}\int_{R_j+C_{nN}^{1/2}g_2(\mathbf{X}_j)}^{R_j+C_{nN}^{1/2}g_1(\mathbf{X}_j)}\{\psi(w_j+u)-\psi(w_j)\}du\right| &&\\
\le\ &2c_1 C_{nN}^{1/2}\|g_1 - g_2\|_\infty. && \text{(B7)}
\end{aligned}$$

Let $\mathcal{N}(\epsilon, \mathcal{F}^*, \|\cdot\|_\infty)$ denote the ϵ-covering number in the sup-norm. By (B6) and Theorem 2.7.11 of [110], we now have

$$\begin{aligned} & \mathcal{N}_{[]}(\epsilon, \{h_g, g \in \mathcal{F}^*\}, L_2(\mathbb{P})) && \text{(B8)} \\ \leq \; & \mathcal{N}(\epsilon/(4c_1 C_{nN}^{1/2}), \mathcal{F}^*, \|\cdot\|_\infty). && \text{(B9)} \end{aligned}$$

By Lemma 5 of [92] and Lemma B.5, we have

$$\begin{aligned} & \log \mathcal{N}(\epsilon/(4c_1 C_{nN}^{1/2}), \mathcal{F}^*, \|\cdot\|_\infty) \\ \leq \; & (s+1)\log(2\epsilon^{-1}(4c_1 C_{nN}^{1/2})(L+1)V^2) \\ \asymp \; & O(1) M \log M \left\{ \log^2 M + \log(C_{nN}^{1/2}\epsilon^{-1}) \right\}, && \text{(B10)} \end{aligned}$$

where the last inequality follows the conditions for L, s and V given in Lemma B.5.

Next, we estimate δ and C_3. By the Schwarz inequality and Assumption (A5), we have

$$\begin{aligned} \mathbb{E}\{|h_g|^2\} &\leq N^{-2}\mathbb{E}\left\{ \left| \sum_{j=1}^N \int_{R_j}^{R_j + C_{nN}^{1/2} g(\mathbf{X}_j)} \{\psi(\epsilon_{ij}+u) - \psi(\epsilon_{ij})\} du \right|^2 \right\} \\ &\leq N^{-1} \sum_{j=1}^N \mathbb{E}\left\{ \left| \int_{R_j}^{R_j + C_{nN}^{1/2} g(\mathbf{X}_j)} \{\psi(\epsilon_{ij}+u) - \psi(\epsilon_{ij})\} du \right|^2 \right\} \\ &\leq N^{-1} \sum_{j=1}^N C_{nN}^{1/2} |g(\mathbf{X}_j)| \mathbb{E}\left\{ \int_{R_j}^{R_j + C_{nN}^{1/2} g(\mathbf{X}_j)} \{\psi(\epsilon_{ij}+u) - \psi(\epsilon_{ij})\}^2 du \right\} \\ &\leq C_{nN}^{1/2} \max_{j \leq N} |g(\mathbf{X}_j)| N^{-1} \sum_{j=1}^N c_2 \{|R_j|^2 + C_{nN}|g(\mathbf{X}_j)|^2\} \\ &\leq 2c_2 C_{nN}^{3/2} \end{aligned}$$

where the last inequality follows by the definition of neural network space $\mathcal{F}$ in (2.4). Since this estimate is uniform on $\mathcal{F}$, we take $\delta^2 = c_0 C_{nN}^{3/2}$, where the constant c_0 will be specified later. In addition, by Assumption (A5) and Lemma B.5, we have

$$\sup_{g \in \mathcal{F}^*} \sup_{\mathbf{w} \in \mathbb{R}^N} |h_g(\mathbf{w})| \leq c_1 C_{nN}^{1/2} \sup_{g \in \mathcal{F}^*} \left\{ \frac{1}{N} \sum_{j=1}^N |g(\mathbf{X}_j)|^2 \right\}^{1/2} \leq c_1 C_{nN}^{1/2}.$$

Let $c_0 = (2c_2) \vee c_1$, then $C_3 = c_0 C_{nN}^{1/2}$ and $C_3/\delta^2 = C_{nN}^{-1}$. With our estimate of δ and the bound on the bracketing number implied by (B8)–(B10), when we take $M^{-\theta} \asymp C_{nN}$, the bracketing integral over $n^{1/2}C_{nN}$ is bounded as follows

$$\begin{aligned}
& (n^{1/2}C_{nN})^{-1} J(\delta, \mathcal{F}^*, L_2(\mathbb{P})) \\
\lesssim\ & C_{nN}^{-1/4} n^{-1/2} M^{1/2} (\log^{3/2} M + (\log M \log C_{nN}^{-1/4})^{1/2}) \\
=\ & O(1) \left(n^{\theta/(4\theta+4)} \log^{-3/2} n \right) n^{-1/2} \left(n^{1/(2\theta+2)} \log^{-3/\theta} n \right) \log^2 n \\
=\ & O(1) n^{-\theta/(4\theta+4)} \log^{(\theta-6)/(2\theta)} n = o(1),
\end{aligned}$$

which also implies $C_3(\delta^2 n^{1/2})^{-1} J(\delta, \mathcal{G}, L_2(\mathbb{P})) = o(1)$. Consequently, inequality (B5) entails (B4). By the previous discussion, (B2) holds. Given the convexity and lower semicontinuity of L_n on $\mathcal{F}$, by Lemma B.6, inequality (B2) entails the existence of a minimize $\widehat{g} = f^* - \widehat{f}$, such that $\|\widehat{g}\|_N^2 = O_p(C_{nN})$. Using the one-to-one relation between g and f if (B2) holds, we obtain $\|\widehat{f} - f^*\|_N^2 = O_p(C_{nN})$. By Lemma B.5, we have

$$\begin{aligned}
\|\widehat{f} - f\|_N^2 \leq 2\|\widehat{f} - f^*\|_N^2 + 2\|f^* - f\|_N^2 &= O_p(C_{nN} + M^{-\theta}) \\
&= O_p(C_{nN} + M^{-\theta}) = O_p(C_{nN}).
\end{aligned}$$

which implies the Theorem 3.1. □

Appendix C

Optimal Classification for Functional Data

We introduce additional notations and definitions that will be used throughout the rest of the paper. $\mathbf{1}_p$ is a p-dimensional vector with elements being 1. For a vector $\mathbf{u}$, $\|\mathbf{u}\|_2$, $\|\mathbf{u}\|_\infty$ denote the L_2 norm and L_∞ norm respectively. For a matrix $\mathbf{A} \in \mathbb{R}_{p\times p}$, $\|\mathbf{A}\|_2$, $\|\mathbf{A}\|_F$ denote the spectral norm, Frobenius norm respectively.

We begin by collecting a few important technical lemmas that will be used in the proofs of the minimax lower bounds.

C.1 Technical lemmas

We define an alternative risk function $L_{\boldsymbol{\theta}}(\widehat{G})$ as follows,

$$L_{\boldsymbol{\theta}}(\widehat{G}) = P\left(\widehat{G}(Z) \neq G^*_{\boldsymbol{\theta}}(Z)\right).$$

This loss function $L_{\boldsymbol{\theta}}(\widehat{G})$ is essentially the probability that $\widehat{G}$ produces a different label than $G^*_{\boldsymbol{\theta}}$, and satisfies the triangle inequality. The connection between $R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})$ and $L_{\boldsymbol{\theta}}(\widehat{G})$ is presented by the following lemma, which shows that it's sufficient to provide a lower bound for $L_{\boldsymbol{\theta}}(\widehat{G})$ to prove Theorem 4.1.

LEMMA **C.7.** *([8]) For any $\boldsymbol{\theta}$, $\tilde{\boldsymbol{\theta}} \in \Theta$ and any classification rule $\widehat{G}$, recall that $G^*_{\tilde{\boldsymbol{\theta}}}$ is the optimal rule w.r.t. $\tilde{\boldsymbol{\theta}}$. If*

$$L_{\boldsymbol{\theta}}(G^*_{\tilde{\boldsymbol{\theta}}}) + L_{\boldsymbol{\theta}}(\widehat{G}) + \sqrt{\frac{KL(P_{\boldsymbol{\theta}}, P_{\tilde{\boldsymbol{\theta}}})}{2}} \leq 1/2,$$

then

$$L_{\boldsymbol{\theta}}(G^*_{\tilde{\boldsymbol{\theta}}}) - L_{\boldsymbol{\theta}}(\widehat{G}) - \sqrt{\frac{KL(P_{\boldsymbol{\theta}}, P_{\tilde{\boldsymbol{\theta}}})}{2}} \leq L_{\tilde{\boldsymbol{\theta}}}(\widehat{G}) \leq L_{\boldsymbol{\theta}}(G^*_{\tilde{\boldsymbol{\theta}}}) + L_{\boldsymbol{\theta}}(\widehat{G}) + \sqrt{\frac{KL(P_{\boldsymbol{\theta}}, P_{\tilde{\boldsymbol{\theta}}})}{2}},$$

where the Kullback–Leibler (KL) divergence of two probability density functions $P_{\boldsymbol{\theta}_1}$ and $P_{\tilde{\boldsymbol{\theta}}_2}$ is defined by

$$KL(P_{\boldsymbol{\theta}_1}, P_{\tilde{\boldsymbol{\theta}}_2}) = \int P_{\boldsymbol{\theta}_1}(z) \log \frac{P_{\boldsymbol{\theta}_1}(z)}{P_{\tilde{\boldsymbol{\theta}}_2}(z)} dz.$$

The following lemma gives a Fano's type minimax lower bound.

LEMMA **C.8.** *(Fano's Lemma in [108]) Let $N \geq 0$ and $\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N$, $\tilde{\boldsymbol{\theta}} \in \Theta$. For some constants $\varrho \in (0, 1/8)$, $c > 0$, and any classification rule $\widehat{G}$, if $KL(P_{\boldsymbol{\theta}_i}, P_{\tilde{\boldsymbol{\theta}}_0}) \geq \varrho \log N/n$ for all $1 \leq i \leq N$, and $L_{\boldsymbol{\theta}_i}(\widehat{G}) < c$ implies $L_{\boldsymbol{\theta}_i}(\widehat{G}) \geq c$ for all $0 \leq i \neq j \leq N$, then* $\inf \sup_{i=1,\ldots,N} E_{\boldsymbol{\theta}_i}[L_{\boldsymbol{\theta}_i}(\widehat{G})] \gtrsim c$.

We need a covering number argument, which is provided by the following lemma.

LEMMA **C.9.** *([108]) Define $\mathcal{A}_{J,J^*} = \left\{\boldsymbol{u} : \boldsymbol{u} \in \{0,1\}^J, \|\boldsymbol{u}\|_0 = J^*\right\}$, where $\|\cdot\|_0$ denotes the number of non-zero entries. If $J > 4J^*$, then there exists a subset $\{\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_N\} \subset \mathcal{A}_{J,J^*}$, such that $\boldsymbol{u}_0 = (0, \ldots, 0)^\top$, $\rho_H(\boldsymbol{u}_i, \boldsymbol{u}_j) \geq J^*/2$ and $\log(N+1) \geq \frac{J^*}{5} \log(\frac{J}{J^*})$, where ρ_H the Hamming distance.*

LEMMA **C.10.** *(Lemma 4.1 in [18]) Suppose $\boldsymbol{\theta} \in \Theta$. There exists a constant $c > 0$, which doesn't depend on n, such that for any classification rule G, if $L_{\boldsymbol{\theta}}(G) < c$, then $L^2_{\boldsymbol{\theta}}(G) \lesssim P_{\boldsymbol{\theta}}(G(Z) \neq L(Z)) - P_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}(Z) \neq L(Z))$.*

Based on Lemma C.10, we use Fano's inequality on a carefully designed least favorable multivariate normal distributions to complete the proof of Theorem 4.1. Without loss of generality, the following proofs assume that $\boldsymbol{z}$ is from class 1.

The following two lemmas show the consistency of the differential direction and graphical direction. With a slight abuse of notation, $\mathbf{D}$ represents the matrix given in (4.1).

LEMMA **C.11.** *The proposed estimators in (4.3) and (4.4) satisfy that, with probability at least $1 - 3/n$, $\|\widehat{\mathbf{D}} - \mathbf{D}\|_F \lesssim \sqrt{\frac{J \log n}{n}}(\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F)$, $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|_2 \lesssim \sqrt{\frac{J \log n}{n}}\|\boldsymbol{\beta}\|_2$.*

Proof. Note that matrices $\mathbf{\Sigma}_1$, $\mathbf{\Sigma}_2$, $\mathbf{D}_1$ and $\mathbf{D}_2$ are diagonal matrices. Hence, $\|(\widehat{\mathbf{D}}-\mathbf{D})\,(\mathbf{\Sigma}_1\mathbf{\Sigma}_2)\,(\widehat{\mathbf{D}}-\mathbf{D})\|_F \asymp \|\widehat{\mathbf{D}}-\mathbf{D}\|_F^2$. Since we have the following decomposition

$$
\begin{aligned}
& \mathbf{\Sigma}_1\mathbf{\Sigma}_2(\widehat{\mathbf{D}}-\mathbf{D}) \\
= \;& \mathbf{\Sigma}_1\mathbf{\Sigma}_2\widehat{\mathbf{D}} - (\mathbf{\Sigma}_1-\mathbf{\Sigma}_2) \\
= \;& \mathbf{\Sigma}_1\mathbf{\Sigma}_2\widehat{\mathbf{D}} - \widehat{\mathbf{\Sigma}}_1\widehat{\mathbf{\Sigma}}_2\widehat{\mathbf{D}} + \widehat{\mathbf{\Sigma}}_1\widehat{\mathbf{\Sigma}}_2\widehat{\mathbf{D}} - (\mathbf{\Sigma}_1-\mathbf{\Sigma}_2) \\
= \;& (\mathbf{\Sigma}_1\mathbf{\Sigma}_2 - \widehat{\mathbf{\Sigma}}_1\widehat{\mathbf{\Sigma}}_2)\widehat{\mathbf{D}} + (\widehat{\mathbf{\Sigma}}_1-\mathbf{\Sigma}_1) + (\widehat{\mathbf{\Sigma}}_2-\mathbf{\Sigma}_2) \\
= \;& (\mathbf{\Sigma}_1\mathbf{\Sigma}_2 - \mathbf{\Sigma}_1\widehat{\mathbf{\Sigma}}_2 + \mathbf{\Sigma}_1\widehat{\mathbf{\Sigma}}_2 - \widehat{\mathbf{\Sigma}}_1\widehat{\mathbf{\Sigma}}_2)\widehat{\mathbf{D}} + (\widehat{\mathbf{\Sigma}}_1-\mathbf{\Sigma}_1) + (\widehat{\mathbf{\Sigma}}_2-\mathbf{\Sigma}_2) \\
= \;& \left\{\mathbf{\Sigma}_1(\mathbf{\Sigma}_2-\widehat{\mathbf{\Sigma}}_2) + \widehat{\mathbf{\Sigma}}_2(\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)\right\}\widehat{\mathbf{D}} + (\widehat{\mathbf{\Sigma}}_1-\mathbf{\Sigma}_1) + (\widehat{\mathbf{\Sigma}}_2-\mathbf{\Sigma}_2) \\
= \;& \left\{\mathbf{\Sigma}_1(\mathbf{\Sigma}_2-\widehat{\mathbf{\Sigma}}_2) + \mathbf{\Sigma}_2(\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1) + (\widehat{\mathbf{\Sigma}}_2-\mathbf{\Sigma}_2)(\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)\right\}\widehat{\mathbf{D}} + (\widehat{\mathbf{\Sigma}}_1-\mathbf{\Sigma}_1) + (\widehat{\mathbf{\Sigma}}_2-\mathbf{\Sigma}_2),
\end{aligned}
$$

thus we have

$$
\begin{aligned}
& \|(\widehat{\mathbf{D}}-\mathbf{D})^\top\,(\mathbf{\Sigma}_1\mathbf{\Sigma}_2)\,(\widehat{\mathbf{D}}-\mathbf{D})\|_F \\
= \;& \Big\|(\widehat{\mathbf{D}}-\mathbf{D})^\top\Big\{(\widehat{\mathbf{\Sigma}}_1-\widehat{\mathbf{\Sigma}}_2) - (\mathbf{\Sigma}_1-\mathbf{\Sigma}_2) + (\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)(\widehat{\mathbf{D}}-\mathbf{D})\widehat{\mathbf{\Sigma}}_2 \\
& + (\mathbf{\Sigma}_2-\widehat{\mathbf{\Sigma}}_2)(\widehat{\mathbf{D}}-\mathbf{D})\mathbf{\Sigma}_1 + (\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)\mathbf{D}\widehat{\mathbf{\Sigma}}_2 + (\mathbf{\Sigma}_2-\widehat{\mathbf{\Sigma}}_2)\mathbf{D}\mathbf{\Sigma}_1\Big\}\Big\|_F \\
= \;& \Big\|(\widehat{\mathbf{D}}-\mathbf{D})^\top\Big\{(\widehat{\mathbf{\Sigma}}_1-\widehat{\mathbf{\Sigma}}_2) - (\mathbf{\Sigma}_1-\mathbf{\Sigma}_2) + (\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)(\widehat{\mathbf{D}}-\mathbf{D})(\widehat{\mathbf{\Sigma}}_2-\mathbf{\Sigma}_2) \\
& + (\mathbf{\Sigma}_2-\widehat{\mathbf{\Sigma}}_2)(\widehat{\mathbf{D}}-\mathbf{D})\mathbf{\Sigma}_1 + (\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)\mathbf{D}(\widehat{\mathbf{\Sigma}}_2-\mathbf{\Sigma}_2) + (\mathbf{\Sigma}_2-\widehat{\mathbf{\Sigma}}_2)\mathbf{D}\mathbf{\Sigma}_1 \\
& + (\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)(\widehat{\mathbf{D}}-\mathbf{D})\mathbf{\Sigma}_2 + (\mathbf{\Sigma}_1-\widehat{\mathbf{\Sigma}}_1)\mathbf{D}\mathbf{\Sigma}_2\Big\}\Big\|_F \\
\leq \;& 2\sqrt{\frac{J\log n}{n}}\|\widehat{\mathbf{D}}-\mathbf{D}\|_F + \frac{J\log n}{n}\|\widehat{\mathbf{D}}-\mathbf{D}\|_F^2 + 2\sqrt{\frac{J\log n}{n}}\|\widehat{\mathbf{D}}-\mathbf{D}\|_F \\
& + \sqrt{\frac{J\log n}{n}}\|(\mathbf{D}\mathbf{\Sigma}_1\|_F + \mathbf{D}\mathbf{\Sigma}_2\|_F)\|\widehat{\mathbf{D}}-\mathbf{D}\|_F + \frac{J\log n}{n}\|\mathbf{D}\mathbf{\Sigma}_1\|_F\|\widehat{\mathbf{D}}-\mathbf{D}\|_F \\
\lesssim \;& \sqrt{\frac{J\log n}{n}}\|\widehat{\mathbf{D}}-\mathbf{D}\|_F^2 + \sqrt{\frac{J\log n}{n}}\|\widehat{\mathbf{D}}-\mathbf{D}\|_F + \sqrt{\frac{J\log n}{n}}(\|\mathbf{D}\mathbf{\Sigma}_1\|_F + \|\mathbf{D}\mathbf{\Sigma}_2\|_F)\|\widehat{\mathbf{D}}-\mathbf{D}\|_F,
\end{aligned}
$$

where the first inequality is derived by Lemma 8.5 in [18]. In the above inequality, we divide $\|\widehat{\mathbf{D}}-\mathbf{D}\|_F$ on both sides, and we have $\|\widehat{\mathbf{D}}-\mathbf{D}\|_F \lesssim \sqrt{\frac{J\log n}{n}}(\|\mathbf{D}\mathbf{\Sigma}_1\|_F + \|\mathbf{D}\mathbf{\Sigma}_2\|_F)$.

Similarly, we have $|(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top \boldsymbol{\Sigma}_2(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})| \asymp \|\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}\|_2^2$. With probability at least $1-O(1/n)$, we have

$$\begin{aligned}
& |(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top \boldsymbol{\Sigma}_2(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})| \\
= \; & |(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top (\boldsymbol{\Sigma}_2-\widehat{\boldsymbol{\Sigma}}_2)\widehat{\boldsymbol{\beta}} + (\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top (\widehat{\boldsymbol{\delta}}-\boldsymbol{\delta})| \\
= \; & |(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top (\boldsymbol{\Sigma}_2-\widehat{\boldsymbol{\Sigma}}_2)\boldsymbol{\beta} + (\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top (\boldsymbol{\Sigma}_2-\widehat{\boldsymbol{\Sigma}}_2)(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}) + (\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})^\top (\widehat{\boldsymbol{\delta}}-\boldsymbol{\delta})| \\
\lesssim \; & \sqrt{\frac{J\log n}{n}}\|\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}\|_2\|\boldsymbol{\beta}\|_2 + \sqrt{\frac{J\log n}{n}}\|\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}\|_2^2 + \sqrt{\frac{\log n}{n}}\|\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}\|_2.
\end{aligned}$$

In the above inequality, we divide $\|\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}\|_2$ on both sides, and we have $\|\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}\|_2 \lesssim \sqrt{\frac{J\log n}{n}}\|\boldsymbol{\beta}\|_2$.

□

LEMMA **C.12.** *The proposed estimators in (4.6) and (4.7) satisfy that, with probability at least* $1-O(1/n)$,

$$\begin{aligned}
\|\widehat{\mathbf{D}}_s-\mathbf{D}\|_F &\lesssim \left(\sqrt{\frac{J\log n}{n}}+\sqrt{J}f_2(M)\right)\left(\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F+\|\mathbf{D}\boldsymbol{\Sigma}_2\|_F\right), \\
\|\widehat{\boldsymbol{\beta}}_s-\boldsymbol{\beta}\|_2 &\lesssim \left(\sqrt{\frac{J\log n}{n}}+\sqrt{J}f_2(M)\right)\|\boldsymbol{\beta}\|_2+\sqrt{J}f_1(M).
\end{aligned}$$

Proof. In the following we omit k for simplicity. Consider the parameter space Θ, we have $\widehat{\boldsymbol{\mu}}-\boldsymbol{\mu}=\bar{\boldsymbol{\xi}}-\boldsymbol{\mu}+\mathbf{B}^{-1}\bar{\boldsymbol{\epsilon}}$, where $\bar{\boldsymbol{\xi}}=\left(\bar{\xi}_{\cdot 1},\bar{\xi}_{\cdot 2},\ldots,\bar{\xi}_{\cdot J}\right)^\top$, $\bar{\boldsymbol{\epsilon}}=\frac{1}{n}\sum_{i=1}^{n_k}\boldsymbol{\epsilon}_i$, $\boldsymbol{\epsilon}_i=(\epsilon_{i1},\epsilon_{i2},\ldots,\epsilon_{iM})^\top$ and $\epsilon_{im}=\sum_{j=J+1}^{\infty}\xi_{ij}\psi_j(t_m)$, $m=1,\ldots,M$. For any M dimensional vector $\boldsymbol{a}$ and $J\le M$, define $\boldsymbol{a}_J$ as the vector of first J elements in $\boldsymbol{a}$.

Since $\bar{\boldsymbol{\epsilon}}=\left(\sum_{j=M+1}^{\infty}\bar{\xi}_j\psi_j(t_1),\ldots,\sum_{j=M+1}^{\infty}\bar{\xi}_j\psi_j(t_M)\right)^\top$, we have

$$\mathbf{B}^{-1}\bar{\boldsymbol{\epsilon}}=\left(\sum_{m=1}^{M}\psi_1(t_m)\sum_{j=M+1}^{\infty}\bar{\xi}_j\psi_j(t_m),\ldots,\sum_{m=1}^{M}\psi_M(t_m)\sum_{j=M+1}^{\infty}\bar{\xi}_j\psi_j(t_m)\right)$$

and $\bar{\xi}_j \sim \mathcal{N}(\mu_j, n^{-1}\lambda_j)$, hence for the first J elements, where $J \leq M$, we have

$$\begin{aligned} \| \left(\mathbf{B}^{-1}\bar{\boldsymbol{\epsilon}}\right)_J \|_2^2 &= \sum_{k=1}^{J} \left\{ \sum_{j=M+1}^{\infty} \left(\sum_{m=1}^{M} \psi_k(t_m)\psi_j(t_m)\bar{\xi}_j \right) \right\}^2 \\ &\lesssim \sum_{j=M+1}^{\infty} \sum_{k=1}^{J} \left\{ \sum_{m=1}^{M} \psi_k(t_m)\psi_j(t_m) \right\}^2 (\mu_j^2 + n^{-1}\log n\lambda_j) \\ &\lesssim Jf_1^2(M) + \frac{\log n}{n} \left(\sum_{j=M+1}^{\infty} \lambda_j \right) \lesssim Jf_1^2(M) + \frac{\log n}{n} \end{aligned}$$

in probability $1 - O(1/n)$, therefore

$$\|\widehat{\boldsymbol{\mu}}_J - \boldsymbol{\mu}_J\|_2 \leq \|\bar{\boldsymbol{\xi}}_J - \boldsymbol{\mu}_J\|_2 + \| \left(\mathbf{B}^{-1}\bar{\boldsymbol{\epsilon}}\right)_J \|_2 \lesssim \sqrt{\frac{\log n}{n}} + \sqrt{J} f_1(M)$$

with probability $1 - O(1/n)$. In the following lemma, for any $J \times J$ square matrix A, let $Diag\{A\}$ denote a $J \times J$ diagonal matrix whose diagonal elements are the same as the diagonal elements of A. We estimate the variance as $\widehat{\lambda}_j = \frac{1}{n}\sum_{i=1}^{n} (\widehat{\mu}_{ij} - \widehat{\mu}_j)^2$, i.e.

$$\begin{aligned} \widehat{\boldsymbol{\Sigma}} &= Diag\left\{ \frac{1}{n}\sum_{i=1}^{n} (\widehat{\boldsymbol{\mu}}_i - \widehat{\boldsymbol{\mu}})(\widehat{\boldsymbol{\mu}}_i - \widehat{\boldsymbol{\mu}})^\top \right\} \\ &= Diag\left\{ \mathbf{B}^{-1} \left[\frac{1}{n}\sum_{i=1}^{n} \left(\mathbf{X}_i\mathbf{X}_i^\top - \bar{\mathbf{X}}\bar{\mathbf{X}}^\top \right) \right] \left(\mathbf{B}^{-1}\right)^\top \right\} \\ &= Diag\left\{ \frac{1}{n}\sum_{i=1}^{n} \left(\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top - \bar{\boldsymbol{\xi}}\bar{\boldsymbol{\xi}}^\top \right) \right\} + Diag\left\{ \frac{2}{n}\sum_{i=1}^{n} \left(\boldsymbol{\xi}_i\boldsymbol{\epsilon}_i^\top - \bar{\boldsymbol{\xi}}\bar{\boldsymbol{\epsilon}}^\top \right) \left(\mathbf{B}^{-1}\right)^\top \right\} \\ &\quad + Diag\left\{ \frac{1}{n}\sum_{i=1}^{n} \mathbf{B}^{-1} \left(\boldsymbol{\epsilon}_i\boldsymbol{\epsilon}_i^\top - \bar{\boldsymbol{\epsilon}}\bar{\boldsymbol{\epsilon}}^\top \right) \left(\mathbf{B}^{-1}\right)^\top \right\}. \end{aligned}$$

The estimation bias is

$$\begin{aligned} \widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma} &= Diag\left\{ \frac{1}{n}\sum_{i=1}^{n} \left(\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top - \bar{\boldsymbol{\xi}}\bar{\boldsymbol{\xi}}^\top \right) \right\} - \boldsymbol{\Sigma} + Diag\left\{ \frac{2}{n}\sum_{i=1}^{n} \left(\boldsymbol{\xi}_i\boldsymbol{\epsilon}_i^\top - \bar{\boldsymbol{\xi}}\bar{\boldsymbol{\epsilon}}^\top \right) \left(\mathbf{B}^{-1}\right)^\top \right\} \\ &\quad + Diag\left\{ \frac{1}{n}\sum_{i=1}^{n} \mathbf{B}^{-1} \left(\boldsymbol{\epsilon}_i\boldsymbol{\epsilon}_i^\top - \bar{\boldsymbol{\epsilon}}\bar{\boldsymbol{\epsilon}}^\top \right) \left(\mathbf{B}^{-1}\right)^\top \right\}, \end{aligned}$$

Note that by Lemma 8.5 in [18],

$$\left\| Diag\left\{ \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top - \bar{\boldsymbol{\xi}}\bar{\boldsymbol{\xi}}^\top\right)\right\} - \boldsymbol{\Sigma}\right\|_2 \lesssim \sqrt{\frac{\log n}{n}},$$

and

$$\begin{aligned}
&\left\| Diag\left\{ \frac{2}{n}\sum_{i=1}^{n}\left(\boldsymbol{\xi}_i\boldsymbol{\epsilon}_i^\top - \bar{\boldsymbol{\xi}}\bar{\boldsymbol{\epsilon}}^\top\right)\left(\mathbf{B}^{-1}\right)^\top\right\}\right\|_2 \\
\lesssim\ & \sqrt{\frac{\log n}{n}}\sup_m\sqrt{\sum_{j=1}^{M}\lambda_j\psi_j(t_m)\psi_j(t_m)}\sup_m\sqrt{\sum_{j=M+1}^{\infty}\lambda_j\psi_j(t_m)\psi_j(t_m)} \\
&\times\left\| Diag\left\{\mathbf{I}_M\left(\mathbf{B}^{-1}\right)^\top\right\}\right\|_2
\end{aligned}$$

and

$$\begin{aligned}
&\left\| Diag\left\{ \frac{1}{n}\sum_{i=1}^{n}\mathbf{B}^{-1}\left(\boldsymbol{\epsilon}_i\boldsymbol{\epsilon}_i^\top - \bar{\boldsymbol{\epsilon}}\bar{\boldsymbol{\epsilon}}^\top\right)\left(\mathbf{B}^{-1}\right)^\top\right\}\right\|_2 \\
\leq\ & \left(1+\sqrt{\frac{\log n}{n}}\right)\sup_{m,m'}\sum_{j=M+1}^{\infty}\lambda_j\psi_j(t_m)\psi_j(t_{m'})\left\| Diag\left\{\mathbf{B}^{-1}\mathbf{I}_M\left(\mathbf{B}^{-1}\right)^\top\right\}\right\|_2
\end{aligned}$$

with probability at least $1-1/n$ separately. Then we have

$$\begin{aligned}
&\|\widehat{\mathbf{\Sigma}}-\mathbf{\Sigma}\|_2 \\
\lesssim\ & \sqrt{\frac{\log n}{n}}+\sqrt{\frac{\log n}{n}}\sup_m\sqrt{\sum_{j=1}^{M}\lambda_j\psi_j(t_m)\psi_j(t_m)}\sup_m\sqrt{\sum_{j=M+1}^{\infty}\lambda_j\psi_j(t_m)\psi_j(t_m)} \\
&\times\left\|Diag\left\{\mathbf{I}_M\left(\mathbf{B}^{-1}\right)^\top\right\}\right\|_2 \\
&+\left(1+\sqrt{\frac{\log n}{n}}\right)\sup_{m,m'}\sum_{j=M+1}^{\infty}\lambda_j\psi_j(t_m)\psi_j(t_{m'})\|Diag\left\{\mathbf{B}^{-1}\mathbf{I}_M\left(\mathbf{B}^{-1}\right)^\top\right\}\|_2 \\
=\ & \sqrt{\frac{\log n}{n}}\left(1+\sup_m\sqrt{\sum_{j=1}^{M}\lambda_j\psi_j(t_m)\psi_j(t_m)}\sup_m\sqrt{\sum_{j=M+1}^{\infty}\lambda_j\psi_j(t_m)\psi_j(t_m)}\right. \\
&\left.\times\sup_{1\le j\le M}\sum_{m=1}^{M}\psi_j(t_m)\right) \\
&+\left(1+\sqrt{\frac{\log n}{n}}\right)\sup_{m,m'}\sum_{j=M+1}^{\infty}\lambda_j\psi_j(t_m)\psi_j(t_{m'})\sup_{1\le j\le M}\left(\sum_{m=1}^{M}\psi_j(t_m)\right)^2,
\end{aligned}$$

with probability at least $1-3/n$.

Thus

$$\begin{aligned}
\|\widehat{\mathbf{\Sigma}}-\mathbf{\Sigma}\|_2 &\lesssim \sqrt{\frac{\log n}{n}}+\sqrt{\frac{\log n}{n}}\frac{1}{\sqrt{M}}\sqrt{\sum_{j=M+1}^{\infty}\lambda_j}+\left(1+\sqrt{\frac{\log n}{n}}\right)\sum_{j=M+1}^{\infty}\lambda_j \\
&\asymp \sqrt{\frac{\log n}{n}}+\sum_{j=M+1}^{\infty}\lambda_j\lesssim\sqrt{\frac{\log n}{n}}+f_2(M),
\end{aligned}$$

with probability at least $1-3/n$. Hence, the result can be easily derived from the proof of Lemma C.11.

□

Let $M(\boldsymbol{z})=Q(\boldsymbol{z};\boldsymbol{\theta})-\widehat{Q}(\boldsymbol{z};\boldsymbol{\theta})$ and $\Delta_{M(\boldsymbol{z})}=\sqrt{\frac{J\log n}{n}}(\|\mathbf{D}\|_F+\|\boldsymbol{\beta}\|_2)$.

LEMMA **C.13.** *We have* $P\left(M(\boldsymbol{z})\lesssim\Delta_{M(\boldsymbol{z})}\right)\ge 1-O(1/n)$.

Proof. Given the definition of $Q(\boldsymbol{z};\boldsymbol{\theta})$ in (4.2) and $\widehat{Q}(\boldsymbol{z};\boldsymbol{\theta})$, we have

$$
\begin{aligned}
& M(\boldsymbol{z}) \\
= & \left\{(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \mathbf{D}(\boldsymbol{z}-\boldsymbol{\mu}_1) - (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top \widehat{\mathbf{D}}(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)\right\} + \left\{2\widehat{\boldsymbol{\beta}}^\top(\boldsymbol{z}-\widehat{\bar{\boldsymbol{\mu}}}) - 2\boldsymbol{\beta}^\top(\boldsymbol{z}-\bar{\boldsymbol{\mu}})\right\} \\
& + \log|\widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1 + \mathbf{I}_J| - \log|\mathbf{D}\boldsymbol{\Sigma}_1 + \mathbf{I}_J|) + 2\log(\pi_1/\pi_2) - 2\log(\widehat{\pi}_1/\widehat{\pi}_2) \\
= & \left\{(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \mathbf{D}(\boldsymbol{z}-\boldsymbol{\mu}_1) - (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top \widehat{\mathbf{D}}(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)\right\} - tr\left(\boldsymbol{\Sigma}_1^{1/2}(\widehat{\mathbf{D}}-\mathbf{D})\boldsymbol{\Sigma}_1^{1/2}\right) \\
& + \left\{2\widehat{\boldsymbol{\beta}}^\top(\boldsymbol{z}-\widehat{\bar{\boldsymbol{\mu}}}) - 2\boldsymbol{\beta}^\top(\boldsymbol{z}-\bar{\boldsymbol{\mu}})\right\} \\
& + \log|\widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1 + \mathbf{I}_J| - \log|\mathbf{D}\boldsymbol{\Sigma}_1 + \mathbf{I}_J|) + tr\left(\boldsymbol{\Sigma}_1^{1/2}(\widehat{\mathbf{D}}-\mathbf{D})\boldsymbol{\Sigma}_1^{1/2}\right) \\
& + 2\log(\pi_1/\pi_2) - 2\log(\widehat{\pi}_1/\widehat{\pi}_2) \\
= & \Xi_1 + \Xi_2 + \Xi_3 + \Xi_4
\end{aligned}
$$

Without loss of generality, assuming $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$. First, we shall bound Ξ_1. Note that

$$
\begin{aligned}
& (\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \mathbf{D}(\boldsymbol{z}-\boldsymbol{\mu}_1) - (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top \widehat{\mathbf{D}}(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1) \\
= & (\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \mathbf{D}(\boldsymbol{z}-\boldsymbol{\mu}_1) - (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D}(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1) + (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top(\widehat{\mathbf{D}}-\mathbf{D})(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1) \\
= & 2(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)\mathbf{D}\boldsymbol{z} + (\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D}(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1) + (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top(\widehat{\mathbf{D}}-\mathbf{D})(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1) \\
\leq & 2\sqrt{(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D}(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)}\sqrt{\boldsymbol{z}^\top\mathbf{D}\boldsymbol{z}} + (\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D}(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1) \\
& + (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top(\widehat{\mathbf{D}}-\mathbf{D})(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1) \\
= & I_1 + I_2 + I_3
\end{aligned}
$$

According to Gaussian Chaos, since $\widehat{\boldsymbol{\mu}}_1 - \boldsymbol{\mu}_1 \sim \mathcal{N}(\mathbf{0}, \frac{1}{n}\boldsymbol{\Sigma}_1)$, $E\left\{(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D}(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)\right\} = n^{-1}\sum_{j=1}^J(\epsilon_j - 1)$, and $\epsilon_j = \lambda_j^{(1)}/\lambda_j^{(2)}$, we have

$$
\begin{aligned}
& P\left((\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D}(\boldsymbol{\mu}_1-\widehat{\boldsymbol{\mu}}_1) > 2\left\|\frac{1}{n}(\text{diag}(\epsilon_1,\ldots,\epsilon_J) - \mathbf{I}_J)\right\|_F \sqrt{x}\right. \\
& \left. + 2\left\|\frac{1}{n}(\text{diag}(\epsilon_1,\ldots,\epsilon_J) - \mathbf{I}_J)\right\|_2 x + \sum_{j=1}^J \frac{\epsilon_j - 1}{n}\right) \leq \exp(-x)
\end{aligned}
$$

Since $\|\frac{1}{n}(\text{diag}(\epsilon_1, \ldots, \epsilon_J) - \mathbf{I}_J)\|_F = \frac{1}{n}\sqrt{\sum_{j=1}^J (\epsilon_j - 1)^2}$ and $\|\frac{1}{n}(\text{diag}(\epsilon_1, \ldots, \epsilon_J) - \mathbf{I}_J)\|_2 = \frac{1}{n}\max_j(\epsilon_j - 1)$. Therefore, take $x = \log n$, we have

$$
\begin{aligned}
& I_2 \\
= \; & (\boldsymbol{\mu}_1 - \widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D} (\boldsymbol{\mu}_1 - \widehat{\boldsymbol{\mu}}_1) \lesssim \frac{\sqrt{\log n}}{n} \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \frac{\log n}{n} + \frac{1}{n} tr\left(\mathbf{D}\boldsymbol{\Sigma}_1\right) \lesssim \frac{\sqrt{J}}{n} \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F
\end{aligned} \tag{C1}
$$

with probability at least $1 - 1/n$. Note that

$$
\boldsymbol{z}^\top \mathbf{D} \boldsymbol{z} = (\boldsymbol{z} - \boldsymbol{\mu}_1)^\top \mathbf{D} (\boldsymbol{z} - \boldsymbol{\mu}_1) + 2\boldsymbol{\mu}_1^\top \mathbf{D} (\boldsymbol{z} - \boldsymbol{\mu}_1) + \boldsymbol{\mu}_1^\top \mathbf{D} \boldsymbol{\mu}_1
$$

Since $\boldsymbol{z} - \boldsymbol{\mu}_1 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_1)$, and we have $E\left\{(\boldsymbol{z} - \boldsymbol{\mu}_1)^\top \mathbf{D} (\boldsymbol{z} - \boldsymbol{\mu}_1)\right\} = \sum_{j=1}^J (\epsilon_j - 1)$, Thus

$$
(\boldsymbol{z} - \boldsymbol{\mu}_1)^\top \mathbf{D} (\boldsymbol{z} - \boldsymbol{\mu}_1) \lesssim \sqrt{\log n} \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \log n + tr\left(\mathbf{D}\boldsymbol{\Sigma}_1\right) \lesssim \sqrt{J} \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F
$$

with probability at least $1 - 1/n$. Note that $2\boldsymbol{\mu}_1^\top \mathbf{D}(\boldsymbol{z} - \boldsymbol{\mu}_1) \sim \mathcal{N}(0, 4\sum_{j=1}^J \frac{\mu_{1j}^2}{\lambda_j^{(1)}} (\epsilon_j - 1)^2)$, we have $\boldsymbol{\mu}_1^\top \mathbf{D}(\boldsymbol{z} - \boldsymbol{\mu}_1) \lesssim \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F \sqrt{\log n}$ with probability at least $1 - O(1/n)$. Since $\boldsymbol{\mu}_1^\top \mathbf{D} \boldsymbol{\mu}_1 \leq \|\boldsymbol{\Sigma}_1^{-1/2} \boldsymbol{\mu}\|_4^2 \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F$, we have $\boldsymbol{z}^\top \mathbf{D} \boldsymbol{z} \lesssim \sqrt{J} \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F$ with probability at least $1 - O(1/n)$. Thus

$$
I_1 = 2\sqrt{(\boldsymbol{\mu}_1 - \widehat{\boldsymbol{\mu}}_1)^\top \mathbf{D} (\boldsymbol{\mu}_1 - \widehat{\boldsymbol{\mu}}_1)} \sqrt{\boldsymbol{z}^\top \mathbf{D} \boldsymbol{z}} \lesssim \sqrt{\frac{J}{n}} \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F. \tag{C2}
$$

Denote the diagonal matrix $\boldsymbol{\Sigma}_1^{1/2}(\widehat{\mathbf{D}}-\mathbf{D})\boldsymbol{\Sigma}_1^{1/2} = \text{diag}(\rho_1, \ldots, \rho_J)$, and $\boldsymbol{z}_0 = (z_{01}, \ldots, z_{0J})^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_J)$ are independent standard normally distributed random variables, then we have

$$\begin{aligned}
& I_3 - tr\left(\boldsymbol{\Sigma}_1^{1/2}(\widehat{\mathbf{D}}-\mathbf{D})\boldsymbol{\Sigma}_1^{1/2}\right) \\
= \; & (\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1)^\top(\widehat{\mathbf{D}}-\mathbf{D})(\boldsymbol{z}-\widehat{\boldsymbol{\mu}}_1) - tr\left(\boldsymbol{\Sigma}_1^{1/2}(\widehat{\mathbf{D}}-\mathbf{D})\boldsymbol{\Sigma}_1^{1/2}\right) \\
= \; & (1+\frac{1}{n})\boldsymbol{z}_0^\top(\boldsymbol{\Sigma}_1^{1/2}(\widehat{\mathbf{D}}-\mathbf{D})\boldsymbol{\Sigma}_1^{1/2})\boldsymbol{z}_0 - tr\left(\boldsymbol{\Sigma}_1^{1/2}(\widehat{\mathbf{D}}-\mathbf{D})\boldsymbol{\Sigma}_1^{1/2}\right) \\
= \; & (1+\frac{1}{n})\sum_{j=1}^{J}\rho_j(z_{0j}^2-1) + \frac{1}{n}\sum_{j=1}^{J}\rho_j \\
\lesssim \; & (1+\frac{1}{n})\sqrt{\frac{J\log n}{n}} + \frac{1}{n}\|\widehat{\mathbf{D}}-\mathbf{D}\|_F\|\boldsymbol{\Sigma}_1\|_2 \\
\lesssim \; & \sqrt{\frac{J\log n}{n}}(\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F)
\end{aligned}$$

with probability at least $1-O(1/n)$, where the second last inequality comes from Gaussian Chaos and the last inequality comes from Lemma C.11. Combining (C1), (C2) and (C3),

$$\Xi_1 \lesssim \sqrt{\frac{J\log n}{n}}(\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F). \tag{C3}$$

Secondly, by Lemma C.11, with probability at least $1-O(1/n)$, we have

$$\begin{aligned}
\Xi_2 = |2\widehat{\boldsymbol{\beta}}(\boldsymbol{z}-\widehat{\bar{\boldsymbol{\mu}}}) - 2\boldsymbol{\beta}(\boldsymbol{z}-\bar{\boldsymbol{\mu}})| &= |2(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})(\boldsymbol{z}-\widehat{\bar{\boldsymbol{\mu}}}) - \boldsymbol{\beta}(\widehat{\bar{\boldsymbol{\mu}}}-\bar{\boldsymbol{\mu}})| \\
&\leq 2\|\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta}\|_2\|\boldsymbol{z}-\widehat{\bar{\boldsymbol{\mu}}}\|_2 + \|\boldsymbol{\beta}\|_2\|\widehat{\bar{\boldsymbol{\mu}}}-\bar{\boldsymbol{\mu}}\|_2 \\
&\lesssim \sqrt{\frac{J\log n}{n}}\|\boldsymbol{\beta}\|_2.
\end{aligned} \tag{C4}$$

Thirdly, we have

$$\begin{aligned}
& \log|\mathbf{D}\boldsymbol{\Sigma}_1 + \mathbf{I}_J| - \log|\widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1 + \mathbf{I}_J| \\
\leq \; & tr\left\{(\mathbf{D}\boldsymbol{\Sigma}_1 + \mathbf{I}_J)^{-1}(\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1)\right\} \\
= \; & tr\left\{-\mathbf{D}\boldsymbol{\Sigma}_2(\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1)\right\} + tr(\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1) \\
\leq \; & \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F\|\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1\|_F + tr(\widehat{\mathbf{D}}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1) + tr(\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\boldsymbol{\Sigma}_1).
\end{aligned}$$

Since with probability at least $1 - O(1/n)$,

$$\begin{aligned}
& \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F \|\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1\|_F \\
\leq\ & \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F \left(\|(\boldsymbol{\Sigma}_1 - \widehat{\boldsymbol{\Sigma}}_1)\widehat{\mathbf{D}}\|_F + \|\boldsymbol{\Sigma}_1(\widehat{\mathbf{D}} - \mathbf{D})\|_F \right) \\
\leq\ & \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F \left(\|(\boldsymbol{\Sigma}_1 - \widehat{\boldsymbol{\Sigma}}_1)(\widehat{\mathbf{D}} - \mathbf{D})\|_F + \|(\boldsymbol{\Sigma}_1 - \widehat{\boldsymbol{\Sigma}}_1)\mathbf{D}\|_F + \|\boldsymbol{\Sigma}_1(\widehat{\mathbf{D}} - \mathbf{D})\|_F \right) \\
\lesssim\ & \sqrt{\frac{J \log n}{n}} (\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F^2)
\end{aligned}$$

and with probability at least $1 - O(1/n)$,

$$\begin{aligned}
tr(\widehat{\mathbf{D}}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\widehat{\boldsymbol{\Sigma}}_1) \leq\ & \|\widehat{\mathbf{D}}\boldsymbol{\Sigma}_1\|_F \|\mathbf{I}_J - \widehat{\boldsymbol{\Sigma}}_1 \boldsymbol{\Sigma}_1^{-1}\|_F \\
\leq\ & \|(\widehat{\mathbf{D}} - \mathbf{D})\boldsymbol{\Sigma}_1\|_F \|\boldsymbol{\Sigma}_1 - \widehat{\boldsymbol{\Sigma}}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F \|\boldsymbol{\Sigma}_1 - \widehat{\boldsymbol{\Sigma}}_1\|_F \\
\leq\ & \|\widehat{\mathbf{D}} - \mathbf{D}\|_F \|\boldsymbol{\Sigma}_1\|_\infty \|\boldsymbol{\Sigma}_1 - \widehat{\boldsymbol{\Sigma}}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F \|\boldsymbol{\Sigma}_1 - \widehat{\boldsymbol{\Sigma}}_1\|_F \\
\lesssim\ & \sqrt{\frac{J \log n}{n}} \left\{ \sqrt{\frac{J \log n}{n}} (\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F) + \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F \right\} \\
\lesssim\ & \sqrt{\frac{J \log n}{n}} \|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \frac{J \log n}{n} \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F
\end{aligned}$$

and

$$tr(\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\boldsymbol{\Sigma}_1) \leq \|\mathbf{D} - \widehat{\mathbf{D}}\|_F \|\boldsymbol{\Sigma}_1\|_F \lesssim \sqrt{\frac{J \log n}{n}} (\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F) .$$

Note that $tr\left(\boldsymbol{\Sigma}_1^{1/2}\left(\mathbf{D} - \widehat{\mathbf{D}}\right)\boldsymbol{\Sigma}_1^{1/2}\right) = tr(\mathbf{D}\boldsymbol{\Sigma}_1 - \widehat{\mathbf{D}}\boldsymbol{\Sigma}_1)$, hence we have with probability at least $1 - O(1/n)$

$$\Xi_3 \lesssim \sqrt{\frac{J \log n}{n}} (\|\mathbf{D}\boldsymbol{\Sigma}_1\|_F + \|\mathbf{D}\boldsymbol{\Sigma}_2\|_F) \tag{C5}$$

Lastly, by Hoeffding inequality, we have $\widehat{\pi}_k - \pi_k \lesssim \sqrt{\frac{\log n}{n}}$ with probability at least $1 - O(1/n)$, $k = 1, 2$. Thus

$$\Xi_4 = \left| \log\left(\frac{\pi_1}{\pi_2}\right) - \log\left(\frac{\widehat{\pi}_1}{\widehat{\pi}_2}\right) \right| \lesssim |\log(\widehat{\pi}_1 - \pi_1)| + |\log(\widehat{\pi}_2 - \pi_2)| \lesssim \sqrt{\frac{\log n}{n}} \tag{C6}$$

in probability $1 - O(1/n)$.

Combining (C3) to (C6), the lemma has been proved. □

LEMMA **C.14.** *Denote $f_{Q(\boldsymbol{z})}(t)$ the probability density function of $Q(\boldsymbol{z})$. When $\Delta_{M(\boldsymbol{z})} = o(1)$, we have*

$$\int_0^{C\Delta_{M(\boldsymbol{z})}} (1-e^{-t}) f_{Q(\boldsymbol{z})}(t)dt \lesssim \ \Delta_{M(\boldsymbol{z})} \int_0^{C\Delta_{M(\boldsymbol{z})}} f_{Q(\boldsymbol{z})}(t)dt \lesssim \Delta^2_{M(\boldsymbol{z})}.$$

When $\Delta_{M(\boldsymbol{z})} = O(1)$ or $\Delta_{M(\boldsymbol{z})} = \infty$, we have

$$\int_0^{C\Delta_{M(\boldsymbol{z})}} (1-e^{-t}) f_{Q(\boldsymbol{z})}(t)dt \lesssim \ \int_0^{C\Delta_{M(\boldsymbol{z})}} f_{Q(\boldsymbol{z})}(t)dt \lesssim \Delta_{M(\boldsymbol{z})}.$$

Proof. Without loss of generality, assuming $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$. By simple calculation, we have

$$Q(\boldsymbol{z}) = \sum_{j=1}^{J} (\epsilon_j - 1)\chi^2(\delta_j^2) - \sum_{j=1}^{J} \frac{(\mu_{j1} - \mu_{j2})^2}{\lambda_j^{(2)}} (\epsilon_j - 1)^{-1} - \sum_{j=1}^{J} \log \epsilon_j, \tag{C7}$$

where $\chi^2(\delta_j^2) = h_j^2$, such that $h_j \sim \mathcal{N}(\delta_j, 1)$, where $\delta_j = \frac{\mu_{j1}-\mu_{j2}}{(\lambda_j^{(2)})^{1/2}} \frac{\epsilon_j^{1/2}}{\epsilon_j - 1}$. Denote $q_z = \sum_{j=1}^{J} (\epsilon_j - 1)\chi_j^2(\delta_j)$. To estimate the density of q_z, without loss of generality, we assume that $\epsilon_1 - 1 \geq \epsilon_2 - 1 \geq \cdots \geq 0$, otherwise, q_z can always be represented as the subtraction of two linear combinations of non-central chi-square random variables with positive coefficients, whose density function can be derived by convolution, thus, the boundedness of the density can be obtained by this simple case. As suggested in [68] , define

$$A_k = \sum_{j=1}^{J} \left\{ (\epsilon_j - 1)^k + k(\epsilon_j - 1)^{k-1} \frac{(\mu_{j1} - \mu_{j2})^2}{\lambda_j^{(2)}} + (\epsilon_j - 1)^{k-2} \frac{(\mu_{j1} - \mu_{j2})^2}{\lambda_j^{(2)}} \right\},$$

then $\mu_q := E(q_z) = \sum_{j=1}^{J} \left\{ \epsilon_j - 1 + \frac{(\mu_{j1}-\mu_{j2})^2}{\lambda_j^{(2)}} + \frac{(\mu_{j1}-\mu_{j2})^2}{\lambda_j^{(2)}} (\epsilon_j - 1)^{-1} \right\}$ and

$$\sigma_q := SD(q_z) = \sqrt{2 \sum_{j=1}^{J} \left\{ (\epsilon_j - 1)^2 + 2 \frac{(\mu_{j1} - \mu_{j2})^2}{\lambda_j^{(2)}} \epsilon_j \right\}}.$$

Define $s_1^2 = A_3^2 A_2^{-3}$ and $s_2 = A_4 A_2^{-2}$.

i) When $s_1^2 > s_2$, define $\mu_\chi = \frac{s_1 - 2\sqrt{s_1^2 - s_2}}{(s_1 - \sqrt{s_1^2 - s_2})^3}$, $\sigma_\chi = \frac{\sqrt{2}}{s_1 - \sqrt{s_1^2 - s_2}}$, $D_q = \frac{\mu_\chi}{\sigma_\chi}\sigma_q - \mu_q$ and $U_q = \frac{\sqrt{s_1^2 - s_2}}{(s_1 - \sqrt{s_1^2 - s_2})^3}$. Denote $\gamma = \frac{s_1 - 3\sqrt{s_1^2 - s_2}}{(s_1 - \sqrt{s_1^2 - s_2})^3}$. Then, the probability density function of q_z is approximately by noncentral chi-square distribution with noncentrality parameter U_q and degree of freedom γ as following

$$f_q(t) = \frac{\sigma_\chi}{2\sigma_q} \exp\left(-\frac{K_q + U_q}{2}\right) \left(\frac{K_q}{U_q}\right)^{\gamma/4 - 1/2} I_{\frac{\gamma}{2} - \gamma}\left(\sqrt{U_q K_q}\right),$$

where $K_q = \frac{\sigma_\chi}{\sigma_q}(t + D_q)$, $I_{\gamma/2-\gamma}(\cdot)$ is a modified Bessel function. Note that

$$\begin{aligned} f_q(t) &\leq \frac{\sigma_\chi}{2\sigma_q} {U_q}^{1/2 - \gamma/4} \exp\left(-\frac{U_q + K_q}{2}\right) K_q^{\gamma/4 - 1/2} \\ &\quad \times (U_q K_q)^{\gamma/4 - 1/2} \exp\left(\sqrt{U_q K_q}\right) 2^{1 - \gamma/2} \Gamma^{-1}(\gamma/2 - 1) \\ &\leq \frac{\sigma_\chi}{2^{\frac{\gamma}{2}}\sigma_q} \Gamma^{-1}(\gamma/2 - 1) \exp\left\{-(U_q + K_q)/2 + \sqrt{U_q K_q}\right\}. \end{aligned}$$

The second last inequality is obtained from [74] and Bessel function has the inequality $1 < \Gamma(\nu + 1)\left(\frac{2}{x}\right)^\nu I_\nu(x) < \cosh x \leq e^x$, where $\nu > -1/2$, $x > 0$. Here we can still find that $f_q(t)$ has lower bound $\frac{\sigma_\chi}{2^{\frac{\gamma}{2}}\sigma_q}\Gamma^{-1}(\gamma/2 - 1)\exp\{-(U_q + K_q)/2\}$, which will be used later. Then we have

$$f_{Q(\boldsymbol{z})}(t) \leq \frac{\sigma_\chi}{2^{\frac{\gamma}{2}}\sigma_q} \Gamma^{-1}(\gamma/2 - 1) \exp\left\{-\frac{\sigma_\chi L_J}{2\sigma_q} - \frac{K_q + U_q}{2} + \sqrt{U_q\left(K_q + \frac{\sigma_\chi}{\sigma_q} L_J\right)}\right\},$$

where $L_J = \sum_{j=1}^J \frac{(\mu_{j1} - \mu_{j2})^2}{\lambda_j^{(2)}}(\epsilon_j - 1)^{-1} + \sum_{j=1}^J \log \epsilon_j$. The upper bound is a function increasing when $t < \frac{\sigma_q}{\sigma_\chi} U_q - D_q - L_J$, and decreasing otherwise. When $t = \frac{\sigma_q}{\sigma_\chi} U_q - D_q - L_J$, the global maximal value of density is $\frac{\sigma_\chi}{2^{\frac{\gamma}{2}}\sigma_q}\Gamma^{-1}(\gamma/2 - 1)$.

When $\Delta_{M(\boldsymbol{z})} = o(1)$, then

$$\int_0^{C\Delta_{M(\boldsymbol{z})}} (1 - e^{-t}) f_{Q(\boldsymbol{z})}(t) dt \lesssim \Delta_{M(\boldsymbol{z})} \int_0^{C\Delta_{M(\boldsymbol{z})}} f_{Q(\boldsymbol{z})}(t) dt \lesssim \Delta_{M(\boldsymbol{z})}^2 \frac{\sigma_\chi}{\sigma_q} \frac{1}{2^{\frac{\gamma}{2}}\Gamma(\frac{\gamma}{2} - 1)},$$

otherwise

$$\int_0^{C\Delta_{M(\boldsymbol{z})}}(1-e^{-t})f_{Q(\boldsymbol{z})}(t)dt \lesssim \int_0^{C\Delta_{M(\boldsymbol{z})}} f_{Q(\boldsymbol{z})}(t)dt \lesssim \Delta_{M(\boldsymbol{z})}\frac{\sigma_\chi}{\sigma_q}\frac{1}{2^{\frac{\gamma}{2}}\Gamma(\frac{\gamma}{2}-1)}.$$

ii) When $s_1^2 \leq s_2$, noncentrality parameter $U_q = 0$, $\gamma = s_1^{-2}$. The probability density function of q_z is

$$f_q(t) = \frac{\sigma_\chi}{\sigma_q}\frac{1}{2^{\gamma/2}\Gamma(\gamma/2)}\exp\left(-K_q/2\right)K_q^{\gamma/2-1}.$$

Note that $\frac{\sigma_\chi}{\sigma_q}(q_z + D_q) \sim \chi^2_\gamma$. Given the parameter space Θ, scale parameter $\sigma_\chi/\sigma_q < \infty$, thus $f_q(t) < \infty$ when $s_1^2 \leq s_2$.

Hence, when $\Delta_{M(\boldsymbol{z})} \to 0$, we have

$$\int_0^{C\Delta_{M(\boldsymbol{z})}}(1-e^{-t})f_{Q(\boldsymbol{z})}(t)dt \lesssim \Delta_{M(\boldsymbol{z})}\int_0^{C\Delta_{M(\boldsymbol{z})}} f_{Q(\boldsymbol{z})}(t)dt \lesssim \Delta^2_{M(\boldsymbol{z})}\frac{\sigma_\chi}{\sigma_q}\frac{1}{\sqrt{4\pi\gamma}} \lesssim \Delta^2_{M(\boldsymbol{z})},$$

otherwise we have

$$\int_0^{C\Delta_{M(\boldsymbol{z})}}(1-e^{-t})f_{Q(\boldsymbol{z})}(t)dt \lesssim \int_0^{C\Delta_{M(\boldsymbol{z})}} f_{Q(\boldsymbol{z})}(t)dt \lesssim \Delta_{M(\boldsymbol{z})}\frac{\sigma_\chi}{\sigma_q}\frac{1}{\sqrt{4\pi\gamma}} \lesssim \Delta_{M(\boldsymbol{z})}.$$

□

LEMMA **C.15.** *Consider parameter space Θ, we have*

$$\sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}_J) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})\right] \lesssim \frac{J\log n}{n} + g(J;\Theta),$$

where $\widehat{G}_J$ is the proposed classifier for first J scores. Thus we can easily conclude the least upper bound

$$\inf_{\widehat{G}}\sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})\right] \lesssim \frac{J^*\log n}{n},$$

where J^* satisfies $\frac{J^*\log n}{n} = g(J^*;\Theta)$.

Proof. Without loss of generality, assuming $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$.

$$
\begin{aligned}
& R_{\boldsymbol{\theta}}(\widehat{G}_J) - R_{\boldsymbol{\theta}}(G_J^*) \\
= \; & \frac{1}{2}\int_{Q(\boldsymbol{z})>0} \frac{\pi_1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_1|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_1)} d\boldsymbol{z} \\
& + \frac{1}{2}\int_{Q(\boldsymbol{z})\le 0} \frac{\pi_2}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_2|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}_2^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_2)} d\boldsymbol{z} \\
& - \frac{1}{2}\int_{\widehat{Q}(\boldsymbol{z})>0} \frac{\pi_1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_1|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_1)} d\boldsymbol{z} \\
& - \frac{1}{2}\int_{\widehat{Q}(\boldsymbol{z})\le 0} \frac{\pi_2}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_2|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}_2^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_2)} d\boldsymbol{z} \\
= \; & \frac{1}{2}\int_{Q(\boldsymbol{z})>0} \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_1|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_1) - \log|\boldsymbol{\Sigma}_1|/2} \left(1 - e^{-Q(\boldsymbol{z})}\right) d\boldsymbol{z} \\
& - \frac{1}{2}\int_{\widehat{Q}(\boldsymbol{z})>0} \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_1|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_1) - \log|\boldsymbol{\Sigma}_1|/2} \left(1 - e^{-Q(\boldsymbol{z})}\right) d\boldsymbol{z} \\
\le \; & \frac{1}{2}\int_{Q(\boldsymbol{z})>0, \widehat{Q}(\boldsymbol{z})\le 0} \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_1|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_1) - \log|\Sigma_1|/2} \left(1 - e^{-Q(\boldsymbol{z})}\right) d\boldsymbol{z} \\
= \; & \frac{1}{2}\int_{Q(\boldsymbol{z})>0, Q(\boldsymbol{z})\le Q(\boldsymbol{z})-\widehat{Q}(\boldsymbol{z})} \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}_1|^{1/2}} e^{-1/2(\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_1) - \log|\boldsymbol{\Sigma}_1|/2} \\
& \times \left(1 - e^{-Q(\boldsymbol{z})}\right) d\boldsymbol{z} \\
= \; & \frac{1}{2} E_{\boldsymbol{z}\sim\mathcal{N}(\boldsymbol{\mu_1},\boldsymbol{\Sigma}_1)} \left\{(1 - e^{-Q(\boldsymbol{z})}) \mathbb{I}\left\{0 < Q(\boldsymbol{z}) \le M(\boldsymbol{z})\right\} \mathbb{I}\left(M(\boldsymbol{z}) \lesssim \Delta_{M(\boldsymbol{z})}\right)\right\} \\
& + \frac{1}{2} E_{\boldsymbol{z}\sim\mathcal{N}(\boldsymbol{\mu_1},\boldsymbol{\Sigma}_1)} \left\{(1 - e^{-Q(\boldsymbol{z})}) \mathbb{I}\left(0 < Q(\boldsymbol{z}) \le M(\boldsymbol{z})\right) \mathbb{I}\left(M(\boldsymbol{z}) \gtrsim \Delta_{M(\boldsymbol{z})}\right)\right\} \\
\lesssim \; & E_{\boldsymbol{z}\sim\mathcal{N}(\boldsymbol{\mu_1},\boldsymbol{\Sigma}_1)} \left\{(1 - e^{-Q(\boldsymbol{z})}) \mathbb{I}\left(0 < Q(\boldsymbol{z}) \lesssim \Delta_{M(\boldsymbol{z})}\right)\right\} + n^{-1}.
\end{aligned}
$$

By Lemma C.13, we have $P\left\{M(\boldsymbol{z}) \lesssim \Delta_{M(\boldsymbol{z})}\right\} = 1 - O(1/n)$. Hence, $R_{\boldsymbol{\theta}}(\widehat{G}_J) - R_{\boldsymbol{\theta}}(G_J^*) \lesssim \int_0^{C\Delta_{M(\boldsymbol{z})}} (1 - e^{-t}) f_{Q(\boldsymbol{z})}(t) dt + n^{-1}$. By Lemma C.14 we have

$$
R_{\boldsymbol{\theta}}(\widehat{G}_J) - R_{\boldsymbol{\theta}}(G_J^*) \lesssim \frac{J \log n}{n}. \tag{C8}
$$

Next, we approximate the first J scores and the whole process. Note that

$$
R_{\boldsymbol{\theta}}(G_J^*) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*) \asymp P(Q_\infty(Z) > 0) - P(Q(\boldsymbol{z}) > 0) = \int_0^\infty (f_\infty(t) - f_{Q(\boldsymbol{z})}(t)) dt,
$$

where f_∞ is the density function of Q_∞. Denote

$$R_J = Q_\infty - Q_J = \sum_{j=J+1}^{\infty} (\epsilon_j - 1)\chi^2(\delta_j^2) - \sum_{j=J+1}^{\infty} \frac{(\mu_{j1} - \mu_{j2})^2}{\lambda_j^{(2)}}(\epsilon_j - 1)^{-1} - \sum_{j=J+1}^{\infty} \log \epsilon_j.$$

Then we have

$$\begin{aligned} & P(Q_\infty > 0) - P(Q_J > 0) \\ = \; & P(Q_\infty > 0) - P(Q_\infty > R_J) = P(Q_\infty > 0) - E\left[P\left(Q_\infty > R_J | R_J\right)\right] \\ = \; & \left(1 - \int_{-\infty}^{0} f_\infty(t)dt\right) - E\left(1 - \int_{-\infty}^{R_J} f_\infty(t)dt\right) = E\left(\int_{R_J}^{0} f_\infty(t)dt\right). \end{aligned}$$

When $0 < f_\infty(t) < \infty$, we have

$$E\left(\int_0^{R_J} f_\infty(t)dt\right) \asymp E\left(R_J\right).$$

Recall in (C7), $\chi^2(\delta_j^2) = h_j^2$ and $h_j \sim \mathcal{N}(\delta_j, 1)$, we have $E\left(R_J\right) = \sum_{j=J+1}^{\infty}(\epsilon_j - 1 - \log \epsilon_j) + \sum_{j=J+1}^{\infty} \frac{(\mu_{j1}-\mu_{j2})^2}{\lambda_j^{(2)}}$. Given parameter space Θ, ϵ_j's are constants for any $j \in \mathbb{N}$, we have $E\left(R_J\right) \asymp \sum_{j=J+1}^{\infty}\left(\epsilon_j - 1\right)^2 + \sum_{j=J+1}^{\infty} \frac{(\mu_{j1}-\mu_{j2})^2}{\lambda_j^{(2)}}$. Hence, we have

$$R_{\boldsymbol{\theta}}(G_J^*) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*) \asymp \sum_{j=J+1}^{\infty}\left(\epsilon_j - 1\right)^2 + \sum_{j=J+1}^{\infty} \frac{\left(\mu_{j1} - \mu_{j2}\right)^2}{\lambda_j^{(2)}}. \tag{C9}$$

Hence, by (C8) and (C9), we have

$$\begin{aligned} & \inf_{\widehat{G}} \sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*)\right] \\ = \; & \inf_{\widehat{G}} \sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G_J^*)\right] + \sup_{\boldsymbol{\theta}\in\Theta}\left[R_{\boldsymbol{\theta}}(G_J^*) - R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}^*)\right] \lesssim \frac{J\log n}{n} + g(J;\Theta). \end{aligned}$$

□

LEMMA **C.16.** *For some n large enough and absolute constants C_{11} and C_{12},*

i) When $M \le M^*$, there exists a $J_1^* \equiv J_1^*(M,n) \le M$ being the unique solution of $C_{11}f(M) = g(J;\Theta)/J$, where $f(M) = f_1^2(M) \vee f_2^2(M)$ for each $M \ge 1$;

ii) When $M > M^*$, there exists a $J_2^* \equiv J_2^*(M,n) \leq M$ being the unique solution of $C_{12}\log n/n = g(J;\Theta)/J$.

Proof. First, note that $f(x)$ and $g(x;\Theta)/x$ are two monotone decreasing functions which converge to 0 as $x \to \infty$.

When $M \leq M^*$, $\log n/n = o\left(f(M)\right)$, thus $f(M)$ dominates the MER for the first J scores. Without loss of generality, we assume $g(1;\Theta) \geq f(1)$, since $g(1;\Theta)$ and $f(1)$ are both absolute constants, we can always realize it by rescaling. For any fixed M, if $f(M) \geq g(M;\Theta)/M$, there must be a J_1^* such that $f(M) = g(J_1^*;\Theta)/J_1^*$; Otherwise, there exists a constant $r_{11} > 1$, such that $f(M) \geq g(r_{11}M;\Theta)/(r_{11}M)$. Let $h(r) = g(rM;\Theta)/g(M;\Theta)$, $r > 1$, then it's easy to see that $h(r)$ is a monotone decreasing function of r, when $r = r_{11}$, there exists a $r_{12} > 1$, such that $h(r_{11}) \geq r_{12}$. Thus we have $r_{11}f(M) \geq r_{12}g(M;\Theta)/M$, i.e. $r_1 f(M) \geq g(M;\Theta)/M$, where $r_1 = r_{11}/r_{12}$.

When $M > M^*$, $\log n/n$ dominates the MER for the first J scores. For some large n, $\log n/n < g(1;\Theta)$ is naturally satisfied. For any fixed n and M, if $\log n/n \geq g(M;\Theta)/M$, there must be a J_2^* such that $\log n/n = g(J_2^*;\Theta)/J_2^*$; Otherwise, since $f(M) < \log n/n < g(M;\Theta)/M$, we can get the similar result with another constant r_2 such that $r_2 \log n/n \geq g(M;\Theta)/M$. □

C.2 Proof of Theorem 4.1

Proof. Theorem 4.1 can be derived easily from Lemma C.15. □

C.3 Proof of Proposition 4.1

Proof. Consider parameter space $\Theta(\alpha)$, for any $\theta \in \Theta(\alpha)$, we have

$$\sum_{j=J+1}^{\infty}(\epsilon_j - 1)^2 \asymp \sum_{j=J+1}^{\infty} j^{-a} \asymp \int_{J+1}^{\infty} t^{-a}dt = (a-1)^{-1}(J+1)^{1-a}$$

and

$$\sum_{j=J+1}^{\infty} \frac{(\mu_{j1}-\mu_{j2})^2}{\lambda_j^{(2)}} \asymp \sum_{j=J+1}^{\infty} j^{-b} \asymp \int_{J+1}^{\infty} t^{-b}dt = (b-1)^{-1}(J+1)^{1-b}.$$

Since $a, b > \alpha$,

$$g(J;\Theta) = \sup_{a,b} \left\{(a-1)^{-1}(J+1)^{1-a} + (b-1)^{-1}(J+1)^{1-b}\right\} \asymp J^{-(\alpha-1)}.$$

□

C.4 Proof of Theorem 4.2

Proof. For any $3 \le \ell_1 < \ell_2 < \ell_3 < J$, define the following partial summations based on the definitions in the proof of Lemma C.14.

$$A_k(\ell_1,\ell_2) = \sum_{j=\ell_1}^{\ell_2} \left[(\epsilon_j-1)^k + k(\epsilon_j-1)^{k-1}\frac{(\mu_{j1}-\mu_{j2})^2}{\lambda_j^{(2)}} + (\epsilon_j-1)^{k-2}\frac{(\mu_{j1}-\mu_{j2})^2}{\lambda_j^{(2)}}\right],$$

$s_1^2(\ell_1,\ell_2) = A_3^2(\ell_1,\ell_2)A_2^{-3}(\ell_1,\ell_2)$, $s_2(\ell_1,\ell_2) = A_4(\ell_1,\ell_2)A_2^{-2}(\ell_1,\ell_2)$,
$\sigma_\chi(\ell_1,\ell_2) = \frac{\sqrt{2}}{s_1(\ell_1,\ell_2)-\sqrt{s_1^2(\ell_1,\ell_2)-s_2(\ell_1,\ell_2)}}$, $\sigma_q(\ell_1,\ell_2) = \sqrt{2\sum_{j=\ell_1}^{\ell_2}\left[(\epsilon_j-1)^2 + 2\frac{\Delta\mu_j^2}{\lambda_j^{(2)}}\epsilon_j\right]}$,
$D_q(\ell_1,\ell_2) = \frac{\mu_\chi(\ell_1,\ell_2)}{\sigma_\chi(\ell_1,\ell_2)}\sigma_q(\ell_1,\ell_2) - \mu_q(\ell_1,\ell_2)$, $U_q(\ell_1,\ell_2) = \frac{\sqrt{s_1^2(\ell_1,\ell_2)-s_2(\ell_1,\ell_2)}}{(s_1(\ell_1,\ell_2)-\sqrt{s_1^2(\ell_1,\ell_2)-s_2(\ell_1,\ell_2)})^3}$, $\gamma(\ell_1,\ell_2) =$
$\frac{s_1(\ell_1,\ell_2)-3\sqrt{s_1^2(\ell_1,\ell_2)-s_2(\ell_1,\ell_2)}}{(s_1(\ell_1,\ell_2)-\sqrt{s_1^2(\ell_1,\ell_2)-s_2(\ell_1,\ell_2)})^3}\mathbb{I}(s_1(\ell_1,\ell_2)^2 > s_2(\ell_1,\ell_2)) + s_1^{-2}(\ell_1,\ell_2))\mathbb{I}(s_1^2(\ell_1,\ell_2) \le s_2(\ell_1,\ell_2))$.

Define a particular parameter space for $\{\lambda_j\}_{j=1}^J$,

$$\begin{aligned}\mathcal{B}_J \;=\; & \left\{\{\lambda_j\}_{j=1}^J : R_{3,\ell_1}\exp\left\{\frac{-U_q(3,\ell_1)-\sum_{j=3}^{\ell_1}\lambda_j}{2}\right\} > c_0^{-1}, \ell_1 < J^*/2,\right. \\ & \left.\ell_2-\ell_1, \ell_3-\ell_2 \ge J^*/2\right\},\end{aligned}$$

where $R_{\ell_1,\ell_2} = \frac{\sigma_\chi(\ell_1,\ell_2)\sigma_q^{-1}(\ell_1,\ell_2)}{2^{\gamma(\ell_1,\ell_2)/2-1}\Gamma(\gamma(\ell_1,\ell_2)/2-1)}$, $H_{\ell_1,\ell_2} = U_q(\ell_1,\ell_2) + \sigma_\chi(\ell_1,\ell_2)\sigma_q(\ell_1,\ell_2)^{-1}(D_q(\ell_1,\ell_2) + \sum_{j=\ell_1}^{\ell_2}\lambda_j)$ and

$$c_0 = \inf_{3\le\ell_1,\ell_2,\ell_3\le J} R_{\ell_1,\ell_2}R_{\ell_2,\ell_3}\exp\left\{\frac{-H_{\ell_1,\ell_2} - H_{\ell_2,\ell_3} - \sigma_\chi(\ell_2,\ell_3)\sigma_q^{-1}(\ell_2,\ell_3)(\sum_{j=\ell_2}^{\ell_3}\lambda_j+1)}{2}\right\} \tag{C10}$$

is a constant. Let $\boldsymbol{e}_j$ be the vector with j-th element 1 and 0 otherwise. Let $\boldsymbol{E}_j$ be the diagonal matrix with j-th diagonal element 1 and 0 otherwise. Let τ_1, $\widetilde{\tau}_1$, $\widetilde{\tau}_2$ be some constants small enough, which are specified later. For any $J > 4J^*$, we consider the parameter space

$$\Theta_u = \Big\{\boldsymbol{\theta}_u = (\tau_1\boldsymbol{e}_1 + \widetilde{\tau}_1\boldsymbol{e}_2, -\tau_1\boldsymbol{e}_1 - \widetilde{\tau}_1\boldsymbol{e}_2, \boldsymbol{\Sigma}_1^u, \boldsymbol{\Sigma}_2) : \boldsymbol{\Sigma}_2 = \text{diag}\left(\lambda_1,\ldots,\lambda_J\right), \{\lambda_j\}_{1\le j\le J}\in\mathcal{B}_J,$$
$$(\boldsymbol{\Sigma}_1^u)^{-1} = \boldsymbol{\Sigma}_2^{-1} + \tau_2\sqrt{\frac{\log n}{n}}\sum_{j=3}^{J-1}u_j\boldsymbol{E}_j + \widetilde{\tau}_2\boldsymbol{E}_J, \boldsymbol{u} = (u_1,\ldots,u_J)\in\mathcal{A}_{J,J^*}\Big\},$$

where $\mathcal{A}_{J,J^*}$ is defined in Lemma C.9. Thus, we have $\Theta_u \subseteq \Theta$.

We first calculate KL divergence between two distributions for u and u'. By Lemma C.9, we have

$$\begin{aligned} KL(P_{\boldsymbol{\theta}_u}, P_{\boldsymbol{\theta}_{u'}}) &= \frac{1}{2}\left\{\log\frac{|\boldsymbol{\Sigma}_1^{u'}|}{|\boldsymbol{\Sigma}_1^u|} - J + tr\left(\left(\boldsymbol{\Sigma}_1^{u'}\right)^{-1}\boldsymbol{\Sigma}_1^u\right)\right\} \\ &\le \frac{1}{4}\sum_{j=3}^{J-1}\left\{\tau_2^2\frac{\log n}{n}(u_j-u_j')^2 + o\left(\frac{\log n}{n}\right)\right\} \\ &\le \frac{\tau_2^2}{2}\frac{J^*\log n}{n} + o\left(\frac{J^*\log n}{n}\right) \le \alpha\frac{\log N}{n}. \end{aligned}$$

We choose some constant τ_2 such that $\alpha \in (0, 1/8)$. By Lemmas C.7 and C.8, given any classifier $\widehat{G}$ in the considered parameter space Θ_u, we have

$$\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}_u})\right] + \left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}_{u'}})\right] \ge \frac{1}{2}\left(L_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}_{u'}}) - \sqrt{\frac{KL\left(P_{\boldsymbol{\theta}_u}, P_{\boldsymbol{\theta}_{u'}}\right)}{2}}\right)^2.$$

Since $KL(P_{\boldsymbol{\theta}_u}, P_{\boldsymbol{\theta}_{u'}}) \leq \frac{\tau_2^2}{2}\frac{J^* \log n}{n}$, it's sufficient to show that $L_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}_{u'}}) \geq c\sqrt{\frac{J^* \log n}{n}}$ for some $c > \tau_2/2$. Now, we need to calculate

$$
\begin{aligned}
& (\boldsymbol{z}-\boldsymbol{\mu}_1)^\top \mathbf{D}^u (\boldsymbol{z}-\boldsymbol{\mu}_1) - 2\boldsymbol{\delta}^\top \boldsymbol{\Omega}_2^u (\boldsymbol{z}-\bar{\boldsymbol{\mu}}) - \log(|\boldsymbol{\Sigma}_1^u|/|\boldsymbol{\Sigma}_2|) \\
= \; & -\tau_2\sqrt{\frac{\log n}{n}}\sum_{j=3}^{J-1} u_j z_j^2 + \frac{4\tau_1}{\lambda_1} z_1 + \frac{4\widetilde{\tau}_1}{\lambda_2} z_2 - \widetilde{\tau}_2 z_J^2 + \log\left(1+\lambda_J \widetilde{\tau}_2\right) \\
& + \sum_{j=3}^{J-1} \log\left(1+\lambda_j \tau_2 \sqrt{\frac{\log n}{n}} u_j\right) \\
= \; & -\tau_2\sqrt{\frac{\log n}{n}}\sum_{j=3}^{J-1} u_j (z_j^2-\lambda_j) + \frac{4\tau_1}{\lambda_1} z_1 + \frac{4\widetilde{\tau}_1}{\lambda_2} z_2 - \widetilde{\tau}_2 (z_J^2-\lambda_J) - \frac{1}{2}\widetilde{\tau}_2^2 \lambda_J^2 \\
& - \frac{1}{2}\sum_{j=3}^{J-1} \lambda_j^2 \tau_2^2 \frac{\log n}{n} u_j + o\left(\frac{\log n}{n}\right) \\
= \; & -\tau_2\sqrt{\frac{\log n}{n}}\sum_{j=3}^{J-1} u_j (z_j^2-\lambda_j) + \frac{4\tau_1}{\lambda_1} z_1 + \frac{4\widetilde{\tau}_1}{\lambda_2} z_2 - \widetilde{\tau}_2 (z_J^2-\lambda_J) - \frac{1}{2}\widetilde{\tau}_2^2 \lambda_J^2 \\
& + O\left(\tau_2^2 \frac{J^* \log n}{n}\right) + o\left(\frac{J^* \log n}{n}\right).
\end{aligned}
$$

Without loss of generality, we assume that $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $u_j = u'_j = 1$ when $i = 3, \ldots, m_1$, $u_j = 1 - u'_j = 1$ when $j = m_1+1, \ldots, m_2$, $u_j = 1 - u'_j = 0$ when $j = m_2+1, \ldots, m_3$ and $u_j = u'_j = 0$ when $j = m_3+1, \ldots, J-1$. Then for $G^*_{\boldsymbol{\theta}_u}$, we have decision function

$$
\begin{aligned}
Q_u(\boldsymbol{z}) \; = \; & -\tau_2\sqrt{\frac{\log n}{n}}\left(\sum_{j=3}^{m_1}(z_j^2-\lambda_j) + \sum_{j=m_1+1}^{m_2}(z_j^2-\lambda_j)\right) + \frac{4\tau_1}{\lambda_1} z_1 \\
& + \frac{4\widetilde{\tau}_1}{\lambda_2} z_2 - \widetilde{\tau}_2 (z_J^2-\lambda_J) - \frac{1}{2}\widetilde{\tau}_2^2 \lambda_J^2 + O\left(\tau_2^2 \frac{J^* \log n}{n}\right) + o\left(\frac{J^* \log n}{n}\right),
\end{aligned}
$$

and for $G^*_{\boldsymbol{\theta}_{u'}}$, we have decision function

$$
\begin{aligned}
Q_{u'}(\boldsymbol{z}) \; = \; & -\tau_2\sqrt{\frac{\log n}{n}}\left[\sum_{j=3}^{m_1}(z_j^2-\lambda_j) + \sum_{j=m_2+1}^{m_3}(z_j^2-\lambda_j)\right] + \frac{4\tau_1}{\lambda_1} z_1 \\
& + \frac{4\widetilde{\tau}_1}{\lambda_2} z_2 - \widetilde{\tau}_2 (z_J^2-\lambda_J) - \frac{1}{2}\widetilde{\tau}_2^2 \lambda_J^2 + O\left(\tau_2^2 \frac{J^* \log n}{n}\right) + o\left(\frac{J^* \log n}{n}\right).
\end{aligned}
$$

Let $T_1 = -\tau_2\sqrt{\frac{\log n}{n}}\sum_{j=3}^{m_1}(z_j^2-\lambda_j) + \frac{4\tau_1}{\lambda_1}z_1 + \frac{4\widetilde{\tau}_1}{\lambda_2}z_2 - \widetilde{\tau}_2(z_J^2-\lambda_J) - \frac{1}{2}\widetilde{\tau}_2^2\lambda_J^2 + O\left(\tau_2^2\frac{J^*\log n}{n}\right)$, $T_2 = -\tau_2\sqrt{\frac{\log n}{n}}\sum_{j=m_1+1}^{m_2}(z_j^2-\lambda_j)$, $T_3 = -\tau_2\sqrt{\frac{\log n}{n}}\sum_{j=m_2+1}^{m_3}(z_j^2-\lambda_j)$, then

$$\begin{aligned}\left\{G^*_{\boldsymbol{\theta}_u} \neq G^*_{\boldsymbol{\theta}_{u'}}\right\} &= \left\{T_3 + o\left(\frac{J^*\log n}{n}\right) < T_1 \leq T_2 + o\left(\frac{J^*\log n}{n}\right),\right.\\ &\qquad \left. T_2 + o\left(\frac{J^*\log n}{n}\right) < T_1 \leq T_3 + o\left(\frac{J^*\log n}{n}\right)\right\}.\end{aligned}$$

Thus, $L_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}_{u'}}) \geq \frac{1}{2}P_{\boldsymbol{z}\sim\mathcal{N}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_2)}\left(T_2 < T_1 \leq T_3\right) + o\left(\frac{J^*\log n}{n}\right)$.

Notice that $Ez_j = 0$ and $Ez_j^2 = \lambda_j$ for all $j \geq 3$. Since $\sum_{j=m_1+1}^{m_2} z_j^2 = \sum_{j=m_1+1}^{m_2}\lambda_j h_j^2$ and $\sum_{j=m_2+1}^{m_3} z_j^2 = \sum_{j=m_2+1}^{m_3}\lambda_j h_j^2$, where $h_j \sim \mathcal{N}(0,1)$, following the same procedure in the proof of Lemma C.14, we have $A'_k = \sum_{j=m_1+1}^{m_2}\lambda_j^k$, $\mu'_q = \sum_{j=m_1+1}^{m_2}\lambda_j$ and $\sigma'_q = \sqrt{2\sum_{j=m_1+1}^{m_2}\lambda_j^2}$, which are all constants. Without loss of generality, we have $s_1^2 > s_2$. Using the lower bound of probability density, with constants σ_χ/σ_q, D_q and U_q, we have

$$\begin{aligned}&E\left[(T_3-T_2)\mathbb{I}\left\{-\tau_2\sqrt{\frac{J^*\log n}{n}} \leq T_2 < T_3 < \tau_2\sqrt{\frac{J^*\log n}{n}}\right\}\right]\\ \geq\ & E\left[(T_3-T_2)\mathbb{I}\left\{-\tau_2\sqrt{\frac{J^*\log n}{n}} \leq T_2 < 0, 0 < T_3 < \tau_2\sqrt{\frac{J^*\log n}{n}}\right\}\right]\\ \geq\ & 2\tau_2\sqrt{\frac{J^*\log n}{n}}P\left(-\tau_2\sqrt{\frac{J^*\log n}{n}} \leq T_2 < 0\right)P\left(0 < T_3 < \tau_2\sqrt{\frac{J^*\log n}{n}}\right)\\ =\ & 2\tau_2\sqrt{\frac{J^*\log n}{n}}P\left(\sum_{j=m_1+1}^{m_2}\lambda_j \leq \sum_{j=m_1+1}^{m_2} z_j^2 < \sqrt{J^*} + \sum_{j=m_1+1}^{m_2}\lambda_j\right)\\ &\times P\left(\left(-\sqrt{J^*} + \sum_{j=m_2+1}^{m_3}\lambda_j\right)\vee 0 < \sum_{j=m_2+1}^{m_3} z_j^2 < \sum_{j=m_2+1}^{m_3}\lambda_j\right)\\ \geq\ & 2\tau_2\sqrt{\frac{J^*\log n}{n}}P\left(\sum_{j=m_1+1}^{m_2}\lambda_j \leq \sum_{j=m_1+1}^{m_2} z_j^2 < 1 + \sum_{j=m_1+1}^{m_2}\lambda_j\right)\\ &\times P\left(\left(-1 + \sum_{j=m_2+1}^{m_3}\lambda_j\right)\vee 0 < \sum_{j=m_2+1}^{m_3} z_j^2 < \sum_{j=m_2+1}^{m_3}\lambda_j\right)\\ \geq\ & 2c_0\tau_2\sqrt{\frac{J^*\log n}{n}}.\end{aligned}$$

Similar as T_2 and T_3, the density function for $T_1^* = -\tau_2\sqrt{\frac{\log n}{n}}\sum_{j=3}^{m_1}(z_j^2 - \lambda_j)$ has the lower bound

$$-\tau_2\sqrt{\frac{\log n}{n}}\frac{\sigma_\chi}{2^{\frac{\gamma}{2}}\sigma_q\Gamma(\gamma/2-1)}\exp\left(\frac{\tau_2\sqrt{\frac{\log n}{n}}K_q - U_q - \sum_{j=3}^{m_1}\lambda_j}{2}\right),$$

and $\frac{4\tau_1}{\lambda_1}z_1 \sim \mathcal{N}\left(\frac{4\tau_1^2}{\lambda_1}, \frac{16\tau_1^2}{\lambda_1}\right)$, $\frac{4\widetilde{\tau}_1}{\lambda_2}z_2 \sim \mathcal{N}\left(\frac{4\widetilde{\tau}_1^2}{\lambda_2}, \frac{16\widetilde{\tau}_1^2}{\lambda_2}\right)$. Since λ_J is decreasing with J, $-\widetilde{\tau}_2(z_J^2 - \lambda_J) - \frac{1}{2}\widetilde{\tau}_2^2\lambda_J^2 = o_p(1)$ and $O\left(\tau_2^2\frac{J^*\log n}{n}\right) = o\left(\frac{J^*\log n}{n}\right)$ when $\widetilde{\tau}_2$ and τ_2 are sufficiently small, T_1 is the sum of T_1^* and it is a normal random variable. By simple calculation, we have

$$f_{T_1}(t) \;\geq\; c_1c_2\exp\left[-\tau_2\sqrt{\frac{\log n}{n}}\frac{\sigma_\chi}{\sigma_q}\left\{t - \left(\frac{12\tau_1^2}{\lambda_1} + \frac{12\widetilde{\tau}_1^2}{\lambda_2}\right) - D_q\right\}\right],$$

where $c_1 = \frac{\sigma_\chi\sigma_q^{-1}}{2^{\frac{\gamma}{2}}\Gamma(\gamma/2-1)}$ and $c_2 = \exp\left(\frac{-U_q-\sum_{j=3}^{m_1}\lambda_j}{2}\right)$. When $|t| < \tau_2\sqrt{\frac{J^*\log n}{n}}$, and $\widetilde{\tau}_2$ is sufficiently small, we have

$$\begin{aligned}
&\inf_{t:|t|<\tau_2\sqrt{\frac{J^*\log n}{n}}} f(t) \\
\geq\;& c_1c_2\exp\left\{-\tau_2^2\frac{\log n}{n}\frac{\sigma_\chi}{\sigma_q}\sqrt{J^*} - \tau_2\sqrt{\frac{\log n}{n}}\frac{\sigma_\chi}{\sigma_q}\left(\frac{12\tau_1^2}{\lambda_1} + \frac{12\widetilde{\tau}_1^2}{\lambda_2}\right) - \tau_2\sqrt{\frac{\log n}{n}}\frac{\sigma_\chi}{\sigma_q}D_q\right\} \\
\geq\;& c_1c_2C_n\left(\tau_1, \widetilde{\tau}_1, \tau_2\right).
\end{aligned}$$

When $n \to \infty$ and τ_1, $\widetilde{\tau}_1$, $\tau_2 \to 0$, $C_n\left(\tau_1, \widetilde{\tau}_1, \tau_2\right) \to 1$, and given parameter space Θ_u, we have $2c_1c_2 \geq c_0^{-1}$, where c_0 is given in (C10). Note that

$$\begin{aligned}
&P_{\boldsymbol{z}\sim\mathcal{N}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_2)}\left(T_2 < T_1 \leq T_3\right) \\
\geq\;& P_{\boldsymbol{z}\sim\mathcal{N}(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_2)}\left(T_2 < T_1 \leq T_3, -\tau_2\sqrt{\frac{J^*\log n}{n}} \leq T_2 < T_3 < \tau_2\sqrt{\frac{J^*\log n}{n}}\right) \\
=\;& E_{T_2}\left[\mathbb{I}\left\{-\sqrt{\frac{J^*\log n}{n}} \leq T_2 < T_3 < \tau_2\sqrt{\frac{J^*\log n}{n}}\right\}\int_{T_2}^{T_3} f_{T_1}(t)dt\right] \\
\geq\;& \frac{1}{2c_0}E\left[(T_3 - T_2)\mathbb{I}\left\{-\tau_2\sqrt{\frac{J^*\log n}{n}} \leq T_2 < T_3 < \tau_2\sqrt{\frac{J\log n}{n}}\right\}\right] \\
\geq\;& \frac{c_0}{2c_0}\tau_2\sqrt{\frac{J^*\log n}{n}} = \frac{\tau_2}{2}\sqrt{\frac{J^*\log n}{n}}.
\end{aligned}$$

Finally, by Lemma C.8, we can conclude that for any $J > 4J^*$,

$$\inf_{\widehat{G}} \sup_{\boldsymbol{\theta}\in\Theta} E\left[R_{\boldsymbol{\theta}}(\widehat{G}) - R_{\boldsymbol{\theta}}(G^*_{\boldsymbol{\theta}})\right] \gtrsim \frac{J^* \log n}{n}.$$

□

C.5 Proof of Theorem 4.3

Proof. The proof can be easily derived from Lemma C.16 and the proof of Lemma C.12 and Theorem 4.1, thus it is omitted.

□

C.6 Proof of Theorem 4.4

Proof. The proof can be easily derived from the proof of Lemma C.12 and Theorem 4.2, thus it is omitted. □

C.7 Proof of Proposition 4.2

Proof. Consider the parameter space $\Theta^*(c, d, \alpha)$, such that $\sum_{j=M+1}^{\infty} \mu_j^2 \asymp (2c' - 1)^{-1} M^{1-2c'}$ and $\sum_{j=M+1}^{\infty} \lambda_j \asymp (d' - 1)^{-1} M^{1-d'}$. Thus we have

$$\|\widehat{\boldsymbol{\mu}}_J - \boldsymbol{\mu}_J\|_2 \lesssim \sqrt{\frac{\log n}{n}} + \sqrt{J}(2c - 1)^{-1} M^{1-2c},$$

with probability at least $1 - O(1/n)$, and

$$\begin{aligned} \|\widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\|_2 &\lesssim \sqrt{\frac{\log n}{n}} + \sqrt{\frac{\log n}{n}} \frac{1}{\sqrt{M}} \sqrt{\sum_{j=M+1}^{\infty} \lambda_j} + \left(1 + \sqrt{\frac{\log n}{n}}\right) \sum_{j=M+1}^{\infty} \lambda_j \\ &\asymp \sqrt{\frac{\log n}{n}} + \sum_{j=M+1}^{\infty} \lambda_j \asymp \sqrt{\frac{\log n}{n}} + (d - 1)^{-1} M^{1-d}, \end{aligned}$$

with probability at least $1 - 3/n$. Hence, the result can be easily derived from the proof of Lemma C.12 and Proposition 4.1, thus it is omitted. □

Appendix D

Functional Classification via Deep Neural Networks

D.1 Proofs of Theorem 5.1

D.1.1 Preliminary

For any $M > 0$, we define $\epsilon = \max_u M^{-\beta_u^*}$ for simplicity. The following lemma provides an error bound for excess risk.

LEMMA **D.17.** *There exists an $\widetilde{f} \in \mathcal{F}(L, J, \boldsymbol{p}, B)$ and $\widetilde{G} = sign(\widetilde{f})$ satisfying*

$$\sup_{h \in \mathcal{H}} E\left[R_h(\widetilde{G}) - R_h(G_J^*)\right] \lesssim \epsilon^{\alpha+1} + \epsilon\epsilon(J),$$

such that $L \lesssim \log_2 M$, $\|\boldsymbol{p}\|_\infty \lesssim \max_{u=0,\ldots,q} d_{(u+1)J} t_u (M+1)^{t_u}$, $B \lesssim M$, where G_J^* is the Bayes classifier for the first J scores and $J_0 \leq J \lesssim \max_{u=0,\ldots,q} M^{t_u}$.

The following two propositions are implied by Assumption 2.

Proposition D.1. *Assumption 2 implies that $|E\left(Q^* - Q_J\right)| \lesssim \epsilon(J)$.*

Proposition D.2. *(Mild density condition) Assumption 2 implies that $P\left(|Q^*(\boldsymbol{\xi})| < \infty\right) = 1$.*

The proofs of Lemma D.17 and two propositions are provided in Appendix B.

D.1.2 Proof of Theorem 5.1 (i)

Let $u^* = \arg\min_{u=0,\ldots,q} \frac{\beta_u^*(\alpha+1)}{\beta_u^*(\alpha+2)+t_u}$, $\beta^* = \beta_{u^*}$, $t^* = t_{u^*}$, $\beta^{**} = \beta_{u^*}^*$ and $\widetilde{\beta} = \prod_{k=u^*+1}^{q}(\beta_k \wedge 1)$. Without loss of generality, we assume u^* is unique.

For an integer $w \geq 1$, define the regular grid on $\mathbb{R}^{t^*}$ as

$$G_w = \left\{ \left(\frac{2k_1+1}{2w}, \ldots, \frac{2k_{t^*}+1}{2w} \right) : k_\ell \in \{0, \ldots, w-1\}, \ell = 1, \ldots, t^* \right\}.$$

Let $n_w(\boldsymbol{x}) \in G_w$ be the closest point to $\boldsymbol{x} \in \mathbb{R}^{t^*}$ among points in G_w. Let $\mathcal{X}_\ell, \ell = 0, \ldots, m$ be the partition of $\mathbb{R}^{t^*}$ defined in the proof of Theorem 4.1 in [7], where $m \leq w^{t^*}$.

Let $p : \mathbb{R}^+ \to \mathbb{R}^+$ be a nonincreasing infinitely differentiable function such that $p = 1$ on $[0, 1/4]$ and $p = 0$ on $[1/2, \infty)$. For instance, p can be constructed as in [7]: $p(x) = \left(\int_{1/4}^{1/2} p_1(s)ds \right)^{-1} \int_x^\infty p_1(t)dt$, where

$$p_1(x) = \begin{cases} \exp\left\{ \frac{1}{(x-1/4)(x-1/2)} \right\}, & x \in (1/4, 1/2), \\ 0, & \text{otherwise.} \end{cases}$$

Let $h_{u^*} : \mathbb{R}^{t^*} \to \mathbb{R}^+$ be a function defined as $h_{u^*}(\boldsymbol{x}) = w^{-\beta^*} C_p p(\|w(\boldsymbol{x} - n_w(\boldsymbol{x}))\|)$, where $D^s h_{u^*}(\boldsymbol{x}) = q^{|s|-\beta^*} C_p D^s p(\|w(\boldsymbol{x} - n_w(\boldsymbol{x}))\|)$ for any $s \in \mathbb{N}^{t^*}$ such that $|s| \leq \lceil \beta^* \rceil$, and C_p is a constant small enough to ensure $h_{u^*} \in \mathcal{C}^{\beta^*}\left(\mathbb{R}^{t^*}, K^*\right)$ for a constant $K^* > 0$. Here, we require C_p being small so that h_{u^*} has Lipschitz constant K^*.

In the following, we construct a special composition function based on h_{u^*}. For $\overrightarrow{\sigma} = (\sigma_1, \ldots, \sigma_m) \in \{-1, 1\}^m$, let $h(\boldsymbol{x}) = \sum_{j=1}^m \sigma_j h_j(\boldsymbol{x})$ such that $h_j(\boldsymbol{x}) = h_{u^*} \mathbb{I}(\boldsymbol{x} \in \mathcal{X}_j)$. It is easy to verify that $h \in \mathcal{C}^{\beta^*}\left(\mathbb{R}^{t^*}, 2K^*\right)$. Define the following functions

$$\begin{cases} g_u(x_1, \ldots, x_{d_u}) = (x_1, \ldots, x_{d_u}), & u < u^*, \\ g_u(x_1, \ldots, x_{d_u}) = (h(x_1, \ldots, x_{t^*}), 0, \ldots, 0), & u = u^*, \\ g_u(x_1, \ldots, x_{d_u}) = (x_1^{\beta_u \wedge 1}, 0, \ldots, 0), & u > u^*. \end{cases}$$

Let J^0 be a (relatively) large integer up to $O(n^c))$ for some universal positive constant c. For $\boldsymbol{z} \in \mathbb{R}^{J^0}$ and $\boldsymbol{x} \in \mathbb{R}^{t^*}$, define $z_g(\boldsymbol{z})$ as the first element of $g_{u^*} \circ g_{u^*-1} \circ \ldots \circ g_0(\boldsymbol{z})$, and

$h(\boldsymbol{x}) = z_g(\boldsymbol{z})$. Let

$$\begin{aligned}\eta_{\vec{\sigma}}(\boldsymbol{z}) &= \frac{1}{2} + \frac{1}{2} g_q \circ \ldots \circ g_{u^*+1} \circ g_{u^*} \circ g_{u^*-1} \circ \ldots \circ g_0(\boldsymbol{z}) \\ &= \frac{1}{2} + \frac{1}{2} \sum_{j=0}^{m} \sigma_j h_{u^*}^{\widetilde{\beta}}(\boldsymbol{x}) \mathbb{I}\left(\boldsymbol{x} \in \mathcal{X}_j\right).\end{aligned}$$

For all $J \geq J^0$, let $\eta(z') = \eta_{\vec{\sigma}}(\boldsymbol{z})$, where $\boldsymbol{z}' \in \mathbb{R}^J$ and $\boldsymbol{z} \in \mathbb{R}^{J^0}$ is the first J^0 elements of $\boldsymbol{z}'$. It is easy to see that $\eta_{\vec{\sigma}} \in \mathcal{G}\left(q, J^0, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}\right)$, and Assumption 2 is satisfied for all $J \geq J^0$. According to Proposition D.2, Q^* is finite in probability, and it is equivalent to the mild density assumption in [7]. Assumption 1 is equivalent to the margin assumption in [7], which can be justified accordingly. The rest of proof can simply follow the proof of Theorem 4.1 in [7], where for any generic classifier $\widehat{G}$, there exists a universal constant C_1, such that

$$\sup_{h \in \mathcal{H}} E\left[R_h(\widehat{G}) - R_h(G^*)\right] \geq \sup_{h \in \mathcal{H}} E\left[R_h(\widehat{G}) - R_h(G^*_{J^0})\right] \geq C_1 \left(\frac{1}{n}\right)^{S_0}.$$

D.1.3 Proof of Theorem 5.1 (ii)

Proof. Let $\mathcal{B}_{M_0} = \{|\xi_j| \leq M_0 \text{ for all } j \in \mathcal{A}\}$. We first find the minimax upper bound on a bounded set when $\xi_j \in [-M_0, M_0]$ for all $j \in \mathcal{A}$ and $M_0 > 0$. Let the δ in inequality (D9) be $\epsilon^{\alpha+1}$, we have

$$\left(\epsilon^{\alpha+1}\right)^{-\max_u \frac{t_u}{(\alpha+1)\beta_u^*} - \frac{(\alpha+2)}{\alpha+1}} \log^3(\epsilon^{-1}) \lesssim n,$$

which leads to

$$\epsilon^{\alpha+1} \gtrsim \left(\frac{\log^3 n}{n}\right)^{\min_u \frac{\beta_u^*(\alpha+1)}{\beta_u^*(\alpha+2)+t_u}}.$$

Together with Lemma D.28 in Appendix B, the excess risk of the first J scores via DNN satisfies

$$\sup_{h \in \mathcal{H}} E\left[R_h(\widetilde{f}) - R_h(G^*_J)\right] \lesssim \left(n^{-1} \log^3 n\right)^{S_0}.$$

The asymptotic order of L, $\max_{0 \leq \ell \leq L} p_\ell$ and B can be simply derived by letting $\epsilon^{\alpha+1} \asymp \left(\frac{\log^3 n}{n}\right)^{\min_u \frac{\beta_u^*(\alpha+1)}{\beta_u^*(\alpha+2)+t_u}}$ and applying the result in Lemma D.17. Note that the input J satisfies $J \lesssim \max_{1 \leq \ell \leq L} p_\ell \lesssim (n \log^{-3} n)^{S_1}$.

Next, we approximate the first J scores and the whole process. Since

$$R_h(G_J^*) - R_h(G^*) \asymp P(Q^* > 0) - P(Q_J^* > 0),$$

and

$$\begin{aligned} & P(Q^* > 0) - P(Q_J^* > 0) \\ = \; & P(Q^* > 0) - P(Q^* > Q^* - Q_J^*) = P(Q^* > 0) - E\left[P\left(Q_\infty > Q^* - Q_J^* | Q^* - Q_J^*\right)\right] \\ = \; & \left(1 - \int_{-\infty}^{0} f_{Q^*}(t)dt\right) - E\left(1 - \int_{-\infty}^{Q^* - Q_J^*} f_{Q^*}(t)dt\right) = E\left(\int_{Q^* - Q_J^*}^{0} f_{Q^*}(t)dt\right). \end{aligned}$$

By considering $h \in \mathcal{H}$, we have $0 < f_{Q^*}(t) < \infty$, and

$$E\left(\int_{0}^{Q^* - Q_J^*} f_{Q^*}(t)dt\right) \asymp E\left(Q^* - Q_J^*\right).$$

Therefore, by Proposition D.1, $R_h(G_J^*) - R_h(G^*) = O(\epsilon(J))$.

By Assumption 2 and Assumption 3(b), we have $\epsilon(J) = O(J^{-\rho}) \lesssim \left(n^{-1}\log^3 n\right)^{S_0}$. When $\rho > S_0/S_1$, there always exists a constant C' and C'', such that $C'(n\log^{-3} n)^{S_0/\rho} < C''(n\log^{-3} n)^{S_1}$, and the optimal J exists.

Therefore, for those optimal J's, we have

$$\sup_{h\in\mathcal{H}} E\left[R_h(\widehat{G}^{FDNN}) - R_h(G^*)\right] \lesssim \left(\frac{\log^3 n}{n}\right)^{S_0}, \text{ under event } \mathcal{B}_{M_0}.$$

Finally, we control the minimax upper bound for $\xi_j \in \mathbb{R}$ for all $j \in \mathcal{A}$. For any h and $M_0 > 0$, we have

$$\begin{aligned} \mathcal{E}_h(\widehat{G}) &= \mathcal{E}_h(\widehat{G}\mathbb{I}(\mathcal{B}_{M_0})) + \mathcal{E}_h(\widehat{G}\mathbb{I}(\mathcal{B}_{M_0}^c)) \\ &\lesssim \left(\frac{\log^3 n}{n}\right)^{S_0} + P(\mathcal{B}_{M_0}^c). \end{aligned}$$

It is trivial to see that, for any $e > 0$, there exists an $M > 0$, such that

$$P(\mathcal{B}_M) = P(|\xi_j| \leq M \text{ for all } j \in \mathcal{A}) \geq 1 - e, \tag{D1}$$

for all $j \in \mathcal{A}$. Therefore, choose $P(\mathcal{B}_{M_0}^c) \leq \left(\frac{\log^3 n}{n}\right)^{S_0}$, there exists a corresponding M_0, to make the aforementioned asymptotic inequality hold. Choose some constant C_2 which depends on $q, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, \alpha, C, \epsilon(\cdot), \Gamma(\cdot)$, the proof is complete. □

D.2 Technical lemmas

In this section, we provide technical lemmas with their proofs.

For any $J \geq 1$, define regression functions

$$\eta^*(\boldsymbol{\xi}) = \frac{h_1(\boldsymbol{\xi})}{h_1(\boldsymbol{\xi}) + h_{-1}(\boldsymbol{\xi})} = \frac{1}{1 + \exp\{Q^*(\boldsymbol{\xi})\}}$$

and

$$\eta^{(J)}(\boldsymbol{\xi}_J) = \frac{1}{1 + \exp\{Q_J^*(\boldsymbol{\xi}_J)\}}.$$

The following lemma describes the behavior of $\eta^{(J)}$ around $\frac{1}{2}$ for large J.

LEMMA **D.18.** *Assumption 1 and Assumption 2 imply that for any sufficiently small $x > 0$ and $J_0 \geq 1$,*

$$Pr(|\eta^{(J)}(\boldsymbol{\xi}_J) - \frac{1}{2}| \leq x) \lesssim x^\alpha + \epsilon(J), \qquad \forall J \geq J_0. \tag{D2}$$

Proof. We have the following connections between η^* and Q^*:

$$\begin{aligned}
\mathbb{P}\left(|\eta^*(\boldsymbol{\xi}) - \frac{1}{2}| \leq x\right) = &\quad \mathbb{P}\left(\left|\frac{h_1(\boldsymbol{\xi})}{h_1(\boldsymbol{\xi}) + h_{-1}(\boldsymbol{\xi})} - \frac{1}{2}\right| \leq x\right) \\
= &\quad \mathbb{P}\left(\left|\frac{1 - e^{-Q^*}}{1 + e^{-Q^*}}\right| \leq 2x\right) \\
= &\quad \mathbb{P}\left(\frac{1 - 2x}{1 + 2x} \leq e^{-Q^*} \leq \frac{1 + 2x}{1 - 2x}\right) \\
\asymp &\quad \mathbb{P}(-4x \leq Q^* \leq 4x),
\end{aligned} \tag{D3}$$

where the last step is obtained by $\log\left(\frac{1-2x}{1+2x}\right) \asymp -4x$ and $\log\left(\frac{1+2x}{1-2x}\right) \asymp 4x$ when $x = o(1)$. It also holds for $\eta^{(J)}$ and Q_J^* for any $J \geq 1$. The intermediate steps indicate that the equivalence between Assumption 1 and the concentration of η^*.

For any $J \geq J_0$, we have

$$\begin{aligned}
\mathbb{P}(-4x \leq Q_J^* \leq 4x) &= \mathbb{P}(-4x + Q^* - Q_J^* \leq Q^* \leq 4x + Q^* - Q_J^*) \\
&\leq \mathbb{P}(-4x - |Q^* - Q_J^*| \leq Q^* \leq 4x + |Q^* - Q_J^*|) \\
&= \mathbb{P}(-4x - |Q^* - Q_J^*| \leq Q^* \leq 4x + |Q^* - Q_J^*|, |Q^* - Q_J^*| \leq 4x) \\
&\quad + \mathbb{P}(-4x - |Q^* - Q_J^*| \leq Q^* \leq 4x + |Q^* - Q_J^*|, |Q^* - Q_J^*| > 4x) \\
&\leq \mathbb{P}(-8x \leq Q^* \leq 8x) + \mathbb{P}(|Q^* - Q_J^*| > 4x) \\
&\leq C(8x)^\alpha + \epsilon(J)\Gamma(4x),
\end{aligned}$$

where the last step is obtained owing to Equation (D3) and Assumptions 2. Since $\Gamma(\cdot)$ is bounded, the proof is complete. □

In the following, we demonstrate the construction of the fully connected neural networks.

LEMMA **D.19.** *For any $m > 0$, there exists a network*

$$Mult_m \in \mathcal{F}(2m + 2, (2, 12, 12, \ldots, 12, 1), 1),$$

such that for all $x, y \in [0, 1]$

$$|Mult_m(x, y) - xy| \leq 4^{-m+1}, \tag{D4}$$

where $Mult_m(x, y) \in [0, 1]$.

The next lemma designs a network which can approximate all monomials up to a certain degree. Denote $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_r)$ and $\boldsymbol{x}^{\boldsymbol{\alpha}} = x_1^{\alpha_1} \ldots x_r^{\alpha_r}$. Let $(\boldsymbol{x}^{\boldsymbol{\alpha}})_{|\alpha|<\beta}$ be the vector of all monomials up to degree $\lceil\beta\rceil - 1$, where $\beta > 0$, $|\boldsymbol{\alpha}| = \alpha_1 + \ldots + \alpha_r$. Let $C_{r,\beta}$ be the length of $(\boldsymbol{x}^{\boldsymbol{\alpha}})_{|\boldsymbol{\alpha}|<\beta}$. Note that $C_{r,\beta} \leq (\beta + 1)^r$.

LEMMA **D.20.** *For $\beta > 0$ and any positive integer m and r, there exists a network $Mon^r_{m,\beta} : [0,1]^r \to [0,1]^{C_{r,\beta}}$, such that*

$$Mon^r_{m,\beta} \in \mathcal{F}\left((m+5)\lceil \log_2(\beta \vee 1)\rceil, (r, 24\lceil\beta\rceil C_{r,\beta}, \ldots, 24\lceil\beta\rceil C_{r,\beta}, C_{r,\beta}), 1\right),$$

and

$$|Mon^r_{m,\beta}(\boldsymbol{x}) - (\boldsymbol{x}^{\boldsymbol{\alpha}})_{|\boldsymbol{\alpha}|<\beta}| \leq |\boldsymbol{\alpha}|^2 4^{-m+1},$$

for all $\boldsymbol{x} \in [0,1]^r$.

Proof. We first investigate the sub-network for any fixed degree $|\boldsymbol{\alpha}|$. When $|\boldsymbol{\alpha}| \leq 1$, it is trivial since the monomials are either linear functions or constants, and we can obtain the network $\mathcal{F}(1,(1,1,1),1)$.

When $|\boldsymbol{\alpha}| \geq 2$, we design the network $Mult^{|\boldsymbol{\alpha}|}_m$ by the following.

1. We construct the first hidden layer as

$$(x_1, x_2, \ldots, x_r) \mapsto (\underbrace{x_1, \ldots, x_1}_{\alpha_1}, \underbrace{x_2, \ldots, x_2}_{\alpha_2}, \ldots, \underbrace{x_r, \ldots, x_r}_{\alpha_r}),$$

   This layer requires $|\boldsymbol{\alpha}|$ number of neurons, since all inputs are non-negative.

2. From the second hidden layer, we apply the network described in Lemma D.19 for each adjacent pair of the first hidden layer, which computes

$$(x_\ell, x_{\ell'}) \mapsto Mult_m(x_\ell, x_{\ell'}), 0 \leq \ell, \ell' \leq |\boldsymbol{\alpha}|.$$

   After $2m+2$ layers, with each layer at most $12\lceil|\boldsymbol{\alpha}|/2\rceil$ number of nodes, we have the the $(2m+3)$-th hidden layer with nodes

$$(Mult_m(x_1, x_\ell), \ldots, Mult_m(x_{\ell'}, x_r)), \ell \in \{1,2\}, \ell' \in \{r-1, r\}.$$

3. Repeat Step 2, until the number of output is one. Denote this resulting network as $Mult^{|\boldsymbol{\alpha}|}_m$. The number of repitition is $\lceil \log_2 |\boldsymbol{\alpha}|\rceil$.

By dichotomy, the total number of hidden layers of $Mult_m^{|\boldsymbol{\alpha}|}$ is $(m+5)\lceil \log_2 |\boldsymbol{\alpha}| \rceil$, with width vector at most $(r, 24|\boldsymbol{\alpha}|, 24|\boldsymbol{\alpha}|, \ldots, 24|\boldsymbol{\alpha}|, 1)$.

The approximation error can be obtained by mathematical deduction. For any $y_1, y_2 \in \{Mult_m(\cdot,\cdot) : Mult_m$*as described in Lemma D.19*$\}$ and $z_1, z_2 \in [0,1]$, we have

$$\begin{aligned} |Mult_m(y_1,y_2) - z_1 z_2| &\leq |Mult_m(y_1,y_2) - y_1 y_2| + |y_1 - z_1| + |y_2 - z_2| \\ &\leq 4^{-m+1} + (\underbrace{4^{-m+1} + \ldots}_{\textit{three terms}}) + (\underbrace{4^{-m+1} + \ldots}_{\textit{three terms}}) \\ &\cdots \\ &\leq \left(2^{\lceil \log_2 |\boldsymbol{\alpha}| \rceil} - 1\right) 4^{-m+1}. \end{aligned}$$

Let $q = 2^{\lceil \log_2 |\boldsymbol{\alpha}| \rceil}$, then $\log_2(q-1) \leq \log_2 |\boldsymbol{\alpha}| \leq \log_2 q$, which leads to

$$\lceil \log_2 |\boldsymbol{\alpha}| \rceil \leq \log_2(q-1) + \log_2 |\boldsymbol{\alpha}| \leq \log_2 |\boldsymbol{\alpha}|^2,$$

where $|\boldsymbol{\alpha}| \geq 2$. Therefore, we have $|Mult_m^{|\boldsymbol{\alpha}|}(\boldsymbol{x}) - \boldsymbol{x}^{\boldsymbol{\alpha}}| \leq |\boldsymbol{\alpha}|^2 4^{-m+1}$.

Next, we stack all the parallel networks for all $|\alpha|$, and embed these networks into one. Denote this network as $Mon_{m,\beta}^r$, then

$$|Mon_{m,\beta}^r(\boldsymbol{x}) - (\boldsymbol{x}^{\boldsymbol{\alpha}})_{|\alpha|<\beta}|_\infty \leq \sup_{|\alpha|<\beta} |\boldsymbol{\alpha}|^2 4^{-m+1} \leq \beta^2 4^{-m+1}.$$

The number of hidden layer is taken to be $\sup_{|\alpha|<\beta}(m+5)\lceil \log_2 |\boldsymbol{\alpha}| \rceil = (m+5)\lceil \log_2(\beta \vee 1) \rceil$, the width vector is taken accordingly as

$$(r, 24\lceil\beta\rceil C_{r,\beta}, 24\lceil\beta\rceil C_{r,\beta}, \ldots, 24\lceil\beta\rceil C_{r,\beta}, C_{r,\beta}),$$

and all weights are bounded by one. □

Define the polynomial approximation of $f(x)$ at $\boldsymbol{x}_0$ up to β as

$$P_{\boldsymbol{x}_0}^\beta f(\boldsymbol{x}) = \sum_{0 \leq |\boldsymbol{\alpha}| < \beta} (\partial^{\boldsymbol{\alpha}} f)(\boldsymbol{x}_0) \frac{(\boldsymbol{x} - \boldsymbol{x}_0)^{\boldsymbol{\alpha}}}{\boldsymbol{\alpha}!}, \tag{D5}$$

where $\boldsymbol{x}_0 \in [0,1]^r$. Let $\boldsymbol{D}(M) = \{\boldsymbol{a}_{\boldsymbol{\ell}} = (\ell_k/M)_{k=1,\ldots,r}\}$ be the set of evenly spaced $M+1$ grid points. Define

$$P^\beta f(x) = \sum_{\boldsymbol{a}_{\boldsymbol{\ell}} \in \boldsymbol{D}(M)} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ P^\beta_{a_k} f(x), \tag{D6}$$

where $\sum_{\boldsymbol{a}_{\boldsymbol{\ell}} \in \boldsymbol{D}(M)} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ = 1$.

The following lemma provides the error bound for a function and its taylor approximation on $M+1$ grid points.

LEMMA **D.21.** *If $f \in \mathcal{C}^\beta([0,1]^r, K)$, then $\|P^\beta f - f\|_\infty \leq K M^{-\beta}$.*

Proof. For any $\boldsymbol{x} \in [0,1]^r$, we have

$$\begin{aligned}
& |P^\beta f(\boldsymbol{x}) - f(\boldsymbol{x})| \\
= \; & |\sum_{\boldsymbol{a}_{\boldsymbol{\ell}} \in \boldsymbol{D}(M)} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ (P^\beta_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) - f(\boldsymbol{x}))| \\
\leq \; & \sum_{\boldsymbol{a}_{\boldsymbol{\ell}} \in \boldsymbol{D}(M)} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ |P^\beta_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) - f(\boldsymbol{x})| \\
= \; & \sum_{\|\boldsymbol{x} - \boldsymbol{a}_\ell\|_\infty < M^{-1}} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ |P^\beta_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) - f(\boldsymbol{x})| \\
\leq \; & \sum_{\|\boldsymbol{x} - \boldsymbol{a}_\ell\|_\infty < M^{-1}} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ (K \|\boldsymbol{x} - \boldsymbol{a}_{\boldsymbol{\ell}}\|^\beta_\infty) \\
< \; & K M^{-\beta} \sum_{\|\boldsymbol{x} - \boldsymbol{a}_\ell\|_\infty < M^{-1}} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ \\
= \; & K M^{-\beta},
\end{aligned} \tag{D7}$$

where we use the fact that

$$\begin{aligned}
|P^\beta_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) - f(\boldsymbol{x})| \; & \leq \; \sum_{\beta - 1 \leq |\boldsymbol{\alpha}| < \beta} \frac{|(\boldsymbol{x} - \boldsymbol{a}_{\boldsymbol{\ell}})^{\boldsymbol{\alpha}}|}{\boldsymbol{\alpha}!} |(\partial^{\boldsymbol{\alpha}} f)(\boldsymbol{a}_{\boldsymbol{\ell}} + \zeta(\boldsymbol{x} - \boldsymbol{a}_{\boldsymbol{\ell}})) - (\partial^{\boldsymbol{\alpha}} f)(\boldsymbol{a}_{\boldsymbol{\ell}})| \\
& \leq \; K \|\boldsymbol{x} - \boldsymbol{a}_{\boldsymbol{\ell}}\|^\beta_\infty
\end{aligned}$$

for some $\zeta > 0$. Since (D7) holds for all $\boldsymbol{x} \in [0,1]^r$, the proof is complete. □

The following lemma provides the approximation of $\left(P^{\beta}_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x})\right)_{\boldsymbol{a}_{\boldsymbol{\ell}} \in \boldsymbol{D}(M)}$.

LEMMA **D.22.** *For $\beta > 0$ and any positive integers m, M and r, there exists a network $u_1 : [0,1]^r \to [-Ke^r, Ke^r]^{(M+1)^r}$, such that*

$$u_1 \in \mathcal{F}\left((m+5)\lceil \log_2(\beta \vee 1)\rceil, (r, 24\lceil\beta\rceil C_{r,\beta}, \ldots, 24\lceil\beta\rceil C_{r,\beta}, 2C_{r,\beta}), 2Ke^r\right),$$

and

$$\left\| u_1(\boldsymbol{x}) - \left(P^{\beta}_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x})\right)_{\boldsymbol{a}_{\boldsymbol{\ell}} \in \boldsymbol{D}(M)} \right\|_{\infty} \leq 2Ke^r \beta^2 4^{-m+1},$$

for all $\boldsymbol{x} \in [0,1]^r$.

Proof. We first write $P^{\beta}_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x})$ as the standard polynomial form. By Equation (D5), we have

$$P^{\beta}_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) = \sum_{0 \leq |\boldsymbol{\alpha}| < \beta} b(\boldsymbol{\alpha}) \boldsymbol{x}^{\alpha},$$

where the coefficients satisfy $|b(\boldsymbol{\alpha})| \leq K/\boldsymbol{\alpha}!$ and $\sum_{0 \leq |\boldsymbol{\alpha}| < \beta} |b(\boldsymbol{\alpha})| \leq Ke^r$ [92]. Therefore, $P^{\beta}_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) \in [-Ke^r, Ke^r]$, and $\frac{1}{2Ke^r} P^{\beta}_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) + \frac{1}{2} \in [0,1]$. According to Lemma D.20, there exists a network

$$u_1' \in \mathcal{F}\left((m+5)\lceil \log_2(\beta \vee 1)\rceil, (r, 24\lceil\beta\rceil C_{r,\beta}, \ldots, 24\lceil\beta\rceil C_{r,\beta}, C_{r,\beta}), 1\right),$$

such that $\left\| u_1' - \left(\frac{1}{2Ke^r} P^{\beta}_{\boldsymbol{a}_{\boldsymbol{\ell}}} f(\boldsymbol{x}) + \frac{1}{2}\right)_{\boldsymbol{a}_{\boldsymbol{\ell}} \in \boldsymbol{D}(M)} \right\|_{\infty} \leq \beta^2 4^{-m+1}$. We can construct $u_1 = 2Ke^r u_1' - Ke^r$ by setting the maximal weights as Ke^r and add one more shift term for the last hidden layer. Then, it follows that

$$u_1 \in \mathcal{F}\left((m+5)\lceil \log_2(\beta \vee 1)\rceil, (r, 24\lceil\beta\rceil C_{r,\beta}, \ldots, 24\lceil\beta\rceil C_{r,\beta}, 2C_{r,\beta}), 2Ke^r\right).$$

□

The following lemma provides the approximation of $\prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+$ by neural networks.

LEMMA **D.23.** *For any positive integers m, M and r, there exists a network $Prod_m^r : [0,1]^r \to [0,1]$, such that*

$$Prod_m^r \in \mathcal{F}\left(2 + (m+5)\lceil \log_2 r \rceil, (r, 24r, 24r, \ldots, 24r, 1), M\right),$$

and

$$\left| Prod_m^r(\boldsymbol{x}) - \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ \right| \leq r^2 4^{-m+1},$$

for all $\boldsymbol{x} \in [0,1]^r$.

Proof. We first construct the network for individual k-th term $(1 - M|x_k - \ell_k/M|)_+$. In fact, for any integers $M > 0$ and non-negative integer $k \leq M$, there exists a network $s_k \in \mathcal{F}(3, (1,2,2,1), M)$, such that

$$s_k(x) = (1 - M|x_k - \ell_k/M|)_+ \text{ for all } x \in [0,1].$$

To see this, we let the first hidden layer computes $M(x_k - \ell_k/M)$ and $M(\ell_k/M - x_k)$, the second layer computes $1 - (M(x_k - \ell_k/M))_+$ and $-(M(\ell_k/M - x_k))_+$, and the output is $(1 - (M(x_k - \ell_k/M))_+ - (M(\ell_k/M - x_k))_+)_+$.

Next, we use the same procedures discussed in the proof of Lemma D.20 and let s_k be the inputs, then build the network by Lemma D.19. Since $\alpha_1 = \ldots = \alpha_r = 1$, the total number of hidden layers of the network $Mult_m^r(\boldsymbol{x})$ is $(m+5)\lceil \log_2 r \rceil$, with width vector at most $(r, 24r, 24r, \ldots, 24r, 1)$. The error bound is $|Mult_m^r(\boldsymbol{s}) - \boldsymbol{s}^{\boldsymbol{\alpha}}| \leq r^2 4^{-m+1}$, where $\boldsymbol{s} = (s_1, \ldots, s_r)$. We combine the networks and the proof is complete. □

Let $B_{\beta,r} = ((\beta+1)^r \lceil \beta \rceil + (M+1)^r r)$. The next Lemma gives the full approximation of $P^\beta f$.

LEMMA **D.24.** *For any positive integers m, M and r, there exists a network $\widetilde{f} : [0,1]^r \to [-Ke^r, Ke^r]$, such that*

$$\widetilde{f} \in \mathcal{F}\left(2(m+5)(1 + \lceil \log_2 \beta \vee r \rceil), (r, 24B_{\beta,r}, \ldots, 24B_{\beta,r}, 1), Ke^r \vee M\right),$$

and

$$|\widetilde{f}(\boldsymbol{x}) - P^{\beta} f| \leq (2K+1)(2e)^r(1+\beta^2+r^2)4^{-m+1},$$

for all $\boldsymbol{x} \in [0,1]^r$.

Proof. By Lemma D.22, we have network u_1, with $(M+1)^r$ number of output entries, and denote the ℓ-th output entry as Q_ℓ. By Lemma D.23, we have $Prod_m^r$ for each ℓ_k/M, in total $(M+1)^r$ number of such networks for all points in $\boldsymbol{D}(M)$, and denote the ℓ-th output entry as $Prod_\ell$. Then we pair Q_ℓ and $Prod_\ell$ for all $\ell = 1, \ldots, (M+1)^r$, and apply $Mult(\cdot,\cdot)$ as discussed in Lemma D.19. Finally we add all $Mult(Q_\ell, Prod_\ell)$ and get the one entry output. The approximation error is given by

$$\begin{aligned}
& \left| \sum_{\boldsymbol{a_\ell} \in \boldsymbol{D}(M)} Mult(Q_\ell, Prod_\ell) - P^{\beta} f \right| \\
= & \left| \sum_{\boldsymbol{a_\ell} \in \boldsymbol{D}(M)} Mult(Q_\ell, Prod_\ell) - \sum_{\boldsymbol{a_\ell} \in \boldsymbol{D}(M)} \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ P^{\beta}_{a_k} f(x) \right| \\
\leq & \sum_{\boldsymbol{a_\ell} \in \boldsymbol{D}(M)} \left| Mult(Q_\ell, Prod_\ell) - \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ P^{\beta}_{a_k} f(x) \right| \\
\leq & \sum_{\boldsymbol{a_\ell} \in \boldsymbol{D}(M)} |Mult(Q_\ell, Prod_\ell) - Q_\ell Prod_\ell| + \sum_{\boldsymbol{a_\ell} \in \boldsymbol{D}(M)} \left| Q_\ell Prod_\ell - Prod_\ell P^{\beta}_{a_k} f(x) \right| \\
& + \sum_{\boldsymbol{a_\ell} \in \boldsymbol{D}(M)} \left| Prod_\ell P^{\beta}_{a_k} f(x) - \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ P^{\beta}_{a_k} f(x) \right| \\
\leq & (M+1)^r 4^{-m+1} + (M+1)^r \sup_{\ell} \left| Q_\ell - P^{\beta}_{a_k} f(x) \right| \\
& + \|P^{\beta}_{a_k} f(x)\|_\infty (M+1)^r \sup_{\ell} \left| Prod_\ell - \prod_{k=1}^{r} (1 - M|x_k - \ell_k/M|)_+ \right| \\
\leq & (M+1)^r \left(4^{-m+1} + 2Ke^r \beta^2 4^{-m+1} + Ke^r r^2 4^{-m+1} \right) \\
\leq & (2K+1)(2e)^r (1+\beta^2+r^2) 4^{-m+1}.
\end{aligned}$$

Totally, the number of hidden layers should be no more than

$$\{2 + (m+5)\lceil \log_2 r \rceil\} \vee \{(m+5)\lceil \log_2 (\beta \vee 1) \rceil\} + (2m+2) + 1,$$

and the width vector is no wider than

$$(r, 24\left((\beta+1)^r\lceil\beta\rceil + (M+1)^r r\right), \ldots, 24\left((\beta+1)^r\lceil\beta\rceil + (M+1)^r r\right), 1),$$

where we use the fact that $C_{r,\beta} \leq (\beta+1)^r$. Since all weights are bounded by $2Ke^r \vee M$, the proof is complete. □

The following lemma provides the approximation of Hölder functions in $\mathcal{C}_r^\beta([a,b]^r, K)$ via networks.

LEMMA **D.25.** *For any $g \in \mathcal{C}^\beta([a,b]^r, K)$, any positive integers m and M, there exists a network such that*

$$\widetilde{g} \in \mathcal{F}\left(2(m+5)(1+\lceil\log_2\beta\vee r\rceil), (r, 24B_{\beta,r}, \ldots, 24B_{\beta,r}, 1), K^* e^r \vee M\right),$$

and

$$|\widetilde{g}(\boldsymbol{x}) - g(\boldsymbol{x})| \leq (2K^*+1)(2e)^r(1+\beta^2+r^2)4^{-m+1} + K^* M^{-\beta},$$

for all $\boldsymbol{x} \in [a,b]^r$, where $K^* = K\left\{(b-a)^{\lfloor\beta\rfloor} \vee 1\right\}$.

Proof. For any $\boldsymbol{x} \in [a,b]^r$, define $g(\boldsymbol{x}) = h\left(\frac{\boldsymbol{x}-a}{b-a}\right)$. Since $g \in \mathcal{C}^\beta([a,b]^r, K)$, it's easy to verify that $h \in \mathcal{C}^\beta([0,1]^r, K(b-a)^{\lfloor\beta\rfloor})$. According to Lemma D.21 and Lemma D.24, there exists a network $\widetilde{h}$, which belongs to

$$\mathcal{F}\left(2(m+5)(1+\lceil\log_2\beta\vee r\rceil), (r, 24B_{\beta,r}, \ldots, 24B_{\beta,r}, 1), K(b-a)^{\lfloor\beta\rfloor} e^r \vee M\right),$$

and

$$|\widetilde{h}(\boldsymbol{t}) - h(\boldsymbol{t})| \leq (2K(b-a)^{\lfloor\beta\rfloor}+1)(2e)^r(1+\beta^2+r^2)4^{-m+1} + K(b-a)^{\lfloor\beta\rfloor} M^{-\beta},$$

for all $\boldsymbol{t} \in [0,1]^r$. When $\widetilde{g}(\boldsymbol{x}) = \widetilde{h}\left(\frac{\boldsymbol{x}-a}{b-a}\right)$, the proof is complete. □

The next lemma provides the error bound of approximating $f = \boldsymbol{g}_q \circ \ldots \circ \boldsymbol{g}_0$ by $\widetilde{\boldsymbol{g}}_q \circ \ldots \circ \widetilde{\boldsymbol{g}}_0$.

LEMMA **D.26.** *For any $\boldsymbol{g}_u = (g_{uv})_{v=1,\ldots,d_{u+1}}$, $u = 1, \ldots, q$, such that $g_{uv} \in \mathcal{C}^{\beta_u}([a_u, b_u]^{t_u}, K_u)$ for all v, $f = \boldsymbol{g}_q \circ \ldots \circ \boldsymbol{g}_0$ satisfies*

$$\|f - \widetilde{\boldsymbol{g}}_q \circ \ldots \circ \widetilde{\boldsymbol{g}}_0\|_{L^\infty[a_0,b_0]^{d_0}} \leq \prod_{u=1}^{q}(K_u \vee 1) \sum_{u=0}^{q} \| |\boldsymbol{g}_u - \widetilde{\boldsymbol{g}}_u|_\infty \|_{L^\infty[a_u,b_u]^{d_u}}^{\prod_{w=u+1}^{q} \beta_w \wedge 1},$$

where $\widetilde{\boldsymbol{g}}_u$ are the estimates of $\boldsymbol{g}_u$.

Proof. We use mathematical deduction to prove the result.

When $q = 1$, we have

$$\begin{aligned}
& |\boldsymbol{g}_1 \circ \boldsymbol{g}_0(\boldsymbol{x}) - \widetilde{\boldsymbol{g}}_1 \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})| \\
\leq\ & |\boldsymbol{g}_1 \circ \boldsymbol{g}_0(\boldsymbol{x}) - \boldsymbol{g}_1 \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})| + |\boldsymbol{g}_1 \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x}) - \widetilde{\boldsymbol{g}}_1 \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})| \\
\leq\ & K_1 |\boldsymbol{g}_0(\boldsymbol{x}) - \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})|_\infty^{\beta_1 \wedge 1} + \| |\boldsymbol{g}_1 - \widetilde{\boldsymbol{g}}_1|_\infty \|_{L^\infty[a_1,b_1]^{d_1}} \\
\leq\ & K_1 \| |\boldsymbol{g}_0 - \widetilde{\boldsymbol{g}}_0|_\infty \|_{L^\infty[a_0,b_0]^{d_0}}^{\beta_1 \wedge 1} + \| |\boldsymbol{g}_1 - \widetilde{\boldsymbol{g}}_1|_\infty \|_{L^\infty[a_1,b_1]^{d_1}} \\
\leq\ & (K_1 \vee 1) \left\{ \| |\boldsymbol{g}_0 - \widetilde{\boldsymbol{g}}_0|_\infty \|_{L^\infty[a_0,b_0]^{d_0}}^{\beta_1 \wedge 1} + \| |\boldsymbol{g}_1 - \widetilde{\boldsymbol{g}}_1|_\infty \|_{L^\infty[a_1,b_1]^{d_1}} \right\} \\
=\ & \prod_{u=1}^{q}(K_u \vee 1) \sum_{u=0}^{q} \| |\boldsymbol{g}_u - \widetilde{\boldsymbol{g}}_u|_\infty \|_{L^\infty[a_u,b_u]^{d_u}}^{\prod_{w=u+1}^{q} \beta_w \wedge 1}.
\end{aligned}$$

If for $q = q^*$, such that

$$\begin{aligned}
& |\boldsymbol{g}_{q^*}(\boldsymbol{x}) \circ \ldots \circ \boldsymbol{g}_0(\boldsymbol{x}) - \widetilde{\boldsymbol{g}}_{q^*}(\boldsymbol{x}) \circ \ldots \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})| \\
\leq\ & \prod_{u=1}^{q^*}(K_u \vee 1) \sum_{u=0}^{q^*} \| |\boldsymbol{g}_u - \widetilde{\boldsymbol{g}}_u|_\infty \|_{L^\infty[a_u,b_u]^{d_u}}^{\prod_{w=u+1}^{q^*} \beta_w \wedge 1},
\end{aligned}$$

then for $q = q^* + 1$, we have

$$
\begin{aligned}
& |\boldsymbol{g}_{q^*+1}(\boldsymbol{x}) \circ \ldots \circ \boldsymbol{g}_0(\boldsymbol{x}) - \widetilde{\boldsymbol{g}}_{q^*+1}(\boldsymbol{x}) \circ \ldots \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})| \\
\leq\ & |\boldsymbol{g}_{q^*+1}(\boldsymbol{x}) \circ \boldsymbol{g}_{q^*}(\boldsymbol{x}) \ldots \circ \boldsymbol{g}_0(\boldsymbol{x}) - \boldsymbol{g}_{q^*+1}(\boldsymbol{x}) \circ \widetilde{\boldsymbol{g}}_{q^*}(\boldsymbol{x}) \ldots \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})| \\
& + |\boldsymbol{g}_{q^*+1}(\boldsymbol{x}) \circ \widetilde{\boldsymbol{g}}_{q^*}(\boldsymbol{x}) \ldots \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x}) - \widetilde{\boldsymbol{g}}_{q^*+1}(\boldsymbol{x}) \circ \widetilde{\boldsymbol{g}}_{q^*}(\boldsymbol{x}) \ldots \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})| \\
\leq\ & K_{q^*+1} |\boldsymbol{g}_{q^*}(\boldsymbol{x}) \circ \ldots \circ \boldsymbol{g}_0(\boldsymbol{x}) - \widetilde{\boldsymbol{g}}_{q^*}(\boldsymbol{x}) \circ \ldots \circ \widetilde{\boldsymbol{g}}_0(\boldsymbol{x})|^{\beta_{q^*+1} \wedge 1} \\
& + \| |\boldsymbol{g}_{q^*+1} - \widetilde{\boldsymbol{g}}_{q^*+1}|_\infty \|_{L^\infty\left[a_{q^*+1}, b_{q^*+1}\right]^{d_{q^*+1}}} \\
\leq\ & \prod_{u=1}^{q^*+1} (K_u \vee 1) \left[\sum_{u=0}^{q^*} \| |\boldsymbol{g}_u - \widetilde{\boldsymbol{g}}_u|_\infty \|_{L^\infty[a_u, b_u]^{d_u}}^{\prod_{w=u+1}^{q^*} \beta_w \wedge 1} \right]^{\beta_{q^*+1} \wedge 1} \\
& + \| |\boldsymbol{g}_{q^*+1} - \widetilde{\boldsymbol{g}}_{q^*+1}|_\infty \|_{L^\infty\left[a_{q^*+1}, b_{q^*+1}\right]^{d_{q^*+1}}} \\
\leq\ & \prod_{u=1}^{q^*+1} (K_u \vee 1) \sum_{u=0}^{q^*} \| |\boldsymbol{g}_u - \widetilde{\boldsymbol{g}}_u|_\infty \|_{L^\infty[a_u, b_u]^{d_u}}^{\prod_{w=u+1}^{q^*+1} \beta_w \wedge 1} \\
& + \| |\boldsymbol{g}_{q^*+1} - \widetilde{\boldsymbol{g}}_{q^*+1}|_\infty \|_{L^\infty\left[a_{q^*+1}, b_{q^*+1}\right]^{d_{q^*+1}}} \\
=\ & \prod_{u=1}^{q^*+1} (K_u \vee 1) \sum_{u=0}^{q^*+1} \| |\boldsymbol{g}_u - \widetilde{\boldsymbol{g}}_u|_\infty \|_{L^\infty[a_u, b_u]^{d_u}}^{\prod_{w=u+1}^{q^*+1} \beta_w \wedge 1}.
\end{aligned}
$$

The proof is complete. □

For any $u = 0, \ldots, q$, define $\beta_u^* = \beta_u \prod_{w=u+1}^{q} \beta_w \wedge 1$.

In the following, we will consider the local properties of the aforementioned function class $\mathcal{G}$. With a little abuse of notation, define the local version function class $\mathcal{G}(q, J, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, \boldsymbol{K})$: let $\mathcal{G}(q, J, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, \boldsymbol{K})$ be the class of functions of the form

$$
g(\boldsymbol{x}) = g_q \circ \cdots \circ g_0(\boldsymbol{x}),\ \forall \boldsymbol{x} \in [a_0, b_0]^{d_0}, \tag{D8}
$$

where for any constants $a_0, b_0 \in \mathbb{R}$, there exists $a_1, \ldots, a_{q+1}$, $b_1, \ldots, b_{q+1}$, $K_0, \ldots, K_q$, such that $g_u = (g_{u1}, \ldots, g_{ud_{u+1}}) : [a_u, b_u]^{d_u} \to [a_{u+1}, b_{u+1}]^{d_{u+1}}$, with $g_{uv} \in \mathcal{C}_{t_u}^{\beta_u}\left([a_u, b_u]^{t_u}, K_u\right)$ being β_u-Hölder functions of radius K_u involving only $t_u (\leq d_u)$ variables. Define the set of the effective inputs: $\mathcal{A} = \{j : \xi_j \text{ is effective for } g_{0v} \text{ for all } v = 1, \ldots, d_1\}$, such that $|\mathcal{A}| \leq t_0 d_1 < \infty$. Owing to finite $|\mathcal{A}|$, it is straightforward to verify the following lemma:

The following lemma provides the network approximation of $\mathcal{G}\left(q, J, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, K\right)$.

LEMMA **D.27.** *For any $f \in \mathcal{G}(q, J, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, K)$ and any sufficiently large positive integer M, there exists a fully connected network $\widetilde{f} \in \mathcal{F}(L(M), \boldsymbol{p}(M), B(M))$ satisfying $L(M) \lesssim \log_2 M$, $\|\boldsymbol{p}(M)\|_\infty \lesssim \max_{u=0,\ldots,q} d_{u+1} t_u (M+1)^{t_u}$, $B(M) \lesssim M$, such that*

$$\|\widetilde{f}(\boldsymbol{x}) - f(\boldsymbol{x})\|_\infty \leq C^* \max_u M^{-\beta_u^*},$$

where C^* is some constant depending on function class $\mathcal{G}(q, \boldsymbol{d}, \boldsymbol{t}, \boldsymbol{\beta}, K)$.

Proof. We first apply Lemma D.25 to each g_{uv}. Let $m_u = \lceil \frac{\beta_u}{2} \log_2(C_u^{-1/\beta_u} M) \rceil$, where $C_u = \frac{K_u^*}{8(K_u^*+1)(2e)^{t_u}(1+\beta_u+t_u^2)}$, then for any positive integer M, there exists a network such that

$$\widetilde{g}_u \in \mathcal{F}\left(2(m_u+5)(1+\lceil \log_2 \beta_u \vee t_u \rceil), (t_u, 24B_{\beta_u,t_u}, \ldots, 24B_{\beta_u,t_u}, 1), K_u^* e^{t_u} \vee M\right),$$

and

$$|\widetilde{g}_u(\boldsymbol{x}) - g_u(\boldsymbol{x})| \leq K_u^* M^{-\beta_u},$$

for all $\boldsymbol{x} \in [a_u, b_u]^{t_u}$, where $K_u^* = K_u \left\{(b_u - a_u)^{\lfloor \beta_u \rfloor} \vee 1\right\}$. We add two more layers to calculate $\widetilde{g}_u^* = ((b_{u+1} - a_{u+1}) - (b_{u+1} - \widetilde{g}_u)_+)_+ + a_{u+1}$, which confines the range of outputs. Therefore, we have d_{u+1} parallel networks for any fixed u, which can be embedded in

$$\mathcal{F}\left(L_u, (d_u, 24d_{u+1}B_{\beta_u,t_u}, \ldots, 24d_{u+1}B_{\beta_u,t_u}, 1), K_u^{**} e^{t_u} \vee M\right),$$

where $L_u = 2(m_u+5)(1+\lceil \log_2 \beta_u \vee t_u \rceil) + 2$, $K_u^{**} = (K_u \vee K_{u+1}) \left\{(b_u - a_u)^{\lfloor \beta_u \rfloor} \vee 1\right\}$. The described network $\widetilde{\boldsymbol{g}}_u^*$ satisfies

$$\||\widetilde{\boldsymbol{g}}_u^* - \boldsymbol{g}_u|_\infty\|_{L^\infty[a_u,b_u]^{d_u}} \leq \||\widetilde{\boldsymbol{g}}_u - \boldsymbol{g}_u|_\infty\|_{L^\infty[a_u,b_u]^{d_u}} \leq K_u^* M^{-\beta_u}.$$

Lastly, we can design $\widetilde{f} = \widetilde{\boldsymbol{g}}_q \circ \widetilde{\boldsymbol{g}}_{q-1}^* \ldots \circ \widetilde{\boldsymbol{g}}_0^*$. This network can be embedded in $\mathcal{F}(L, \boldsymbol{p}, B)$, where $L = q - 1 + \sum_{u=0}^q L_u$, $\boldsymbol{p} = (d_0, 24B_{\beta,t,d}, \ldots, 24B_{\beta,t,d}, 1)$, with $B_{\beta,t,d} = \max_u d_{u+1} B_{\beta_u,t_u}$, and $B = \max_u K^{**} e^{t_u} \vee M$. By choosing sufficiently large M, we have $L \lesssim \log_2 M$, $\|\boldsymbol{p}\|_\infty \lesssim \max_u d_{u+1} t_u (M+1)^{t_u}$ and $B \lesssim M$.

By applying Lemma D.26, we have

$$\begin{aligned}\|f-\widetilde{f}\|_{L^\infty[a_0,b_0]^{d_0}} &\leq \prod_{u=1}^{q}(K_u\vee 1)\sum_{u=0}^{q}(K_u^*M^{-\beta_u})^{\prod_{w=u+1}^{q}\beta_w\wedge 1}\\ &\leq (\max_u K_u)^{2q}\sum_{u=0}^{q}M^{-\beta_u\prod_{w=u+1}^{q}\beta_w\wedge 1}\\ &\leq (q+1)(\max_u K_u)^{2q}\max_u M^{-\beta_u^*}=C^*\max_u M^{-\beta_u^*}.\end{aligned}$$

□

In the following, we provide the complete proof of Lemma D.17.

Proof. We use the network $\widetilde{\eta}$ obtained in Lemma D.27 and construct a network

$$\widetilde{f}=2\left(\sigma\left(\epsilon^{-1}\left(\widetilde{\eta}-\frac{1}{2}\right)\right)-\sigma\left(\epsilon^{-1}\left(\widetilde{\eta}-\frac{1}{2}\right)-1\right)\right)-1.$$

We need two more layers from $\widetilde{\eta}$ to $\widetilde{f}$, which can be obtained by

$$\widetilde{\eta}\mapsto\sigma\left(\epsilon^{-1}\left(\widetilde{\eta}-\frac{1}{2}\right)\right),$$

$$\widetilde{\eta}\mapsto\sigma\left(\epsilon^{-1}\left(\widetilde{\eta}-\frac{1}{2}\right)-1\right)$$

with the maximal value of weights is bounded above by ϵ^{-1}. Since the subtraction is multiplied by two, we need double the width, and the last layer of additive structure with bias term -1. The construction doesn't change the asymptotic orders of L and $\boldsymbol{p}$. Define $\boldsymbol{A}=\left\{\boldsymbol{\xi}_J:|\eta^{(J)}(\boldsymbol{\xi}_J)-\frac{1}{2}|>2\epsilon\right\}$, then we have the identity $\widetilde{f}(\boldsymbol{\xi}_J)=f_\phi^*(\boldsymbol{\xi}_J)$ when $\boldsymbol{\xi}_J\in\boldsymbol{A}$, where $f_\phi^*=\arg\min_{f\in\mathcal{F}}E\left[\phi(Yf(\boldsymbol{\xi}_J))\right]$ for all measurable real-valued functions with J inputs. This is because when $\eta^{(J)}(\boldsymbol{\xi}_J)-\frac{1}{2}>2\epsilon$, we have

$$\widetilde{\eta}(\boldsymbol{\xi}_J)-\frac{1}{2}=\widetilde{\eta}(\boldsymbol{\xi}_J)-\eta^{(J)}(\boldsymbol{\xi}_J)+\eta^{(J)}(\boldsymbol{\xi}_J)-\frac{1}{2}>-\epsilon+2\epsilon=\epsilon$$

and when $\eta^{(J)}(\boldsymbol{\xi}_J) - \frac{1}{2} < -2\epsilon$, we have

$$\widetilde{\eta}(\boldsymbol{\xi}_J) - \frac{1}{2} = \widetilde{\eta}(\boldsymbol{\xi}_J) - \eta^{(J)}(\boldsymbol{\xi}_J) + \eta^{(J)}(\boldsymbol{\xi}_J) - \frac{1}{2} < \epsilon - 2\epsilon = -\epsilon.$$

For appropriately chosen M, ϵ can be sufficiently small around 0. Note that ϕ is Fisher consistent, i.e., $sign(f_\phi^*) = G_J^*$, and by Theorem 2.31 of [100],

$$\sup_{h\in\mathcal{H}} E\left[R_h(\widetilde{G}) - R_h(G_J^*)\right] \lesssim \sup_{h\in\mathcal{H}} E\left[\phi\left(Y\widetilde{f}\right) - \phi\left(YG_J^*\right)\right].$$

Therefore,

$$\begin{aligned}
& E\left[\phi\left(Y\widetilde{f}(\boldsymbol{\xi}_J)\right) - \phi\left(YG_J^*(\boldsymbol{\xi}_J)\right)\right] \\
= & \int \left|\widetilde{f}(\boldsymbol{x}) - G_J^*(\boldsymbol{x})\right| \left|2\eta^{(J)}(\boldsymbol{x}) - 1\right| dP_{\widetilde{\boldsymbol{\theta}}}(\boldsymbol{x}) \\
= & 2\int_{\boldsymbol{A}^c} \left|\widetilde{f}(\boldsymbol{x}) - G_J^*(\boldsymbol{x})\right| \left|\eta^{(J)}(\boldsymbol{x}) - \frac{1}{2}\right| dP_{\widetilde{\boldsymbol{\theta}}}(\boldsymbol{x}) \\
\leq & 4\int_{\boldsymbol{A}^c} \left|\eta^{(J)}(\boldsymbol{x}) - \frac{1}{2}\right| dP_{\widetilde{\boldsymbol{\theta}}}(\boldsymbol{x}) \\
\leq & 8\epsilon\mathbb{P}\left(\left|\eta^{(J)}(\boldsymbol{\xi}_J) - \frac{1}{2}\right| \leq 2\epsilon\right) \lesssim \epsilon^{\alpha+1} + \epsilon\epsilon(J),
\end{aligned}$$

and for any $J \geq J_0$, the upper bound of the EMR for the first J scores is derived. Combining the result in Lemma D.27 , the proof is complete. □

We introduce the complexity measures of a given function class. Let $\mathcal{F}$ be a given class of real valued functions on $\mathcal{C}$.

Definition 1. *(Covering number) Let $\kappa > 0$ and $\|f\|_\infty = \sup_{\boldsymbol{z}\in\mathcal{C}} |f(\boldsymbol{z})|$. A subset $\{f_k \in \mathcal{F}\}_{k\geq 1}$ is called a κ-covering set of $\mathcal{F}$ with respect to $\|f\|_\infty$, if for all $f \in \mathcal{F}$, there exists an f_k such that $\|f_k - f\|_\infty \leq \kappa$. The κ-covering number of $\mathcal{F}$ with respect to $\|f\|_\infty$ is defined by*

$$\mathcal{N}(\kappa, \mathcal{F}, \|\cdot\|_\infty) = \inf\left\{N \in \mathbb{N} : \exists f_1, \ldots, f_N, \textit{s.t. } \mathcal{F} \subset \bigcup_{k=1}^{N} \{f \in \mathcal{F} : \|f_k - f\|_\infty \leq \kappa\}\right\}.$$

Definition 2. *(Bracketing entropy) A collection of pairs* $\left\{\left(f_k^L, f_k^U\right) \in \mathcal{F} \times \mathcal{F}\right\}_{k\geq 1}$ *is called a* κ*-bracketing set of* $\mathcal{F}$ *with respect to* $\|f\|_\infty$*, if* $\|f_k^L - f_k^U\|_\infty \leq \kappa$ *and for all* $f \in \mathcal{F}$*, there exists a pair* $\left(f_k^L, f_k^U\right)$ *such that* $f_k^L \leq f \leq f_k^U$*. The cardinality of the minimal* κ*-bracketing set with respect to* $\|f\|_\infty$ *is called the* κ*-bracketing number, which is denoted by* $\mathcal{N}_B(\kappa, \mathcal{F}, \|\cdot\|_\infty)$*. Define* κ*-bracketing entropy as* $H_B(\kappa, \mathcal{F}, \|\cdot\|_\infty) = \log \mathcal{N}_B(\kappa, \mathcal{F}, \|\cdot\|_\infty)$*. Given any* $\kappa > 0$*, it is known that*

$$\log \mathcal{N}(\kappa, \mathcal{F}, \|\cdot\|_\infty) \leq H_B(\kappa, \mathcal{F}, \|\cdot\|_\infty) \leq \log \mathcal{N}(\kappa/2, \mathcal{F}, \|\cdot\|_\infty).$$

The following lemma provides the excess risk of FDNN classifier under some regularity conditions.

LEMMA **D.28.** *Under Assumption 1 and some regularity conditions:*

(i) For a positive sequence $\{\delta_n^*\}_{n\geq 1}$*, there exists a sequence of function classes* $\{\mathcal{F}_n\}_{n\geq 1}$ *such that* $E\left[\phi\left(Y f_n(\boldsymbol{x})\right) - \phi\left(Y C^*(\boldsymbol{x})\right)\right] \leq \delta_n^*$ *for some* $f_n \in \mathcal{F}_n$*;*

(ii) There exists a sequence $\{\delta_n\}_{n\geq 1}$*, such that* $\delta_n \leq n^{-\frac{\alpha+1}{\alpha+2}}$*,* $H_B^{1/2}(\delta_n, \mathcal{F}_n, \|\cdot\|_2) \leq n\delta_n^{(\alpha+2)/(\alpha+1)}$ *, and* $\{\mathcal{F}_n\}_{n\geq 1}$ *in (i),*

when $n\left(\max\{\delta_n^*, \delta_n\}\right)^{2(\alpha+2)/(\alpha+1)} \gtrsim \log_2^{1+a} n$ *for any small enough* $a > 0$*, then* $\widehat{f}_{\phi,n}$ *satisfies* $E\left[R(\widehat{G}^{FDNN}) - R(G_J^*)\right] \lesssim \left(\max\{\delta_n^*, \delta_n\}\right)^2$.

Proof. This result can be derived similarly as Theorem A.2 in [52]. Therefore, since condition (C1) can be directly verified if (C3) holds, we only need to verify the conditions (C3) in Theorem A.1. Following the same notations and (A.4) in [52], since $\epsilon_n^2 \geq 2^7\delta_n/c_1$ for some small enough ϵ_n, when $\delta_n \leq n^{-\frac{\alpha+1}{\alpha+2}}$, we have

$$H_B^{1/2}(\delta_n, \mathcal{F}_n, \|\cdot\|_2)\left(\epsilon_n^2\right)^{\frac{\alpha+2}{2(\alpha+1)}} \leq n\delta_n^{\frac{\alpha+2}{\alpha+1}}\left(\epsilon_n^2\right)^{\frac{\alpha+2}{2(\alpha+1)}} \leq n\delta_n^{\frac{\alpha+2}{2(\alpha+1)}}\left(\delta_n/\epsilon_n^2\right)^{\frac{\alpha+2}{2(\alpha+1)}},$$

which has the upper bound $(2^7/c_1)^{\frac{\alpha+2}{2(\alpha+1)}} n^{1/2}$. Therefore (C3) is verified. □

The following lemma provides the covering number for fully connected networks.

LEMMA **D.29.** *Given a fully connected network class $\mathcal{F}(L, J, \boldsymbol{p}, B)$ and any $\delta > 0$, the upper bound for $\log \mathcal{N}(\delta, \mathcal{F}(L, J, \boldsymbol{p}, B), \|\cdot\|_\infty)$ is*

$$2L\left(\sum_{l=1}^{L}(p_l+1)p_{l+1} + Jp_1 + 1\right)\log\left\{\delta^{-1}(L+1)(\max\{\|\boldsymbol{p}\|_\infty, J\}+1)(B\vee 1)\right\}.$$

Proof. Let S be the number of active neurons for the neural network. According to Lemma 3 of [102], since the fully connected network has $S \leq \sum_{l=0}^{L}(p_l+1)p_{l+1}$, the proof is complete. □

Denote $\mathcal{F}^* = \mathcal{F}(L(M), J, \boldsymbol{p}(M), B(M))$. As a result of Lemma D.29, we restrict that $J \lesssim \|\boldsymbol{p}\|_\infty$ (we show it holds under Assumption 2 in Proof of Theorem 5.1(ii)), when $\epsilon^\alpha \gtrsim \epsilon(J)$, for relatively large M, we have

$$\begin{aligned}
\log \mathcal{N}(\delta, \mathcal{F}^*, \|\cdot\|_\infty) &\lesssim \max_{u=0,\ldots,q} M^{2t_u}\left[\log^3 M + \log M \log(\delta^{-1})\right] \\
&\lesssim \left(\epsilon^{-1}\right)^{\max_u 2t_u/\beta_u^*}\left[\log^3(\epsilon^{-1}) + \log(\epsilon^{-1})\log(\delta^{-1})\right] \\
&= \left(\epsilon^{\alpha+1}\right)^{-\max_u \frac{2t_u}{(\alpha+1)\beta_u^*}}\left[\log^3(\epsilon^{-1}) + \log(\epsilon^{-1})\log(\delta^{-1})\right],
\end{aligned}$$

where the second inequality is obtained by $M = (\epsilon^{-1})^{\max_u 1/\beta_u^*}$.

D.2.1 Proof of Proposition D.1

Proof. Without loss of generality, suppose $\int_0^\infty \Gamma(x)dx = 1$, otherwise we scale it by $\int_0^\infty \Gamma(x)dx$.

$$\begin{aligned}
|E\left(Q^* - Q_J\right)| &\leq E|Q^* - Q_J| = \int_0^\infty \mathbb{P}(|Q^* - Q_J| > x)dx \\
&\lesssim \int_0^\infty \epsilon(J)\Gamma(x)dx = \epsilon(J).
\end{aligned}$$

□

D.2.2 Proof of Proposition D.2

Proof. For any $x > 0$ and $J \geq J_0$, we have

$$\begin{aligned} P\left(|Q^*| > x\right) &\leq P\left(|Q^* - Q_J^*| + |Q_J^*| > x\right) \\ &= P\left(|Q^* - Q_J^*| > x - |Q_J^*|\right) \\ &= P\left(|Q^* - Q_J^*| > x - |Q_J^*|, |Q_J^*| \leq x/2\right) \\ &\quad + P\left(|Q^* - Q_J^*| > x - |Q_J^*|, |Q_J^*| > x/2\right) \\ &\leq P\left(|Q^* - Q_J^*| > x/2\right) + P\left(|Q_J^*| > x/2\right). \end{aligned}$$

Let $x \to \infty$, according to Assumption 2, right hand side of the inequality has the limit zero for any finite J. Thus, the proof is complete. □

D.2.3 Extension to independent t distribution

The following proposition provides a sufficient condition for equivalent measures under student's t distribution. It is an extension from Gaussian functional data. Similar results for the Gaussian case have been well studied in [31, 30, 12].

Proposition D.3. *When* $\sum_{j=1}^{\infty} \log\left[2\left(1+\nu_j|\delta_j|\right)^{\nu_j+1}\right] < \infty$ *and* $\sum_{j=1}^{\infty} \nu_j^{5/2}|\delta_j| < \infty$ *are satisfied,* $\mathbb{P}\left(|Q^*(\boldsymbol{\xi})| < \infty\right) = 1$.

Proof. We first show some preliminaries. For some positive a and $r \geq 2$, we have

$$\begin{aligned} & \int_{-\infty}^{\infty} (1+x^2)^{-r} \log(1+ax^2) dx \\ = & \int_{-\infty}^{\infty} (1+x^2)^{-r} \log(1+x^2) \frac{\log(1+ax^2)}{\log(1+x^2)} dx \\ \leq & \; a \int_{-\infty}^{\infty} (1+x^2)^{-r} \log(1+x^2) dx \\ = & \; 2a \int_{-\pi/2}^{\pi/2} (\sec y)^{-r+2} \log(\sec y) dy \\ \leq & \; -4a \int_{0}^{\pi/2} \log(\cos y) dy \\ = & \; 2a\pi \log 2, \end{aligned} \tag{D9}$$

where we use the substitution $x = \tan y$ for the second equation, and $(\sec y)^{-r+2} \leq 1$ for the second inequality.

We have $h_j^{(1)}(x) = \frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}}$ and $h_j^{(1)}(x) = \frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\mu_j+1}{2}}$. According to Wallis' inequality, for any positive integers m,

$$\frac{1}{\sqrt{\pi(m+4/\pi-1)}} \leq \frac{(2m-1)!!}{(2m)!!} < \frac{1}{\sqrt{\pi(m+1/4)}}.$$

After some simple calculation, when $\nu_j = 2m$,

$$\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} = \frac{1}{2\sqrt{\nu_j}}\frac{(2m-1)!!}{(2m-2)!!} \in \frac{1}{\sqrt{2\pi}}\left[\sqrt{\frac{\nu_j-1}{\nu_j+1}}, 1\right], \tag{D10}$$

and when $\nu_j = 2m-1$,

$$\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} = \frac{1}{\pi\sqrt{\nu_j}}\frac{(2m-2)!!}{(2m-3)!!} \in \frac{1}{\sqrt{2\pi}}\left[\sqrt{\frac{\nu_j-1}{\nu_j+1}}, 1\right]. \tag{D11}$$

Thus, the ratio of coefficients of $h_j^{(1)}$ and $h_j^{(-1)}$ satisfies

$$\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1} \in \left[\sqrt{\frac{\nu_j-1}{\nu_j+1}}, \sqrt{\frac{\mu_j+1}{\mu_j-1}}\right],$$

and

$$\log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1}\right) \in \frac{1}{2}\left[-\log\left(1+\frac{2}{\nu_j-1}\right), \log\left(1+\frac{2}{\mu_j-1}\right)\right].$$

By assuming that $\mu_j, \nu_j \geq 3$, we have

$$\log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1}\right) \in [-(\log 2)/2, (\log 2)/2]. \tag{D12}$$

Without loss of generality, in the following we assume $\epsilon_j \geq 0$ for all $j \geq 1$. When $X(t) \sim$ class 1, we have

$$
\begin{aligned}
& E\left[Q^*(X, \boldsymbol{\theta})\right] \\
= & \sum_{j=1}^{\infty} \int_{-\infty}^{\infty} h_j^{(1)} \log\left(\frac{h_j^{(1)}}{h_j^{(-1)}}\right) \\
= & \sum_{j=1}^{\infty} \int_{-\infty}^{\infty} \frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}} \left\{ \log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)} \right)^{-1} \right) \right. \\
& \left. + \left(\frac{\mu_j+1}{2}\right) \log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right) \log\left(1+\frac{x^2}{\nu_j}\right) \right\} dx \\
= & \sum_{j=1}^{\infty} \log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)} \right)^{-1} \right) + \frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \\
& \times \int_{-\infty}^{\infty} \left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}} \left\{ \left(\frac{\mu_j+1}{2}\right) \log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right) \log\left(1+\frac{x^2}{\nu_j}\right) \right\} dx \\
= & \sum_{j=1}^{\infty} \log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)} \right)^{-1} \right) + \frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \\
& \times \int_{-\infty}^{\infty} \left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}} \left\{ \left(\frac{\mu_j+1}{2}\right) \log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right) \log\left(1+\frac{x^2}{\mu_j}\right) \right. \\
& \left. + \left(\frac{\nu_j+1}{2}\right) \log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right) \log\left(1+\frac{x^2}{\nu_j}\right) \right\} dx \\
= & \sum_{j=1}^{\infty} \log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)} \right)^{-1} \right) + \frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)} \\
& \times \left\{ \frac{\mu_j-\nu_j}{2} \int_{-\infty}^{\infty} \left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}} \log\left(1+\frac{x^2}{\mu_j}\right) dx \right. \\
& \left. + \frac{\nu_j+1}{2} \int_{-\infty}^{\infty} \left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}} \left[\log\left(1+\frac{x^2}{\mu_j}\right) - \log\left(1+\frac{x^2}{\nu_j}\right) \right] dx \right\}.
\end{aligned}
$$

According to inequality (D9), we have

$$
\frac{\mu_j-\nu_j}{2} \int_{-\infty}^{\infty} \left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}} \log\left(1+\frac{x^2}{\mu_j}\right) dx \in \left[-(\pi \log 2)\nu_j^{5/2}|\epsilon_j|, (\pi \log 2)\nu_j^{5/2}|\epsilon_j| \right]. \tag{D13}
$$

Since

$$\begin{aligned}\log\left(1+\frac{x^2}{\mu_j}\right)-\log\left(1+\frac{x^2}{\nu_j}\right) &= \log\left(1+\frac{\nu_j\delta_j x^2}{\nu_j+x^2}\right)\\ &\leq \begin{cases}\log\left(1+\frac{\delta_j x^2}{2\nu_j}\right), & \nu_j \leq x^2\\ \log\left(1+\frac{\nu_j\delta_j}{2}\right), & \nu_j > x^2\end{cases}\end{aligned}$$

and $\delta_j \geq 0$, we have

$$\begin{aligned}&\frac{\nu_j+1}{2}\int_{-\infty}^{\infty}\left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}}\left[\log\left(1+\frac{x^2}{\mu_j}\right)-\log\left(1+\frac{x^2}{\nu_j}\right)\right]dx\\ =\;&(\nu_j+1)\int_0^{\infty}\left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}}\left[\log\left(1+\frac{x^2}{\mu_j}\right)-\log\left(1+\frac{x^2}{\nu_j}\right)\right]dx\\ \leq\;&(\nu_j+1)\int_0^{\sqrt{\nu_j}}\left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}}\log\left(1+\frac{\delta_j x^2}{2\nu_j}\right)dx\\ &+(\nu_j+1)\int_{\sqrt{\nu_j}}^{\infty}\left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}}\log\left(1+\frac{\nu_j\delta_j}{2}\right)dx\\ \leq\;&\frac{\pi\log 2}{2}\delta_j\left(\nu_j^{3/2}+\nu_j^{1/2}\right)+(\nu_j+1)\log\left(1+\frac{\nu_j\delta_j}{2}\right)\int_0^{\infty}\left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}}dx\\ =\;&\frac{\pi\log 2}{2}\delta_j\left(\nu_j^{3/2}+\nu_j^{1/2}\right)+\left(\frac{\nu_j+1}{2}\right)\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\right)^{-1}\log\left(1+\frac{\nu_j\delta_j}{2}\right). \quad \text{(D14)}\end{aligned}$$

Therefore, according to Equations (D10) to (D14), we have

$$\begin{aligned}&\left|E\left[Q^*(X,\boldsymbol{\theta})\right]\right|\\ \leq\;&\sum_{j=1}^{\infty}\sqrt{\frac{\pi\log 2}{8}}\left(2\nu_j^{5/2}+\nu_j^{3/2}+\nu_j^{1/2}\right)\delta_j\\ &+\sum_{j=1}^{\infty}(\log 2)/2+\sum_{j=1}^{\infty}\left(\frac{\nu_j+1}{2}\right)\log\left(1+\frac{\nu_j\delta_j}{2}\right).\\ \leq\;&\left(\sqrt{2\pi\log 2}\right)\sum_{j=1}^{\infty}\nu_j^{5/2}\delta_j+\sum_{j=1}^{\infty}\left(\frac{1}{2}\right)\log\left[2\left(1+\frac{\nu_j\delta_j}{2}\right)^{\nu_j+1}\right].\end{aligned}$$

When $X(s)$'s class label is -1, we have

$$
\begin{aligned}
& E\left[Q^*(X,\boldsymbol{\theta})\right] \\
= & \sum_{j=1}^{\infty}\int_{-\infty}^{\infty} h_j^{(-1)} \log\left(\frac{h_j^{(1)}}{h_j^{(-1)}}\right)(x)dx \\
= & \sum_{j=1}^{\infty}\int_{-\infty}^{\infty} \frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\left(1+\frac{x^2}{\nu_j}\right)^{-\frac{\nu_j+1}{2}} \left\{\log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1}\right)\right. \\
& \left. + \left(\frac{\mu_j+1}{2}\right)\log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right)\log\left(1+\frac{x^2}{\nu_j}\right)\right\} dx \\
= & \sum_{j=1}^{\infty}\log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1}\right) + \frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)} \times \\
& \int_{-\infty}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\nu_j+1}{2}}\left\{\left(\frac{\mu_j+1}{2}\right)\log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right)\log\left(1+\frac{x^2}{\nu_j}\right)\right\} dx \\
= & \sum_{j=1}^{\infty}\log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1}\right) + \frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)} \\
& \times \int_{-\infty}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\nu_j+1}{2}}\left\{\left(\frac{\mu_j+1}{2}\right)\log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right)\log\left(1+\frac{x^2}{\mu_j}\right)\right. \\
& \left. + \left(\frac{\nu_j+1}{2}\right)\log\left(1+\frac{x^2}{\mu_j}\right) - \left(\frac{\nu_j+1}{2}\right)\log\left(1+\frac{x^2}{\nu_j}\right)\right\} dx \\
= & \sum_{j=1}^{\infty}\log\left(\frac{\Gamma\left(\frac{\nu_j+1}{2}\right)}{\sqrt{\nu_j\pi}\Gamma\left(\frac{\nu_j}{2}\right)}\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1}\right) + \frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)} \\
& \times \left\{\frac{\mu_j-\nu_j}{2}\int_{-\infty}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\nu_j+1}{2}}\log\left(1+\frac{x^2}{\mu_j}\right)dx\right. \\
& \left. + \frac{\nu_j+1}{2}\int_{-\infty}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\nu_j+1}{2}}\left[\log\left(1+\frac{x^2}{\mu_j}\right) - \log\left(1+\frac{x^2}{\nu_j}\right)\right]dx\right\}.
\end{aligned}
$$

According to inequality (D9) and some simple calculation, we have

$$
\frac{\mu_j-\nu_j}{2}\int_{-\infty}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\mu_j+1}{2}}\log\left(1+\frac{x^2}{\mu_j}\right)dx \in \left[-(\pi\log 2)\nu_j^{5/2}|\delta_j|, (\pi\log 2)\nu_j^{5/2}|\delta_j|\right], \tag{D15}
$$

where we use the fact that $\delta_j \nu_j \geq 0$. We also have

$$
\begin{aligned}
& \frac{\nu_j+1}{2}\int_{-\infty}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\mu_j+1}{2}}\left[\log\left(1+\frac{x^2}{\mu_j}\right)-\log\left(1+\frac{x^2}{\nu_j}\right)\right]dx \\
= \;& (\nu_j+1)\int_{0}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\mu_j+1}{2}}\left[\log\left(1+\frac{x^2}{\mu_j}\right)-\log\left(1+\frac{x^2}{\nu_j}\right)\right]dx \\
\leq \;& (\nu_j+1)\int_{0}^{\sqrt{\nu_j}}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\mu_j+1}{2}}\log\left(1+\frac{\delta_j x^2}{2\nu_j}\right)dx \\
& +(\nu_j+1)\int_{\sqrt{\nu_j}}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\mu_j+1}{2}}\log\left(1+\frac{\nu_j\delta_j}{2}\right)dx \\
\leq \;& \frac{\pi\log 2}{2}\delta_j\left(\nu_j^{3/2}+\nu_j^{1/2}\right)+(\nu_j+1)\log\left(1+\frac{\nu_j\delta_j}{2}\right)\int_{0}^{\infty}\left(1+\frac{x^2}{\mu_j}\right)^{-\frac{\mu_j+1}{2}}dx \\
= \;& \frac{\pi\log 2}{2}\delta_j\left(\nu_j^{3/2}+\nu_j^{1/2}\right)+\left(\frac{\nu_j+1}{2}\right)\left(\frac{\Gamma\left(\frac{\mu_j+1}{2}\right)}{\sqrt{\mu_j\pi}\Gamma\left(\frac{\mu_j}{2}\right)}\right)^{-1}\log\left(1+\frac{\nu_j\delta_j}{2}\right), \qquad \text{(D16)}
\end{aligned}
$$

where the last inequality uses the fact that $\delta_j \nu_j \geq 0$.

Therefore, according to Equations (D10), (D11), (D12), (D15) and (D16), we have

$$
\begin{aligned}
& |E\left[Q^*(X,\boldsymbol{\theta})\right]| \\
\leq \;& \sum_{j=1}^{\infty}\sqrt{\frac{\pi\log 2}{8}}\left(2\nu_j^{5/2}+\nu_j^{3/2}+\nu_j^{1/2}\right)\delta_j \\
& +\sum_{j=1}^{\infty}(\log 2)/2+\sum_{j=1}^{\infty}\left(\frac{\nu_j+1}{2}\right)\log\left(1+\frac{\nu_j\delta_j}{2}\right). \\
\leq \;& \sqrt{2\pi\log 2}\sum_{j=1}^{\infty}\nu_j^{5/2}\delta_j+\sum_{j=1}^{\infty}\left(\frac{1}{2}\right)\log\left\{2\left(1+\frac{\nu_j\delta_j}{2}\right)^{\nu_j+1}\right\}.
\end{aligned}
$$

Since $\nu_j\delta_j = o(1)$ and positive, we have

$$
\log\left(1+\frac{\nu_j\delta_j}{2}\right) \leq -\log\left(1-\frac{\nu_j\delta_j}{2}\right) = \log\left(1+\frac{\nu_j\delta_j}{2-\nu_j\delta_j}\right) \leq \log(1+\nu_j\delta_j),
$$

and the proof is done. □

Appendix E

Empirical Likelihood Ratio Tests for Varying Coefficient Geo Models

E.1 Regularity assumptions

Without loss of generality, let the area of Ω be 1. For the univariate splines, we consider equally-spaced knots in our theoretical derivation. For a univariate function $\psi(\cdot)$, denote $\psi'(\cdot)$, $\psi''(\cdot)$ and $\psi^{(v)}(\cdot)$ be its first, second and v-th order derivative, respectively. For any bivariate function g defined on Ω, let $\|g(\boldsymbol{s})\|_{\infty,\Omega} = \sup_{\mathbf{s}\in\Omega} |g(\boldsymbol{s})|$ be the supremum norm of g, and let $|g|_{v,\infty,\Omega} = \max_{i+j=v} \|\nabla_{s_1}^i \nabla_{s_2}^j g(\mathbf{s})\|_{\infty,\Omega}$ be the maximum norms of all the v-th order derivatives of g over Ω. Let v be a nonnegative integer, and $\delta \in (0, 1]$ such that $\varrho = \delta + v \geq 1$. Let $\mathcal{H}^{(\varrho)}([a, b])$ be the class of functions ψ on $[a, b]$ whose v-th derivative exists and satisfies a Lipschitz condition of order δ: $|\psi^{(v)}(x) - \psi^{(v)}(x')| \leq C_v|x - x'|^\delta$, for $x, x' \in [a, b]$. Let $\mathcal{D}^0([a, b]) = \{g : Eg(Z) = 0, Eg^2(Z) < \infty\}$ be the function space defined on $[a, b]$ and $\mathcal{W}^{d+1,\infty}(\Omega) = \{g : |g|_{k,\infty,\Omega} < \infty, 0 \leq k \leq d + 1\}$ be the standard Sobolev space.

The following are the technical assumptions needed to facilitate the technical details,

(A1) For $k = 1, \ldots, p$, $\beta_{0k} \in \mathcal{H}^{(\varrho)} \cap \mathcal{D}^0$ and the true bivariate function $\alpha_0(\cdot) \in \mathcal{W}^{d+1,\infty}(\Omega)$.

(A2) The density function $f(\mathbf{x}, z, \boldsymbol{s})$ of $(X_1, \ldots, X_p, Z, \boldsymbol{S})$ satisfies

$$0 < c_f \leq \inf_{(\mathbf{x},z,\boldsymbol{s})\in\mathbb{R}^{p+1}\times\Omega} f(\mathbf{x}, z, \boldsymbol{s}) \leq \sup_{(\mathbf{x},z,\boldsymbol{S})\in\mathbb{R}^{p+1}\times\Omega} f(\mathbf{x}, z, \boldsymbol{s}) \leq C_f < \infty.$$

The marginal density function $f_z(\cdot)$ of Z is twice continuously differentiable and the marginal density function $f_{\boldsymbol{s}}(\cdot)$ of $\boldsymbol{S}$ is bounded away from zero and infinity on Ω.

(A3) Recall that $\mathbb{S}_d^r(\triangle)$ denotes the spline space of degree d and smoothness r over $\triangle$. For every $\alpha \in \mathbb{S}_{3r+2}^r$ and every $\tau \in \triangle$, there exists a positive constant F_1, independent of α and τ, such that

$$F_1\|\alpha\|_{\infty,\tau} \leq \left\{ \sum_{\boldsymbol{S}_i \in \tau, i \in \{1,\ldots,n\}} \alpha(\boldsymbol{S}_i)^2 \right\}^{1/2} \leq F_2\|\alpha\|_{\infty,\tau},$$

where $\|\alpha\|_{\infty,\tau}$ denotes the supremum norm of α over triangle τ, F_2 is the largest among the numbers of observations in triangles $\tau \in \triangle$ and $F_2/F_1 = O(1)$.

(A4) The errors satisfy

$$E\left\{\varepsilon_i | \mathbf{X}_i = \mathbf{x}_i, Z_i = z_i, \boldsymbol{S}_i = \boldsymbol{s}_i\right\} = 0$$

and

$$E\left\{\varepsilon_i^{2+\nu} | \mathbf{X}_i = \mathbf{x}_i, Z_i = z_i, \boldsymbol{S}_i = \boldsymbol{s}_i\right\} < \infty$$

for some $\nu \in (3, \infty)$.

(A5) For some positive constant π, $(\min_{\tau \in \triangle} T_\tau)^{-1} \leq |\triangle| \leq \pi$, where T_τ is the radius of the largest disk contained in τ.

(A6) The number of knots J_n for the univariate splines and the triangulation size $|\triangle|$ satisfy that $J_n \to \infty$, $|\triangle| \to 0$, and $J_n \ll |\triangle|^2 n \log^{-1}(n)$; and the smoothness penalty parameter $\lambda_n n^{-1}|\triangle|^{-3} \to 0$.

(A6′) $h = o(n^{-1/5})$. For some $\varrho \geq 1$ and $d \geq 2$, $|\triangle| \ll n^{-2/(5d+5)}$ and $|\triangle|^{1/(\varrho+1)} n^{2/(5\varrho+5)} \ll J_n \ll |\triangle|^2 n \log^{-1}(n)$ and $\lambda_n n^{-1}|\triangle|^{-3} n^{2/5} = o(1)$.

(A7) The kernel function $K(\cdot)$ is a symmetric probability density with bounded support in $[-1, 1]$.

(A8) $\boldsymbol{\Omega}(z) = E(\mathbf{X}_1\mathbf{X}_1^\top | Z = z)$ and $\boldsymbol{\Gamma}(z) = E(\mathbf{X}_1\mathbf{X}_1^\top\mathbf{X}_1^\top\mathbf{X}_1 | Z = z)$ are twice continuously differentiable. $\mathbf{C}(z)$ is uniformly bounded in $[a, b]$.

The above assumptions are regularity conditions that can be satisfied in many practical situations. Assumption (A1) describes the requirement on the varying coefficient functions,

which are frequently used in the literature of non and semi-parametric estimation. Assumptions (A1) and (A2) are similar to Assumptions (A1) and (A2) in [129]. Assumptions (A3) and (A5) are analogue to Assumptions (A2) and (A5) in [129], which has been widely used in the triangulation based literature [116, 57]. Assumptions (A6) and (A6′) show the requirement of the number of interior knots and the size of triangulation to ensure the consistency property of spline estimator and to obtain the local linear estimator, respectively. Note that the Assumption (A6′) only provides the order of $h = o(n^{-1/5})$ to be satisfied. This upper bounds on the bandwidth h in Assumption (A6′), is adapted from [114], which is a necessary condition for Proposition 6.1. The naive empirical log-likelihood ratio is asymptotically non-central if the optimal bandwidth is used, which has been discussed in [124]. To make the likelihood ratio asymptotically parameter free, we adopt the undersmoothing Assumption (A6′). Assumptions (A4), (A7) and (A8) which are analogue to conditions 1, 2 and 3 in [114], are common regularity conditions in non-parametric smoothing literature.

E.2 Preliminaries

In this section, we depict the following bivariate splines properties. We first introduce some notations. For any vector $\mathbf{a} = (a_1, ..., a_n)^\top \in \mathbb{R}^n$, denote the norm $\|\mathbf{a}\|_r = (|a_1|^r + \cdots + |a_n|^r)^{1/r}$, $1 \leq r < +\infty$, $\|\mathbf{a}\|_\infty = \max(|a_1|, \ldots, |a_n|)$. For any matrix $\mathbf{A} = (a_{ij})_{i=1,j=1}^{m,n}$, denote its L_r norm as $\|\mathbf{A}\|_r = \max_{\mathbf{a}\in\mathbb{R}^n, \mathbf{a}\neq\mathbf{0}} \|\mathbf{A}\mathbf{a}\|_r \|\mathbf{a}\|_r^{-1}$, for $r < +\infty$ and $\|\mathbf{A}\|_r = \max_{1\leq i\leq m} \sum_{j=1}^n |a_{ij}|$, for $r = \infty$. Given sequences of positive numbers a_n and b_n, $a_n \lesssim b_n$ means a_n/b_n is bounded, and $a_n \asymp b_n$ means both $a_n \lesssim b_n$ and $a_n \gtrsim b_n$ hold. We define the norm on the space $\mathcal{G}$. For any functions $\phi_1, \phi_2 \in \mathcal{G}$, define their theoretical inner product $\langle \phi_1, \phi_2 \rangle$ as $\langle \phi_1, \phi_2 \rangle = E\phi_1(\mathbf{X}, Z, \boldsymbol{S})\phi_2(\mathbf{X}, Z, \boldsymbol{S})$. Define their empirical inner product $\langle \phi_1, \phi_2 \rangle_n$ as $\langle \phi_1, \phi_2 \rangle_n = \frac{1}{n}\sum_{i=1}^n \phi_1(\mathbf{X}_i, Z_i, \boldsymbol{S}_i)\phi_2(\mathbf{X}_i, Z_i, \boldsymbol{S}_i)$. Hence, $\|\phi\| = \sqrt{\langle \phi, \phi \rangle}$ and $\|\phi\|_n = \sqrt{\langle \phi, \phi \rangle_n}$.

LEMMA **E.30.** *(Theorem 10.2, [56]) Suppose that $|\triangle|$ is a π-quasi-uniform triangleation of a polygonal domain Ω, and $\phi(\cdot) \in \mathcal{W}^{d+1,\infty}(\Omega)$.*

(i) For bi-integer(α_1, α_2) *with* $0 \leq a_1 + a_2 \leq d$*, there exists a spline* $\phi^*(\cdot) \in \mathbb{S}_d^0(\triangle)$ *such that* $\|\nabla_{s_1}^{a_1} \nabla_{s_2}^{a_2}(\phi - \phi^*)\|_\infty \leq C|\triangle|^{d+1-a_1-a_2}|\phi|_{d+1,\infty}$ *where* C *is a constant depending on* d *and shape parameter* π*.*

(ii) For bi-integer(α_1, α_2) *with* $0 \leq a_1 + a_2 \leq d$*, there exists a spline* $\phi^{**}(\cdot) \in \mathbb{S}_d^0(\triangle)$ *(*$d \geq 3r + 2$*) such that* $\|\nabla_{s_1}^{a_1} \nabla_{s_2}^{a_2}(\phi - \phi^{**})\|_\infty \leq C|\triangle|^{d+1-a_1-a_2}|\phi|_{d+1,\infty}$ *where* C *is a constant depending on* d*,* r *and shape parameter* π*.*

Lemma E.30 shows that $\mathbb{S}_d^0(\triangle)$ has full approximation power, and $\mathbb{S}_d^0(\triangle)$ also has full approximation power if $d \geq 3r + 2$.

LEMMA **E.31.** *(Lemma B.4, [129]) For any* $k = 1, \cdots, p$*,* $\phi_k \in \mathcal{H}^{(\varrho)} \cap \mathcal{D}_k^0$*, there exist a constant* c *and a function* $\phi_k^* \in \mathcal{U}_k^0$ *such that* $\|\phi_k - \phi_k^*\|_\infty \leq c\|\phi_k^{(\varrho+1)}\|_\infty J_n^{-\varrho-1}$*.*

LEMMA **E.32.** *Suppose that Assumptions (A2), (A5) and (A6) hold. Then*

$$\sup_{\phi_1, \phi_2 \in \mathcal{A}} \left| \frac{\langle \phi_1, \phi_2 \rangle_n - \langle \phi_1, \phi_2 \rangle}{\|\phi_1\| \|\phi_2\|} \right| = O_{a.s.}\left(J_n^{1/2} |\triangle|^{-1} n^{-1/2} \log^{1/2} n \right)$$

where $\mathcal{A} = \Big\{ \phi : \phi(\mathbf{x}, z, \boldsymbol{S}) = \sum_{k=1}^p \sum_{j \in \mathcal{J}} \eta_{kj} U_{kj}(z) x_k + \sum_{m \in \mathcal{M}} \gamma_m B_m(\boldsymbol{s}),$
$x_k, z, \eta_{kj}, \gamma_m \in \mathbb{R}, \boldsymbol{s} \in \Omega \}$.

Proof. The proof is similar as the proof of Lemma B.7 in [129]. □

LEMMA **E.33.** *Under Assumptions (A2), (A5) and (A6), there exist constants* $0 < c_A < C_A < \infty$*, such that* $c_A \leq \lambda_{min}(n\mathbf{A}_{11}) \leq \lambda_{max}(n\mathbf{A}_{11}) \leq C_A$*, where* $\mathbf{A}_{11}$ *is given in (6.2.2).*

Proof. The proof is similar as the proof of Lemma B.8 in [129]. Details are omitted. □

E.3 Proof of Theorem 6.1

Proof. We first prove the consistency of $\widehat{\alpha}$. Define $\mathbf{H}_w = \mathbf{I} - \mathbf{W}(\mathbf{W}^\top \mathbf{W})^{-1} \mathbf{W}^\top$. Note that

$$\begin{aligned} \widehat{\boldsymbol{\theta}} &= \mathbf{A}_{22} \mathbf{Q}_2^\top \mathbf{B}^\top \mathbf{H}_w \mathbf{Y} \\ &= \mathbf{A}_{22} \mathbf{Q}_2^\top \mathbf{B}^\top \mathbf{H}_w (\boldsymbol{\beta}_0^\top(Z)\mathbf{X} + \alpha_0(\boldsymbol{S})) + \mathbf{A}_{22} \mathbf{Q}_2^\top \mathbf{B}^\top \mathbf{H}_w \boldsymbol{\varepsilon} \\ &= \widetilde{\boldsymbol{\theta}}_\mu + \widetilde{\boldsymbol{\theta}}_\varepsilon. \end{aligned}$$

According to Lemmas E.30 and E.31, there exist $\alpha^*(\boldsymbol{S}) = \mathbf{B}(\boldsymbol{S})\mathbf{Q}_2\boldsymbol{\theta}_0$ and $\boldsymbol{\beta}^*(z) = U(z)\boldsymbol{\eta}_0$, which are the best approximation to α_0 and $\boldsymbol{\beta}_0$ with the approximation rate at $\|\alpha^* - \alpha_0\|_\infty \leq C_\alpha|\triangle|^{d+1}|\alpha_0|_{d+1,\infty}$ and $\|\boldsymbol{\beta}_0(z) - U(z)\boldsymbol{\eta}_0\|_\infty \leq C_\beta J_n^{-\varrho-1}$. Hence, it is easy to find that $\|\boldsymbol{\beta}_0(Z)^\top\mathbf{X} - \mathbf{W}\boldsymbol{\eta}_0\|_\infty = O_p(C_\beta J_n^{-\varrho-1})$. Denote by $\boldsymbol{\gamma}_0 = \mathbf{Q}_2\boldsymbol{\theta}_0$ the spline coefficients of α^*. We have the following decomposition: $\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0 = \widetilde{\boldsymbol{\theta}}_\mu - \boldsymbol{\theta}_0 + \widetilde{\boldsymbol{\theta}}_\varepsilon$. Note that

$$\begin{aligned}\|\widetilde{\boldsymbol{\theta}}_\mu - \boldsymbol{\theta}_0\| \;\leq\; & \left\|\mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{H}_w\boldsymbol{\beta}_0^\top(Z)\mathbf{X}\right\| \\ & + \left\|\mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{H}_w(\alpha_0 - \mathbf{B}\mathbf{Q}_2\boldsymbol{\theta}_0) - \lambda_n\mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{P}\mathbf{Q}_2\boldsymbol{\theta}_0\right\|.\end{aligned}$$

For any vector $\mathbf{a}$, according to Lemma E.33 and the proof of Theorem 2 in [116], one has $n\mathbf{a}^\top\mathbf{A}_{22}\mathbf{a} \leq C|\triangle|^{-2}$. Hence, we have

$$\begin{aligned}& \left\|\mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{H}_w\boldsymbol{\beta}_0^\top(Z)\mathbf{X}\right\| \leq C^{1/2}|\triangle|^{-1}n^{-1}\left\|\mathbf{B}^\top\mathbf{H}_w(\mathbf{W}\boldsymbol{\eta} + O_p(h^p)\mathbf{1})\right\| \\ \leq\; & O_p\left(J_n^{-\varrho-1}\right)|\triangle|^{-1}n^{-1}\left[\sum_{m\in\mathcal{M}}\{\mathbf{B}_m^\top\mathbf{H}_w\mathbf{1}\}^2\right]^{1/2} = O_p\left(J_n^{-p}\right).\end{aligned}$$

Similarly,

$$\begin{aligned}& \left\|\mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{H}_w(\alpha_0 - \mathbf{B}\mathbf{Q}_2\boldsymbol{\theta}_0)\right\| \\ \leq\; & C^{1/2}|\triangle|^{-1}n^{-1}\left[\sum_{m\in\mathcal{M}}\{\mathbf{B}_m^\top\mathbf{H}_w(\alpha_0 - \mathbf{B}\mathbf{Q}_2\boldsymbol{\theta}_0)\}^2\right]^{1/2} \\ =\; & O_p\left(|\triangle|^d|\alpha_0|_{d+1,\infty}\right),\end{aligned}$$

and $\lambda_n\|\mathbf{A}_{22}\mathbf{Q}_2^\top\mathbf{P}\mathbf{Q}_2\boldsymbol{\theta}_0\| \leq \frac{\lambda_n}{n|\triangle|^4}\left(|\alpha_0|_{2,\infty} + |\triangle|^{d-1}|\alpha_0|_{d+1,\infty}\right)$.

Thus,

$$\|\widetilde{\boldsymbol{\theta}}_\mu - \boldsymbol{\theta}_0\| = O_p\left\{J_n^{-\varrho-1} + \frac{\lambda_n}{n|\triangle|^4}|\alpha_0|_{2,\infty} + \left(1 + \frac{\lambda_n}{n|\triangle|^5}\right)|\triangle|^d|\alpha_0|_{d+1,\infty}\right\}.$$

For any $\boldsymbol{b}$ with $\|\boldsymbol{b}\| = 1$, we have $\boldsymbol{b}^\top\widetilde{\boldsymbol{\theta}}_\varepsilon = \sum_{i=1}^n \alpha_i\varepsilon_i$ and

$$\alpha_i^2 = \boldsymbol{b}^\top\mathbf{A}_{22}\mathbf{Q}_2\mathbf{B}^\top\mathbf{H}_w\mathbf{B}\mathbf{Q}_2\mathbf{A}_{22}\boldsymbol{b}.$$

Following the similar argument in Lemma S.7 in [116], we have $\max_{1\leq i\leq n}\alpha_i^2 = O_p\left(n^{-2}|\triangle|^{-2}\right)$. Thus,

$$\|\widetilde{\boldsymbol{\theta}}_\varepsilon\| \leq |\triangle|^{-1}|\boldsymbol{\alpha}^\top\widetilde{\boldsymbol{\theta}}_\varepsilon| = |\triangle|^{-1}\left|\sum_{i=1}^n \alpha_i\varepsilon_i\right| = O_p\left(n^{-1/2}|\triangle|^{-2}\right).$$

Hence,

$$\begin{aligned} & \|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0\| \\ = \;& O_p\left\{J_n^{-\varrho-1} + n^{-1/2}|\triangle|^{-2} + \frac{\lambda_n}{n|\triangle|^4}|\alpha_0|_{2,\infty} + \left(1+\frac{\lambda_n}{n|\triangle|^5}\right)|\triangle|^d|\alpha_0|_{d+1,\infty}\right\}. \end{aligned}$$

Observing that $\widehat{\alpha}(\boldsymbol{S}) = \mathbf{B}(\boldsymbol{S})\widehat{\boldsymbol{\gamma}} = \mathbf{B}(\boldsymbol{S})\mathbf{Q}_2\widehat{\boldsymbol{\theta}}$, we have

$$\begin{aligned} & \|\widehat{\alpha} - \alpha_0\|_{L_2} \\ \leq \;& \|\widehat{\alpha} - \rho_{0,\alpha_0}\|_{L_2} + |\triangle|^{d+1}|\alpha_0|_{d+1,\infty} \\ \leq \;& C\left(|\triangle|\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0\| + |\triangle|^{d+1}|\alpha_0|_{d+1,\infty}\right) \\ = \;& O_p\left\{J_n^{-\varrho-1}|\triangle| + n^{-1/2}|\triangle|^{-1} + \frac{\lambda_n}{n|\triangle|^3}|\alpha_0|_{2,\infty}\right. \\ & \left. + \left(1+\frac{\lambda_n}{n|\triangle|^5}\right)|\triangle|^{d+1}|\alpha_0|_{d+1,\infty}\right\}. \end{aligned}$$

Next, we prove the consistency for $\widehat{\boldsymbol{\beta}}$.

Define $\mathbf{H}_B = \mathbf{I} - \mathbf{B}\mathbf{Q}_2\left\{\mathbf{Q}_2^\top\left(\mathbf{B}^\top\mathbf{B} + \lambda_n\mathbf{P}\right)\mathbf{Q}_2\right\}^{-1}\mathbf{Q}_2^\top\mathbf{B}^\top$. Let $\boldsymbol{\alpha}_0 = (\alpha_0(\boldsymbol{S}_1), \ldots, \alpha_0(\boldsymbol{S}_n))^\top$ and note that

$$\widehat{\boldsymbol{\eta}} = \mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B\mathbf{Y} = \mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B\left(\boldsymbol{\beta}_0^\top(Z)\mathbf{X} + \boldsymbol{\alpha}_0\right) + \mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B\boldsymbol{\varepsilon} = \widetilde{\boldsymbol{\eta}}_\mu + \widetilde{\boldsymbol{\eta}}_\varepsilon.$$

Note that,

$$\begin{aligned} \|\widetilde{\boldsymbol{\eta}}_\mu - \boldsymbol{\eta}_0\| \;&\leq\; \left\|\mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B(\boldsymbol{\beta}_0^\top(Z)\mathbf{X} - \mathbf{W}\boldsymbol{\eta}_0)\right\| + \left\|\mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B\boldsymbol{\alpha}_0\right\| \\ &\leq\; O_p\left(J_n^{-\varrho-1}\right)\|\mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B\mathbf{1}\| + \left\|\mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B\boldsymbol{\alpha}_0\right\| \\ &=\; O(1)\|\mathbf{A}_{11}\mathbf{W}^\top\mathbf{H}_B\boldsymbol{\alpha}_0\|. \end{aligned}$$

By the Lemma E.33, there exist constants $0 \leq c_A < C_A < \infty$, such that with probability approaching 1 as $n \to \infty$,

$$c_A \mathbf{I}_{((J_n+\varrho+1))\times(J_n+\varrho+1)} \leq n\mathbf{A}_{11} \leq C_A \mathbf{I}_{(J_n+\varrho+1)\times(J_n+\varrho+1)}.$$

Hence, we have

$$\begin{aligned} & \|\widetilde{\boldsymbol{\eta}}_\mu - \boldsymbol{\eta}_0\| \\ \leq\ & O(1)\|n^{-1}\mathbf{W}^\top \left(\mathbf{I} - \mathbf{B}\mathbf{Q}_2\{\mathbf{Q}_2^\top(\mathbf{B}^\top\mathbf{B} + \lambda_n\mathbf{P})\mathbf{Q}_2\}^{-1}\mathbf{Q}_2^\top\mathbf{B}^\top\right)\boldsymbol{\alpha}_0\| \\ =\ & O(1)\|\mathbf{R}\|, \end{aligned}$$

where $\mathbf{R} = (R_1, \ldots, R_{p(J_n+\varrho+1)})^\top$, with

$$R_j = n^{-1}\mathbf{W}_j^\top \left[\boldsymbol{\alpha}_0 - \mathbf{B}\mathbf{Q}_2\{\mathbf{Q}_2^\top(\mathbf{B}^\top\mathbf{B} + \lambda_n\mathbf{P})\mathbf{Q}_2\}^{-1}\mathbf{Q}_2^\top\mathbf{B}^\top\boldsymbol{\alpha}_0\right]$$

for $\mathbf{W}_j^\top = (W_{1j}, \ldots, W_{nj})$. Next we derive the order of R_j, $j = 1, \ldots, p(J_n + \varrho + 1)$. For any $\alpha_j \in \mathbb{S}$, we have $R_j = \langle w_j, \alpha_0 - \rho_{\lambda,\alpha_0}\rangle_n = \langle w_j - \alpha_j, \alpha_0 - \rho_{\lambda,\alpha_0}\rangle_n + \lambda_n n^{-1}\langle \rho_{\lambda,\alpha_0}, \alpha_j\rangle_{\mathcal{E}}$, where $\rho_{\lambda,\alpha_0} = \arg\min_{\rho\in\mathbb{S}} \sum_{i=1}^n \{\alpha_0(\boldsymbol{S}_i) - \rho(\boldsymbol{S}_i)\}^2 + \frac{\lambda}{2}\mathcal{E}(\rho)$ is the penalized least-squares splines of $\alpha(\cdot,\cdot)$.

By Assumptions (A1)-(A6) and Lemma S.6 in [116], $|R_j| = o_p(n^{-1/2})$, for $j = 1, \ldots, p(J_n + \varrho + 1)$. Therefore, $\|\widetilde{\boldsymbol{\eta}}_\mu - \boldsymbol{\eta}_0\| = O_p(n^{-1/2}J_n^{1/2})$.

Note that $\widetilde{\boldsymbol{\eta}}_\varepsilon = \mathbf{A}_{11}\mathbf{W}^\top \left(\mathbf{I} - \mathbf{B}\mathbf{Q}_2\mathbf{V}_{22}^{-1}\mathbf{Q}_2^\top\mathbf{B}^\top\right)\boldsymbol{\varepsilon}$. For any $\boldsymbol{b}$ with $\|\boldsymbol{b}\| = 1$, we have $\boldsymbol{b}^\top\widetilde{\boldsymbol{\eta}}_\varepsilon = \sum_{i=1}^n \alpha_i\varepsilon_i$ and

$$\alpha_i^2 = n^{-2}\boldsymbol{b}^\top(n\mathbf{A}_{11})\left(\mathbf{W}_i^\top - \mathbf{V}_{21}\mathbf{V}_{22}^{-1}\mathbf{Q}_2^\top\mathbf{B}_i\right)\left(\mathbf{W}_i - \mathbf{B}_i^\top\mathbf{Q}_2\mathbf{V}_{22}^{-1}\mathbf{V}_{21}\right)(n\mathbf{A}_{11})\boldsymbol{b},$$

and conditioning on $\{(\mathbf{W}_i, \boldsymbol{S}_i), i = 1, \ldots, n\}$, $\alpha_i\varepsilon_i$'s are independent. By Lemma E.33, we have that $\max_{1\leq i\leq n}\alpha_i^2 \leq Cn^{-2}\max_{1\leq i\leq n}\left\{\|\mathbf{W}_i\|^2 + \|\mathbf{V}_{12}\mathbf{V}_{22}^{-1}\mathbf{Q}_2^\top\mathbf{B}_i\|^2\right\}$, where for any $\mathbf{b} \in$

$\mathbb{R}^p$,

$$\begin{aligned} & \mathbf{b}^\top \mathbf{V}_{12} \mathbf{V}_{22}^{-1} \mathbf{Q}_2^\top \mathbf{B}_i \mathbf{b} \\ = \; & n^{-1} \mathbf{b}^\top \mathbf{V}_{12} \left(\mathbf{Q}_2^\top \mathbf{\Gamma}_{n,\lambda} \mathbf{Q}_2 \right)^{-1} \mathbf{Q}_2^\top \mathbf{B}_i \mathbf{b} \leq C n^{-1} |\triangle|^{-2} \mathbf{b}^\top \mathbf{W}^\top \mathbf{B} \mathbf{B}_i \mathbf{b} \end{aligned}$$

and the j-th component of $n^{-1}\mathbf{W}^\top \mathbf{B}\mathbf{B}_i$ is

$$n^{-1} \sum_{i'=1}^{n} W_{i'j} \sum_{m \in \mathcal{M}} B_m(\boldsymbol{S}_{i'}) B_m(\boldsymbol{S}_i).$$

Under Assumption (A2), we have

$$E \left\{ n^{-1} \sum_{i'=1}^{n} W_{i'j} \sum_{m \in \mathcal{M}} B_m(\boldsymbol{S}_{i'}) B_m(\boldsymbol{S}_i) \right\}^2 = O(1),$$

for large n. Thus with probability approaching 1,

$$\begin{aligned} & \max_{1 \leq i \leq n} \left| \frac{1}{n} \sum_{i'=1}^{n} W_{i'j} \sum_{m \in \mathcal{M}} B_m(\boldsymbol{S}_{i'}) B_m(\boldsymbol{S}_i) \right| = O_p(1), \\ & \max_{1 \leq i \leq n} \| \mathbf{V}_{12} \mathbf{V}_{22}^{-1} \mathbf{Q}_2^\top \mathbf{B}_i \|^2 = O_p(|\triangle|^{-2}). \end{aligned}$$

Therefore, $\max_{1 \leq i \leq n} \alpha_i^2 = O_p\{n^{-2}(|\triangle|^{-2} + J_n)\}$ and $\|\widetilde{\boldsymbol{\eta}}_\varepsilon\| = O_p(n^{-1}|\triangle|^{-1} + n^{-1} J_n^{1/2})$. Let $\widehat{\boldsymbol{\eta}} = (\widehat{\boldsymbol{\eta}}_1, \ldots, \widehat{\boldsymbol{\eta}}_p)$. $\|\boldsymbol{\beta}_0(z) - U(z)\boldsymbol{\eta}_0\|_\infty \leq C_\beta J_n^{-\varrho-1}$ and observing that $\widehat{\beta}_k(Z) = \mathbf{U}_k^\top(Z)\widehat{\boldsymbol{\eta}}_k$, we have $\|\widehat{\beta}_k - \beta_{0k}\|_{L_2} \leq C\left(\|\widehat{\boldsymbol{\eta}}_k - \boldsymbol{\eta}_{0k}\| + J_n^{-\varrho-1}\right) = O_p\left(n^{-1/2} J_n^{1/2} + n^{-1}|\triangle|^{-1} + J_n^{-\varrho-1}\right)$, and the consistency of $\widehat{\boldsymbol{\beta}}$ is proved. □

E.4 Proof of Proposition 6.1

Proof. Recall that $\mathbf{\Omega}(z) = E\left(\mathbf{X}_i \mathbf{X}_i^\top | Z = z\right)$, $\mathbf{\Gamma}(z) = E\left(\mathbf{X}_i \mathbf{X}_i^\top \mathbf{X}_i^\top \mathbf{X}_i | Z = z\right)$. By the definition of $g_i\{\boldsymbol{\beta}_0(z)\}$, we have the following decomposition,

$$\begin{aligned} g_i\{\boldsymbol{\beta}_0(z)\} &= \left\{ Y_i - \boldsymbol{\beta}_0^\top(z)\mathbf{X}_i - \widehat{\alpha}(\boldsymbol{S}_i) \right\} \mathbf{X}_i K_h(Z_i - z) \\ &= \left\{ Y_i - \boldsymbol{\beta}_0^\top(Z_i)\mathbf{X}_i - \alpha_0(\boldsymbol{S}_i) + \boldsymbol{\beta}_0^\top(Z_i)\mathbf{X}_i - \boldsymbol{\beta}_0^\top(z)\mathbf{X}_i \right. \end{aligned}$$

$$
\begin{aligned}
& +\alpha_0(\boldsymbol{S}_i) - \widehat{\alpha}(\boldsymbol{S}_i)\} \mathbf{X}_i K_h(Z_i - z) \\
= & \left\{\epsilon_i + [\boldsymbol{\beta}_0(Z_i) - \boldsymbol{\beta}_0(z)]^\top \mathbf{X}_i + [\alpha_0(\boldsymbol{S}_i) - \widehat{\alpha}(\boldsymbol{S}_i)]\right\} \\
& \times \mathbf{X}_i K_h(Z_i - z) \\
= & \epsilon_i \mathbf{X}_i K_h(Z_i - z) + \mathbf{X}_i \mathbf{X}_i^\top [\boldsymbol{\beta}_0(Z_i) - \boldsymbol{\beta}_0(z)] K_h(Z_i - z) \\
& + [\alpha_0(\boldsymbol{S}_i) - \widehat{\alpha}(\boldsymbol{S}_i)] \mathbf{X}_i K_h(Z_i - z) \\
:= & \boldsymbol{\xi}_i + \mathbf{L}_{1i} + \mathbf{L}_{2i}.
\end{aligned}
$$

Denote $\boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}) = \left(\beta'_{01}(z_1), \beta'_{02}(z_2), \ldots, \beta'_{0p}(z_p)\right)^\top$. By the smoothness of β_{0k}, $k = 1, 2, \ldots, p$, we have $\boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}) = O(1)\mathbf{1}_{p\times 1}$ for all $\boldsymbol{z}$.

It is clear that $E\boldsymbol{\xi}_i = \mathbf{0}$, and we have

$$
\begin{aligned}
E\left(\mathbf{L}_{1i}\right) &= E\left\{E(\mathbf{X}_i\mathbf{X}_i^\top | Z_i)\left[\boldsymbol{\beta}_0^{(1)}(z^*)(Z_i - z)\right] K_h(Z_i - z)\right\} \\
&= \boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}^*) E\left[\boldsymbol{\Omega}(Z_i)(Z_i - z)K_h(Z_i - z)\right]) \\
&= \boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}^*) \int_a^b \boldsymbol{\Omega}(u)(u - z)K_h(u - z)f(u)du \\
&= \boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}^*) \Big[h \int_a^b v(\boldsymbol{\Omega}(z) + \boldsymbol{\Omega}'(z)hv + 1/2\boldsymbol{\Omega}'(z)h^2v^2)K(v)(f(z) \\
&\quad + f'(z)hv + 1/2f''(z)h^2v^2)dv\Big] \\
&= O(h^2)\mathbf{1}_{n\times 1}.
\end{aligned}
$$

According to the proof of Theorem 6.1, we also have

$$
\begin{aligned}
E\left(\mathbf{L}_{2i}\right) &= E\{\mathbf{X}_i K_h(Z_i - z)(\alpha_0(\boldsymbol{S}_i) - \widehat{\alpha}(\boldsymbol{S}_i))\} \\
&= E\left\{\mathbf{X}_i K_h(Z_i - z) E\left[(\alpha_0(\boldsymbol{S}_i) - \widehat{\alpha}(\boldsymbol{S}_i)) \mid \{\mathbf{X}_i, Z_i, \boldsymbol{S}_i\}_{i=1}^n\right]\right\} \\
&= E\left\{\mathbf{X}_i K_h(Z_i - z)\left(\alpha_0(\boldsymbol{S}_i) - E\left[\widehat{\alpha}(\boldsymbol{S}_i) | \{\mathbf{X}_i, Z_i, \boldsymbol{S}_i\}_{i=1}^n\right]\right)\right\} \\
&= E\left\{\mathbf{X}_i K_h(Z_i - z)\left(\alpha_0(\boldsymbol{S}_i) - \boldsymbol{B}(\boldsymbol{S}_i)\boldsymbol{Q}_2\widetilde{\boldsymbol{\theta}}_\mu\right)\right\} \\
&\leq E\left\{|\mathbf{X}_i K_h(Z_i - z)|\right\} \\
&\quad \times E\left(\left\|\alpha_0(\boldsymbol{S}_i) - \alpha^*(\boldsymbol{S}_i) + \boldsymbol{B}(\boldsymbol{S}_i)\boldsymbol{Q}_2\left(\boldsymbol{\theta}_0 - \widetilde{\boldsymbol{\theta}}_\mu\right)\right\|_\infty\right) \\
&\lesssim E\left\{|\mathbf{X}_i K_h(Z_i - z)|\right\} E\left(\|\alpha_0 - \alpha^*\|_\infty + O(|\triangle|)\|\boldsymbol{\theta}_0 - \widetilde{\boldsymbol{\theta}}_\mu\|_2\right),
\end{aligned}
$$

where by Theorem 6.1, we have

$$E\left(\|\alpha_0(\boldsymbol{S}_i)-\alpha^*\|_\infty+|\Delta|\|\boldsymbol{\theta}_0-\widetilde{\boldsymbol{\theta}}_\mu\|_2\right)=O\left(J_n^{-\varrho-1}|\triangle|+\frac{\lambda_n}{n|\triangle|^3}+|\triangle|^{d+1}\right),$$

and $E\{|\mathbf{X}_iK_h(Z_i-z)|\}=E\left[E\{|\mathbf{X}_i|\big|Z_i\}K_h(Z_i-z)\right]\asymp E\left[K_h(Z_i-z)\right]\times\mathbf{1}_{p\times1}=O(f(z))\mathbf{1}_{p\times1}$. If $h=o(n^{-1/5})$, when $|\triangle|\ll n^{-2/(5d+5)}$ and $J_n\gg|\triangle|^{1/(\varrho+1)}n^{2/(5\varrho+5)}$, we have $E\mathbf{L}_{2i}=O(h^2)\mathbf{1}_{p\times1}$ by Assumption (A6′). Therefore, we have $E\{g_i\{\boldsymbol{\beta}_0(z)\}\}=O(h^2)\mathbf{1}_{p\times1}$. In the following, we calculate the variance of $g_i\{\boldsymbol{\beta}_0(z)\}$. Firstly, we have

$$\begin{aligned}E\left(\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top\right)&=E\left\{\epsilon_i^2\mathbf{X}_i\mathbf{X}_i^\top K_h^2(Z_i-z)\right\}\\&=\sigma^2E\left[E\left(\mathbf{X}_i\mathbf{X}_i^\top|Z_i\right)K_h^2(Z_i-z)\right]\\&=\sigma^2\boldsymbol{\Omega}(z)f(z)\mu_{20}h^{-1}\left(1+o(1)\right).\end{aligned}$$

Secondly, we have

$$\begin{aligned}&E\left(\mathbf{L}_{1i}\mathbf{L}_{1i}^\top\right)\\=\;&E\left\{\mathbf{X}_i\mathbf{X}_i^\top K_h^2(Z_i-z)\mathbf{X}_i^\top(\boldsymbol{\beta}_0(Z_i)-\boldsymbol{\beta}_0(z))(\boldsymbol{\beta}_0(Z_i)-\boldsymbol{\beta}_0(z))^\top\mathbf{X}_i\right\}\\=\;&E\Big\{\mathbf{X}_i\mathbf{X}_i^\top K_h^2(Z_i-z)\Big[(Z_i-z)^2\mathbf{X}_i^\top\boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}^*)\boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}^*)^\top\mathbf{X}_i\\&+o(Z_i-z)^2\mathbf{X}_i^\top\mathbf{X}_i\Big]\Big\}\\=\;&E\left\{E\left[\mathbf{X}_i\mathbf{X}_i^\top\mathbf{X}_i^\top\boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}^*)\boldsymbol{\beta}_0^{(1)}(\boldsymbol{z}^*)^\top\mathbf{X}_i\big|Z_i\right]K_h^2(Z_i-z)(Z_i-z)^2\right\}\\&\times\{1+o(1)\}\\\asymp\;&E\left\{E\left[\mathbf{X}_i\mathbf{X}_i^\top\mathbf{X}_i^\top\mathbf{X}_i\big|Z_i\right]K_h^2(Z_i-z)(Z_i-z)^2\right\}(1+o(1))\\=\;&E\left\{\boldsymbol{\Gamma}(Z_i)K_h^2(Z_i-z)(Z_i-z)^2\right\}(1+o(1))=\boldsymbol{\Gamma}(z)f(z)\mu_{22}h\left(1+o(1)\right).\end{aligned}$$

Finally,

$$\begin{aligned}E\left(\mathbf{L}_{2i}\mathbf{L}_{2i}^\top\right)&=E\left\{\mathbf{X}_i\mathbf{X}_i^\top(\alpha_0(\boldsymbol{S}_i)-\widehat{\alpha}(\boldsymbol{S}_i))^2K_h^2(Z_i-z)\right\}\\&=E\left\{E\left[\mathbf{X}_i\mathbf{X}_i^\top(\alpha_0(\boldsymbol{S}_i)-\widehat{\alpha}(\boldsymbol{S}_i))^2K_h^2(Z_i-z)|\left\{\mathbf{X}_i,Z_i,\boldsymbol{S}_i\right\}_{i=1}^n\right]\right\}\\&=E\left\{E\left[(\alpha_0(\boldsymbol{S}_i)-\widehat{\alpha}(\boldsymbol{S}_i))^2|\left\{\mathbf{X}_i,Z_i,\boldsymbol{S}_i\right\}_{i=1}^n\right]\mathbf{X}_i\mathbf{X}_i^\top K_h^2(Z_i-z)\right\}\end{aligned}$$

$$
\begin{aligned}
&= E\Big\{E\Big[\mathbf{B}(\boldsymbol{S}_i)\mathbf{Q}_2(\boldsymbol{\theta}_0-\widehat{\boldsymbol{\theta}})(\boldsymbol{\theta}_0-\widehat{\boldsymbol{\theta}})^\top\mathbf{Q}_2^\top\mathbf{B}^\top(\boldsymbol{S}_i)\Big] \\
&\quad \times \mathbf{X}_i\mathbf{X}_i^\top K_h^2(Z_i-z)\Big\} \\
&= E\Big\{\mathbf{X}_i\mathbf{X}_i^\top K_h^2(Z_i-z)\|\mathbf{B}(\boldsymbol{S}_i)\mathbf{Q}_2(\boldsymbol{\theta}_0-\widetilde{\boldsymbol{\theta}}_\mu)\|_2^2\Big\} \\
&\quad +\sigma^2 E\left\{\mathbf{X}_i\mathbf{X}_i^\top K_h^2(Z_i-z)\|\mathbf{B}(\boldsymbol{S}_i)\mathbf{Q}_2\boldsymbol{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{H}_{\boldsymbol{\omega}}\|_2^2\right\}.
\end{aligned}
$$

On the one hand, for (k,k')-th entry, k, $k'=1,2,\ldots,p$, we have

$$
\begin{aligned}
& E\Big\{X_{ik}X_{ik'}K_h^2(Z_i-z)\|\mathbf{B}(\boldsymbol{S}_i)\mathbf{Q}_2(\boldsymbol{\theta}_0-\widetilde{\boldsymbol{\theta}}_\mu)\|_2^2\Big\} \\
\le\ & E\Big\{|\Delta|^2X_{ik}X_{ik'}K_h^2(Z_i-z)\|\mathbf{Q}_2(\boldsymbol{\theta}_0-\widetilde{\boldsymbol{\theta}}_\mu)\|_2^2\Big\} \\
\le\ & O\left(|\Delta|^2\right)\left(E\left\{X_{ik}^2X_{ik'}^2K_h^4(Z_i-z)\right\}\right)^{1/2}\left(E\|\boldsymbol{\theta}_0-\widetilde{\boldsymbol{\theta}}_\mu\|_2^4\right)^{1/2} \\
=\ & O\left(|\Delta|^2h^{-3/2}\right)\left(E\left\{\|\boldsymbol{\theta}_0-\widetilde{\boldsymbol{\theta}}_\mu\|_2^4\right\}\right)^{1/2}. \qquad \text{(E1)}
\end{aligned}
$$

On the other hand, for (k,k')-th entry, k, $k'=1,2,\ldots,p$, we have

$$
\begin{aligned}
& E\left\{X_{ik}X_{ik'}K_h^2(Z_i-z)\|\mathbf{B}(\boldsymbol{S}_i)\mathbf{Q}_2\boldsymbol{A}_{22}\mathbf{Q}_2^\top\mathbf{B}^\top\mathbf{H}_{\boldsymbol{\omega}}\|_2^2\right\} \\
\le\ & C|\Delta|^{-4}n^{-2}E\left\{X_{ik}X_{ik'}K_h^2(Z_i-z)\|\mathbf{B}(\boldsymbol{S}_i)\mathbf{B}^\top\mathbf{H}_{\boldsymbol{\omega}}\|_2^2\right\} \\
\le\ & C|\Delta|^{-4}n^{-2}\left(E\left\{X_{ik}^2X_{ik'}^2K_h^4(Z_i-z)\right\}\right)^{1/2}\left(E\left\{\|\mathbf{B}(\boldsymbol{S}_i)\mathbf{B}^\top\mathbf{H}_{\boldsymbol{\omega}}\|_2^4\right\}\right)^{1/2} \\
=\ & O\left(n^{-1}h^{-3/2}\right). \qquad \text{(E2)}
\end{aligned}
$$

Combining (E1) and (E2), we have $E\mathbf{L}_{2i}\mathbf{L}_{2i}^\top = O(|\Delta|^2h^{-3/2}+n^{-1}h^{-3/2})\mathbf{1}_{p\times p}$. Hence, we have

$$
\begin{aligned}
Var\{g_i\{\boldsymbol{\beta}_0(z)\}\} &= E\left\{g_i\{\boldsymbol{\beta}_0(z)\}g_i\{\boldsymbol{\beta}_0(z)\}^\top\right\} \\
&= E\left(\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top+\mathbf{L}_{1i}\mathbf{L}_{1i}^\top+\mathbf{L}_{2i}\mathbf{L}_{2i}^\top\right) \\
&= E\left(\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top\right)(1+o(1)) \\
&= \sigma^2\boldsymbol{\Omega}(z)f(z)\mu_{20}h^{-1}(1+o(1)).
\end{aligned}
$$

□

E.5 Proof of Theorem 6.2

Proof. First, for convenience we suppress the argument z in the functions such as $\boldsymbol{\beta}(z)$, $\boldsymbol{\Omega}(z)$ and so on, since we fix $z \in [a, b]$ in this proof.

For the minimization problem (6.8), we use the Lagrange multiplier method:

$$\min \frac{1}{n} \sum_{i=1}^{n} \log \left[1 + \boldsymbol{\delta}^\top(z) g_i\{\boldsymbol{\beta}(z)\}\right] + \boldsymbol{\nu}^\top(z) H\{\boldsymbol{\beta}(z)\},$$

where $\boldsymbol{\nu}(z)$ is a $q \times 1$ vector of Lagrange multipliers. Define

$$M_{1n}(\boldsymbol{\beta}, \boldsymbol{\delta}) = \frac{1}{n} \sum_{i=1}^{n} \frac{g_i(\boldsymbol{\beta})}{1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta})}$$

and

$$M_{2n}(\boldsymbol{\beta}, \boldsymbol{\delta}) = \frac{1}{n} \sum_{i=1}^{n} \frac{\frac{\partial g_i^\top(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \boldsymbol{\delta}}{1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta})} + \boldsymbol{\nu}^\top \mathbf{C}(\boldsymbol{\beta}).$$

We first obtain their derivatives with respect to the three variables $\boldsymbol{\beta}, \boldsymbol{\delta}$ and $\boldsymbol{\nu}$.

$$\frac{\partial M_{1n}(\boldsymbol{\beta}, \boldsymbol{\delta})}{\partial \boldsymbol{\beta}^\top} = \frac{1}{n} \sum_{i=1}^{n} \frac{\frac{\partial g_i(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top}(1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta})) - g_i(\boldsymbol{\beta}) \boldsymbol{\delta}^\top \frac{\partial g_i(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top}}{(1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta}))^2},$$

$$\frac{\partial M_{1n}(\boldsymbol{\beta}, \boldsymbol{\delta})}{\partial \boldsymbol{\delta}^\top} = -\frac{1}{n} \sum_{i=1}^{n} \frac{g_i(\boldsymbol{\beta}) g_i^\top(\boldsymbol{\beta})}{(1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta}))^2}, \quad \frac{\partial M_{1n}(\boldsymbol{\beta}, \boldsymbol{\delta})}{\partial \boldsymbol{\nu}^\top} = 0,$$

$$\begin{aligned} \frac{\partial M_{2n}(\boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\nu})}{\partial \boldsymbol{\beta}^\top} = & \frac{1}{n} \sum_{i=1}^{n} \frac{\frac{\partial^2 g_i^\top(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top \partial \boldsymbol{\beta}} \boldsymbol{\delta}(1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta})) - \frac{\partial g_i^\top(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \boldsymbol{\delta} \boldsymbol{\delta}^\top \frac{\partial g_i(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top}}{(1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta}))^2} \\ & + \frac{\partial \mathbf{C}^\top(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top} \boldsymbol{\nu}, \end{aligned}$$

$$\frac{\partial M_{2n}(\boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\nu})}{\partial \boldsymbol{\delta}^\top} = \frac{1}{n} \sum_{i=1}^{n} \frac{\frac{\partial g_i^\top(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top} - \frac{\partial g_i^\top(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top} \boldsymbol{\delta} g_i^\top(\boldsymbol{\beta})}{(1 + \boldsymbol{\delta}^\top(\boldsymbol{\beta}) g_i(\boldsymbol{\beta}))^2}, \quad \frac{\partial M_{2n}(\boldsymbol{\beta}, \boldsymbol{\delta}, \boldsymbol{\nu})}{\partial \boldsymbol{\nu}^\top} = \mathbf{C}^\top(\boldsymbol{\beta}),$$

$$\frac{\partial H(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^\top} = \mathbf{C}(\boldsymbol{\beta}), \quad \frac{\partial H(\boldsymbol{\beta})}{\partial \boldsymbol{\delta}^\top} = 0, \quad \frac{\partial H(\boldsymbol{\beta})}{\partial \boldsymbol{\nu}^\top} = 0.$$

Hence, we have the following Taylor expansions of the system of equations at $(\boldsymbol{\beta}_0, 0, 0)$. Denote the solution to this equation system as $\left\{\tilde{\boldsymbol{\beta}}(z), \tilde{\boldsymbol{\delta}}(z), \tilde{\boldsymbol{\nu}}(z)\right\}$. Let $\Delta_n = \|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\| + \|\tilde{\boldsymbol{\delta}}\| + \|\tilde{\boldsymbol{\nu}}\|$.

$$\begin{aligned}
0 &= M_{1n}\left(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\delta}}, \tilde{\boldsymbol{\nu}}\right) \\
&= M_{1n}(\boldsymbol{\beta}_0, 0) + \frac{\partial M_{1n}(\boldsymbol{\beta}_0, 0)}{\partial \boldsymbol{\beta}^\top}\left(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\right) + \frac{\partial M_{1n}(\boldsymbol{\beta}_0, 0)}{\partial \boldsymbol{\delta}^\top}\left(\tilde{\boldsymbol{\delta}} - 0\right) \\
&\quad + \frac{\partial M_{1n}(\boldsymbol{\beta}_0, 0)}{\partial \boldsymbol{\nu}^\top}(\tilde{\boldsymbol{\nu}} - 0) + o_p(\Delta_n) \\
&= \frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0) + \frac{1}{n}\sum_{i=1}^{n}\frac{\partial g_i(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta}^\top}\left(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\right) - \frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0) g_i^\top(\boldsymbol{\beta}_0)\tilde{\boldsymbol{\delta}} + o_p(\Delta_n),
\end{aligned}$$

$$\begin{aligned}
0 &= M_{2n}\left(\tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\delta}}, \tilde{\boldsymbol{\nu}}\right) \\
&= M_{2n}(\boldsymbol{\beta}_0, 0, 0) + \frac{\partial M_{2n}(\boldsymbol{\beta}_0, 0, 0)}{\partial \boldsymbol{\beta}^\top}\left(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\right) + \frac{\partial M_{2n}(\boldsymbol{\beta}_0, 0, 0)}{\partial \boldsymbol{\delta}^\top}\left(\tilde{\boldsymbol{\delta}} - 0\right) \\
&\quad + \frac{\partial M_{2n}(\boldsymbol{\beta}_0, 0, 0)}{\partial \boldsymbol{\nu}^\top}(\tilde{\boldsymbol{\nu}} - 0) + o_p(\Delta_n) \\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial g_i^\top(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta}}\tilde{\boldsymbol{\delta}} + \mathbf{C}^\top(\boldsymbol{\beta}_0)\tilde{\boldsymbol{\nu}} + o_p(\Delta_n),
\end{aligned}$$

and $0 = H\left(\tilde{\boldsymbol{\beta}}\right) = H\left(\boldsymbol{\beta}_0\right) + \mathbf{C}^\top(\boldsymbol{\beta}_0)\left(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\right) + o_p(\Delta_n) = \mathbf{C}^\top(\boldsymbol{\beta}_0)\left(\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0\right) + o_p(\Delta_n)$. Putting the above equations into a matrix form, we obtain

$$\begin{pmatrix} -n^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0) + o_p(\Delta_n) \\ o_p(\Delta_n) \\ -H(\boldsymbol{\beta}_0) + o_p(\Delta_n) \end{pmatrix} = \boldsymbol{\Sigma}_n \begin{pmatrix} C_n^2 n^{-1}\tilde{\boldsymbol{\delta}} \\ \tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0 \\ \tilde{\boldsymbol{\nu}} \end{pmatrix}.$$

where

$$\boldsymbol{\Sigma}_n = \begin{pmatrix} -C_n^{-2}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0) g_i^\top(\boldsymbol{\beta}_0) & n^{-1}\sum_{i=1}^{n}\frac{\partial g_i(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta}^\top} & 0 \\ n^{-1}\sum_{i=1}^{n}\frac{\partial g_i^\top(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta}} & 0 & \mathbf{C}^\top(\boldsymbol{\beta}_0) \\ 0 & \mathbf{C}(\boldsymbol{\beta}_0) & 0 \end{pmatrix}.$$

Then we have $\boldsymbol{\Sigma}_n \xrightarrow{\mathcal{P}} \boldsymbol{\Sigma} = \begin{pmatrix} -\boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} & 0 \\ \boldsymbol{\Sigma}_{12} & 0 & \boldsymbol{\Sigma}_{23}^\top \\ 0 & \boldsymbol{\Sigma}_{23} & 0 \end{pmatrix}$, and $\boldsymbol{\Sigma}_{23} = \mathbf{C}(\boldsymbol{\beta}_0)$. By Proposition 6.1, it is easy to find that

$$\boldsymbol{\Sigma}_{11} = \sigma_0^2 \boldsymbol{\Omega}(z) f(z) \mu_{20}, \boldsymbol{\Sigma}_{12} = \boldsymbol{\Omega}(z) f(z). \tag{E3}$$

By the simple calculation, we have

$$\boldsymbol{\Sigma}^{-1} = \begin{pmatrix} -\boldsymbol{\Sigma}_{11}^{-1} + \boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\boldsymbol{\Upsilon}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1} & \boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\boldsymbol{\Upsilon} & \boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\mathbf{S}^\top \\ \boldsymbol{\Upsilon}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1} & \boldsymbol{\Upsilon} & \mathbf{S}^\top \\ \mathbf{S}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1} & \mathbf{S} & -\mathbf{R} \end{pmatrix},$$

where $\boldsymbol{\Upsilon} = \mathbf{V}\left(\mathbf{I} - \boldsymbol{\Sigma}_{23}^\top \mathbf{S}\right)$, $\mathbf{R} = \left(\boldsymbol{\Sigma}_{23}\mathbf{V}\boldsymbol{\Sigma}_{23}^\top\right)^{-1}$, $\mathbf{S} = \mathbf{R}\boldsymbol{\Sigma}_{23}\mathbf{V}$, and $\mathbf{V} = \left(\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\right)^{-1}$. Thus, we have the following

$$\begin{pmatrix} C_n^2 n^{-1}\tilde{\boldsymbol{\delta}} \\ \tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0 \\ \tilde{\boldsymbol{\nu}} \end{pmatrix} = \boldsymbol{\Sigma}^{-1} \begin{pmatrix} -n^{-1}\sum_{i=1}^n g_i(\boldsymbol{\beta}_0) \\ 0 \\ -H(\boldsymbol{\beta}_0) \end{pmatrix} + o_p(\Delta_n).$$

By this, under the local alternative hypothesis H_1, we could figure out that

$$\begin{aligned} \Delta_n &= \left\| \begin{pmatrix} \tilde{\boldsymbol{\delta}} \\ \tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0 \\ \tilde{\boldsymbol{\nu}} \end{pmatrix} \right\| \leq \left\| \begin{pmatrix} C_n^2 n^{-1}\tilde{\boldsymbol{\delta}} \\ \tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0 \\ \tilde{\boldsymbol{\nu}} \end{pmatrix} \right\| \\ &= \left\| \boldsymbol{\Sigma}^{-1} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \left\{ -\frac{1}{n}\sum_{i=1}^n g_i(\boldsymbol{\beta}_0) \right\} - \boldsymbol{\Sigma}^{-1} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} H(\boldsymbol{\beta}_0) + o_p(\Delta_n) \right\| \\ &\leq O_p\left(n^{-1/2}h^{-1/2}\right) + o_p(\Delta_n), \end{aligned}$$

which implies that $\Delta_n = O_p\left(n^{-1/2}h^{-1/2}\right)$.

Combining the above results, we have

$$\begin{pmatrix} C_n^2 n^{-1}\tilde{\boldsymbol{\delta}} \\ \tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0 \\ \tilde{\boldsymbol{\nu}} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\Sigma}_{11}^{-1} + \boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\boldsymbol{\Upsilon}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1} \\ \boldsymbol{\Upsilon}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1} \\ \mathbf{S}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1} \end{pmatrix} \left\{ -\frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0) \right\} - \begin{pmatrix} \boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\mathbf{S}^\top \\ \mathbf{S}^\top \\ -\mathbf{R} \end{pmatrix} H(\boldsymbol{\beta}_0) + o_p\left(n^{-1/2}h^{-1/2}\right). \tag{E4}$$

Given the following results $-\mathbf{S}\boldsymbol{\Sigma}_{12}^\top\boldsymbol{\Sigma}_{11}^{-1} = -\mathbf{R}\boldsymbol{\Sigma}_{23}\mathbf{V}\mathbf{V}^{-1}\boldsymbol{\Sigma}_{12}^{-1} = -\mathbf{R}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}$ we have the asymptotic expression for $\tilde{\boldsymbol{\nu}}$,

$$\begin{aligned} \tilde{\boldsymbol{\nu}} &= -\mathbf{S}\boldsymbol{\Sigma}_{12}^\top\boldsymbol{\Sigma}_{11}^{-1}\left\{\tfrac{1}{n}\textstyle\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} + \mathbf{R}H(\boldsymbol{\beta}_0) + o_p\left(n^{-1/2}h^{-1/2}\right) \\ &= -\mathbf{R}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\left\{\tfrac{1}{n}\textstyle\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} + \mathbf{R}H(\boldsymbol{\beta}_0) + o_p\left(n^{-1/2}h^{-1/2}\right). \end{aligned} \tag{E5}$$

By equation (E5), under the null hypothesis $H_0 : H\left\{\boldsymbol{\beta}_0(z)\right\} = 0$, we have

$$\tilde{\boldsymbol{\nu}} = n^{-1}\mathbf{R}^{1/2}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0) + o_p\left(n^{-1/2}h^{-1/2}\right).$$

Since

$$\begin{aligned} -\boldsymbol{\Upsilon}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1} &= -\boldsymbol{\Upsilon}\mathbf{V}^{-1}\boldsymbol{\Sigma}_{12}^{-1} = -\mathbf{V}(\mathbf{I} - \boldsymbol{\Sigma}_{23}^\top\mathbf{S})\mathbf{V}^{-1}\boldsymbol{\Sigma}_{12}^{-1} \\ &= -\boldsymbol{\Sigma}_{12}^{-1} + \mathbf{V}\boldsymbol{\Sigma}_{23}^\top\mathbf{S}\mathbf{V}^{-1}\boldsymbol{\Sigma}_{12}^{-1} = -\boldsymbol{\Sigma}_{12}^{-1} + \mathbf{V}\boldsymbol{\Sigma}_{23}^\top\mathbf{R}\boldsymbol{\Sigma}_{23}\mathbf{V}\mathbf{V}^{-1}\boldsymbol{\Sigma}_{12}^{-1} \\ &= -\boldsymbol{\Sigma}_{12}^{-1} + \mathbf{V}\boldsymbol{\Sigma}_{23}^\top\mathbf{R}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}, \end{aligned}$$

for the asymptotic expression of $\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0$, (E4) together with (E5) gives

$$\begin{aligned} \tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0 &= \left(-\boldsymbol{\Sigma}_{12}^{-1} + \mathbf{V}\boldsymbol{\Sigma}_{23}^\top\mathbf{R}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\right)\left\{\frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} + o_p\left(n^{-1/2}h^{-1/2}\right) \\ &= -\boldsymbol{\Sigma}_{12}^{-1}\left\{\frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} + \mathbf{V}\boldsymbol{\Sigma}_{23}^\top\mathbf{R}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\left\{\frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} \\ &\quad + o_p\left(n^{-1/2}h^{-1/2}\right) \end{aligned}$$

$$= \quad -\boldsymbol{\Sigma}_{12}^{-1}\left\{\frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} - \mathbf{V}\boldsymbol{\Sigma}_{23}^{\top}\tilde{\boldsymbol{\nu}} + o_p\left(n^{-1/2}h^{-1/2}\right).$$

Using the expression of $\tilde{\boldsymbol{\delta}}$

$$\begin{aligned}
\tilde{\boldsymbol{\delta}} &= \left\{n^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)g_i^{\top}(\boldsymbol{\beta}_0)\right\}^{-1}\left\{n^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} \\
&\quad + \left\{n^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)g_i^{\top}(\boldsymbol{\beta}_0)\right\}^{-1}\left\{n^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\frac{[\tilde{\boldsymbol{\delta}}^{\top}g_i(\boldsymbol{\beta}_0)]^2}{1+\tilde{\boldsymbol{\delta}}^{\top}g_i(\boldsymbol{\beta}_0)}\right\} \\
&= \left\{n^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)g_i^{\top}(\boldsymbol{\beta}_0)\right\}^{-1}\left\{n^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} + o_p\left(n^{-1/2}h^{-1/2}\right),
\end{aligned}$$

and the above asymptotic expression for $\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}_0$, the empirical log-likelihood ratio statistic can be written as

$$\begin{aligned}
2\ell(z) &= 2\sum_{i=1}^{n}\tilde{\boldsymbol{\delta}}^{\top}g_i\left(\tilde{\boldsymbol{\beta}}\right) - \sum_{i=1}^{n}\tilde{\boldsymbol{\delta}}^{\top}g_i\left(\tilde{\boldsymbol{\beta}}\right)g_i^{\top}\left(\tilde{\boldsymbol{\beta}}\right)\tilde{\boldsymbol{\delta}} + o_p(1) \\
&= 2n\left\{\frac{1}{n}\sum_{i=1}^{n} g_i^{\top}\left(\tilde{\boldsymbol{\beta}}\right)\right\}\tilde{\boldsymbol{\delta}} - n\tilde{\boldsymbol{\delta}}^{\top}\left\{\frac{1}{n}\sum_{i=1}^{n} g_i\left(\tilde{\boldsymbol{\beta}}\right)g_i^{\top}\left(\tilde{\boldsymbol{\beta}}\right)\right\}\tilde{\boldsymbol{\delta}} + o_p(1) \\
&= 2nh\left\{\frac{1}{n}\sum_{i=1}^{n} g_i^{\top}\left(\tilde{\boldsymbol{\beta}}\right)\right\}\boldsymbol{\Sigma}_{11}^{-1}\left(\frac{1}{n}\sum_{i=1}^{n} g_i\left(\tilde{\boldsymbol{\beta}}\right)\right) \\
&\quad - nh\left\{\frac{1}{n}\sum_{i=1}^{n} g_i\left(\tilde{\boldsymbol{\beta}}\right)\right\}^{\top}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{11}\boldsymbol{\Sigma}_{11}^{-1}\left\{\frac{1}{n}\sum_{i=1}^{n} g_i^{\top}\left(\tilde{\boldsymbol{\beta}}\right)\right\} + o_p(1) \\
&= nh\left\{\frac{1}{n}\sum_{i=1}^{n} g_i^{\top}\left(\tilde{\boldsymbol{\beta}}\right)\right\}\boldsymbol{\Sigma}_{11}^{-1}\left\{\frac{1}{n}\sum_{i=1}^{n} g_i^{\top}\left(\tilde{\boldsymbol{\beta}}\right)\right\} + o_p(1) \\
&= nh\tilde{\boldsymbol{\nu}}^{\top}\boldsymbol{\Sigma}_{23}\mathbf{V}\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}\mathbf{V}\boldsymbol{\Sigma}_{23}^{\top}\tilde{\boldsymbol{\nu}} + o_p(1) = nh\tilde{\boldsymbol{\nu}}^{\top}\mathbf{R}^{-1}\tilde{\boldsymbol{\nu}} + o_p(1).
\end{aligned}$$

We see that $E\left(\mathbf{R}^{1/2}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right) = 0$ and as $n \to \infty$,

$$\begin{aligned}
C_n^{-1}\text{Var}\left(\mathbf{R}^{1/2}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right) &\to \mathbf{R}^{1/2}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\boldsymbol{\Sigma}_{11}\boldsymbol{\Sigma}_{12}^{-1}\boldsymbol{\Sigma}_{23}^{\top}\mathbf{R}^{1/2} \\
&= \mathbf{R}^{1/2}\boldsymbol{\Sigma}_{23}(\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{11}\boldsymbol{\Sigma}_{12})^{-1}\boldsymbol{\Sigma}_{23}^{\top}\mathbf{R}^{1/2} \\
&= \mathbf{R}^{1/2}\boldsymbol{\Sigma}_{23}\mathbf{V}\boldsymbol{\Sigma}_{23}^{\top}\mathbf{R}^{1/2} \\
&= \mathbf{R}^{1/2}\mathbf{R}^{-1}\mathbf{R}^{1/2}
\end{aligned}$$

$$= \quad \mathbf{I}_{q\times q}.$$

Thus, by the Central Limit Theorem, under the null hypothesis $H_0 : H\{\boldsymbol{\beta}_0(z)\} = 0$, we have $n^{-1/2}h^{1/2}\mathbf{R}^{1/2}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0) \xrightarrow{d} N(\mathbf{0}, \mathbf{I}_q)$ which means

$$\sqrt{nh}\mathbf{R}^{-1/2}\tilde{\boldsymbol{\nu}} \xrightarrow{d} N(\mathbf{0}, \mathbf{I}_q).$$

Thus, $2\ell(z) \xrightarrow{d} \chi^2_q$. Under local alternative hypothesis $H_1 : H\{\boldsymbol{\beta}_0(z)\} = (nh)^{1/2}\mathbf{d}(z)$, we have

$$\sqrt{nh}\mathbf{R}^{-1/2}\tilde{\boldsymbol{\nu}} \xrightarrow{d} N(\mathbf{R}^{1/2}\mathbf{d}, \mathbf{I}_q).$$

Thus, $2\ell(z) = nh\tilde{\boldsymbol{\nu}}^\top\mathbf{R}^{-1}\tilde{\boldsymbol{\nu}} + o_p(1) \xrightarrow{d} \chi^2_q(\mathbf{d}^\top\mathbf{R}\mathbf{d})$. □

E.6 Proof of Theorem 6.3

Proof. We first prove the asymptotic normality of D_n under the null hypothesis.

We have the decomposition for $D_n = D_{n1} + D_{n2}$, where

$$\begin{aligned}
D_{n1} &= C_n^{-2}\sum_{i=1}^{n}\int_a^b \boldsymbol{\xi}_i^\top(z)\mathbf{G}^\top(z)\mathbf{G}(z)\boldsymbol{\xi}_i(z)w(z)dz \\
&= C_n^{-2}\sum_{i=1}^{n}\epsilon_i^2\int_a^b K_h(Z_i - z)^2\mathbf{X}_i^\top\mathbf{G}^\top(z)\mathbf{G}(z)\mathbf{X}_i w(z)dz, \\
D_{n2} &= C_n^{-2}\sum_{i=1}^{n}\sum_{k\neq i}^{n}\int_a^b \boldsymbol{\xi}_i^\top(z)\mathbf{G}^\top(z)\mathbf{G}(z)\boldsymbol{\xi}_k(z)w(z)dz \\
&= C_n^{-2}\sum_{i=1}^{n}\sum_{k\neq i}^{n}\epsilon_i\epsilon_k\int_a^b K_h(Z_i - z)K_h(Z_k - z)\mathbf{X}_i^\top\mathbf{G}^\top(z)\mathbf{G}(z)\mathbf{X}_k w(z)dz.
\end{aligned}$$

Note that,

$$\begin{aligned}
E\left\{\mathbf{X}_i^\top\mathbf{G}^\top(z)\mathbf{G}(z)\mathbf{X}_i\middle|Z_i = z\right\} &= tr(\mathbf{G}(z)\boldsymbol{\Omega}(z)\mathbf{G}^\top(z)) \\
&= tr\left[(\sigma^2\mu_{20}f(z))^{-1}\boldsymbol{I}_{q\times q}\right] = q(\sigma^2\mu_{20}f(z))^{-1},
\end{aligned}$$

and $ED_{n2} = 0$. We also have

$$\begin{aligned} ED_{n1} &= h\sigma^2 \int_a^b E\left\{K_h(Z_i - z)^2 \mathbf{X}_i^\top \mathbf{G}^\top(z)\mathbf{G}(z)\mathbf{X}_i\right\} w(z)dz \\ &= h\sigma^2 \int_a^b E\left\{E\left[\mathbf{X}_i^\top \mathbf{G}^\top(z)\mathbf{G}(z)\mathbf{X}_i \middle| Z_i\right] K_h(Z_i - z)^2\right\} w(z)dz \\ &= \sigma^2 \int_a^b \left[E\left\{\mathbf{X}_i^\top \mathbf{G}^\top(z)\mathbf{G}(z)\mathbf{X}_i \middle| Z_i = z\right\} f(z)\mu_{20} + O(h^2)\right] w(z)dz \\ &= q + O(h^2). \end{aligned}$$

Define $K^{(4)}(x) = \int K^2(y)K^2(y-x)\,dy$ and $I_{ik}(z) = \mathbf{X}_i^\top \mathbf{G}^\top(z)\mathbf{G}(z)\mathbf{X}_k$.

Thus, $\mathbf{X}_i^\top \mathbf{G}^\top(z_1)\mathbf{G}(z_1)\mathbf{X}_k\mathbf{X}_{i'}^\top \mathbf{G}^\top(z_2)\mathbf{G}(z_2)\mathbf{X}_{k'} = I_{ik}(z_1) \times I_{i'k'}(z_2)$. When $i \neq k$, we have

$$\begin{aligned} & E(I_{ii}(z_1)I_{kk}(z_2)\big|Z_i = z_1, Z_k = z_2) \\ = \;& E(I_{ii}(z_1)\big|Z_i = z_1)E(I_{kk}(z_2)\big|Z_k = z_2) \\ = \;& tr(E\left[\mathbf{G}^\top(z_1)\mathbf{X}_i^\top \mathbf{X}_i \mathbf{G}(z_1)\middle|Z_i = z_1\right]) \\ & \times tr(E\left[\mathbf{G}^\top(z_2)\mathbf{X}_k^\top \mathbf{X}_k \mathbf{G}(z_2)\middle|Z_k = z_2\right]) \\ = \;& q^2\mu_{20}^{-2}\sigma^{-4}f^{-1}(z_1)f^{-1}(z_2). \end{aligned}$$

Thus, we have

$$\begin{aligned} & ED_{n1}^2 \\ = \;& h^2 n^{-2} \sum_{i=1}^n \sum_{i'=1}^n \int_a^b \int_a^b E\left\{\boldsymbol{\xi}_{i1}^\top \mathbf{G}^\top(z_1)\mathbf{G}(z_1)\boldsymbol{\xi}_{i1}\boldsymbol{\xi}_{i'2}^\top \mathbf{G}^\top(z_2)\mathbf{G}(z_2)\boldsymbol{\xi}_{i'2}\right\} \\ & \times w(z_1)w(z_2)dz_1dz_2 \\ = \;& \sum_{i' \neq i}^n h^2 n^{-2} \int_a^b \int_a^b E\left\{\epsilon_i^2 \epsilon_{i'}^2 K_h(Z_i - z_1)^2 K_h(Z_{i'} - z_2)^2 I_{ii}(z_1)I_{i'i'}(z_2)\right\} \\ & \times w(z_1)w(z_2)dz_1dz_2 \\ & + \sum_{i=1}^n h^2 n^{-2} \int_a^b \int_a^b E\left\{\epsilon_i^4 K_h(Z_i - z_1)^2 K_h(Z_i - z_2)^2 I_{ii}(z_1)I_{ii}(z_2)\right\} \\ & \times w(z_1)w(z_2)dz_1dz_2 \\ = \;& \sum_{i' \neq i}^n n^{-2}\sigma^4 \int_a^b \int_a^b \left\{E(I_{ii}(z_1) \times I_{kk}(z_2)\big|Z_i = z_1, Z_k = z_2)f(z_1)f(z_2)\mu_{20}^2\right. \end{aligned}$$

$$
\begin{aligned}
&+O(h^2)\big\} \times w(z_1)w(z_2)dz_1dz_2 \\
&+\sum_{i=1}^{n} n^{-2}E\epsilon_i^4 \int_a^b \int_a^b \big\{hE(I_{ii}(z_1) \times I_{ii}(z_2)\big|Z_i = z_1\big)\, f(z_1)K^{(4)}\left(\frac{z_2 - z_1}{h}\right) \\
&+o(h^2)\big\} \times w(z_1)w(z_2)dz_1dz_2 \\
\asymp\ & q^2 + O\left(h^2\right).
\end{aligned}
$$

Next, we calculate ED_{n2}^2. Note that

$$
\begin{aligned}
& ED_{n2}^2 \\
=\ & h^2n^{-2}\sum_{i=1}^{n}\sum_{i'=1}^{n}\sum_{k=1}^{n}\sum_{k'=1}^{n}\int_a^b\int_a^b E\left\{\boldsymbol{\xi}_{i1}^\top\mathbf{G}^\top(z_1)\mathbf{G}(z_1)\boldsymbol{\xi}_{k1}\boldsymbol{\xi}_{i'2}^\top\mathbf{G}^\top(z_2)\mathbf{G}(z_2)\boldsymbol{\xi}_{k'2}\right\} \\
& \times w(z_1)w(z_2)dz_1dz_2.
\end{aligned}
$$

When $k \neq i$ and $k' \neq i'$, $E\left(\epsilon_i\epsilon_{i'}\epsilon_k\epsilon_{k'}\right) \neq 0$ only in two cases, the first one is $\{i = i', k = k'\}$, and the second one is $\{i = k', k = i'\}$. In particularly, we have $ED_{n2}^2 = h^2n^{-2}\sigma^4\sum_{i=1}^n\sum_{k\neq i}^n$ $\int_a^b\int_a^b(\Pi_1 + \Pi_2)w(z_1)w(z_2)dz_1dz_2$, where

$$
\begin{aligned}
\Pi_1 &= E\left\{I_{ik}(z_1)I_{ik}(z_2)K_h(Z_i - z_1)K_h(Z_k - z_1)K_h(Z_i - z_2)K_h(Z_k - z_2)\right\} \\
&= E\left\{E\left[I_{ik}(z_1)I_{ik}(z_2)\big|Z_i, Z_k\right]K_h(Z_i - z_1)K_h(Z_k - z_1)\right. \\
&\quad \left. K_h(Z_i - z_2)K_h(Z_k - z_2)\right\} \\
&= \int_a^b\int_a^b E\left[I_{ik}(z_1)I_{ik}(z_2)\big|Z_i = x, Z_k = y\right]f(x)f(y) \\
&\quad \times K_h(x - z_1)K_h(x - z_2)K_h(y - z_1)K_h(y - z_2)dxdy \\
&= \int_a^b\left\{h^{-1}E\left[I_{ik}(z_1)I_{ik}(z_2)\big|Z_i = z_1, Z_k = y\right]f(z_1)K^{(2)}\left(\frac{z_2 - z_1}{h}\right)\right. \\
&\quad \left. +O(h)\right\} \times f(y)K_h(y - z_1)K_h(y - z_2)dy \\
&= h^{-2}E\left[I_{ik}(z_1)I_{ik}(z_2)\big|Z_i = z_1, Z_k = z_2\right]f(z_1)f(z_2)\left\{K^{(2)}\left(\frac{z_2 - z_1}{h}\right)\right\}^2 \\
&\quad +O(1)
\end{aligned}
$$

and

$$
\Pi_2 = E\left\{I_{ik}(z_1)I_{ki}(z_2)K_h(Z_i - z_1)K_h(Z_k - z_1)K_h(Z_i - z_2)K_h(Z_k - z_2)\right\}
$$

$$
\begin{aligned}
&= E\left\{E\left[I_{ik}(z_1)I_{ki}(z_2)\middle|Z_i,Z_k\right]K_h(Z_i-z_1)K_h(Z_k-z_1)K_h(Z_i-z_2)\right.\\
&\quad\left.\times K_h(Z_k-z_2)\right\}\\
&= \int_a^b\int_a^b E\left[I_{ik}(z_1)I_{ki}(z_2)\middle|Z_i=x,Z_k=y\right]\\
&\quad\times f(x)f(y)K_h(x-z_1)K_h(x-z_2)K_h(y-z_1)K_h(y-z_2)dxdy
\end{aligned}
$$

$$
\begin{aligned}
&= \int_a^b\left\{h^{-1}E\left[I_{ik}(z_1)I_{ki}(z_2)\middle|Z_i=z_1,Z_k=y\right]f(z_1)K^{(2)}\left(\frac{z_2-z_1}{h}\right)+O(h)\right\}\\
&\quad\times f(y)K_h(y-z_1)K_h(y-z_2)dy\\
&= h^{-2}E\left[I_{ik}(z_1)I_{ki}(z_2)\middle|Z_i=z_1,Z_k=z_2\right]f(z_1)f(z_2)\left\{K^{(2)}\left(\frac{z_2-z_1}{h}\right)\right\}^2\\
&\quad+O(1).
\end{aligned}
$$

Since

$$
\begin{aligned}
&E\left[I_{ik}(z_1)I_{ik}(z_2)\middle|Z_i=z_1,Z_k=z_2\right]\\
&= E\left\{E\left[\mathbf{X}_i^\top\mathbf{G}^\top(z_1)\mathbf{G}(z_1)\mathbf{X}_k\mathbf{X}_i^\top\mathbf{G}^\top(z_2)\mathbf{G}(z_2)\mathbf{X}_k\right]|Z_i=z_1,Z_k=z_2\right\}\\
&= E\left\{\mathbf{X}_k^\top\mathbf{G}^\top(z_1)\mathbf{G}(z_1)E\left[\mathbf{X}_i\mathbf{X}_i^\top\right]\mathbf{G}^\top(z_2)\mathbf{G}(z_2)\mathbf{X}_k|Z_i=z_1,Z_k=z_2\right\}\\
&= tr\left\{E\left[\mathbf{X}_k^\top\mathbf{G}^\top(z_1)\mathbf{G}(z_1)E\left(\mathbf{X}_i\mathbf{X}_i^\top\right)\mathbf{G}^\top(z_2)\mathbf{G}(z_2)\mathbf{X}_k|Z_i=z_1,Z_k=z_2\right]\right\}\\
&= tr\left\{E\left[\mathbf{G}(z_1)E\left(\mathbf{X}_i\mathbf{X}_i^\top\right)\mathbf{G}^\top(z_2)\mathbf{G}(z_2)\mathbf{X}_k\mathbf{X}_k^\top\mathbf{G}^\top(z_1)|Z_i=z_1,Z_k=z_2\right]\right\}\\
&= \mathbf{G}(z_1)E\left(\mathbf{X}_i\mathbf{X}_i^\top|Z_i=z_1\right)\mathbf{G}^\top(z_2)\mathbf{G}(z_2)E\left(\mathbf{X}_k\mathbf{X}_k^\top|Z_k=z_2\right)\mathbf{G}^\top(z_1)\\
&= \mathbf{G}(z_1)\mathbf{\Omega}(z_1)\mathbf{G}^\top(z_2)\mathbf{G}(z_2)\mathbf{\Omega}(z_2)\mathbf{G}^\top(z_1)\asymp q,
\end{aligned}
$$

and similarly, we have $E\left[I_{ik}(z_1)I_{ki}(z_2)\middle|Z_i=z_1,Z_k=z_2\right]\asymp q$. Combining the results for Π_1 and Π_2, we have

$$
\begin{aligned}
ED_{n2}^2 &\asymp \sigma^4 h\int\int f(v)f(v+hu)w(v)w(v+hu)\left(K^{(2)}(u)\right)^2dudv\\
&= h\int\left(K^{(2)}(u)\right)^2du\int f^2(v)w^2(v)dv+O(h^2),
\end{aligned}
$$

Hence we have $Var(D_{n1}) = o(Var(D_{n2}))$, and it follows that

$$D_n - E(D_n) = D_{n2}\left\{1 + o(1)\right\}.$$

We can write D_{n2} as $D_{n2} = \frac{1}{n}\sum_{i\neq k}^{n}\int_a^b \mathcal{Z}_i^\top(z)\mathcal{Z}_k(z)w(z)dz$, where $\mathcal{Z}_i(z) = \sqrt{h}\mathbf{G}(z)\boldsymbol{\xi}_i(z)$. Let $\mathcal{U}_n = \frac{1}{n-1}D_{n2} = \frac{2}{n(n-1)}\sum_{1\leq i<k\leq n}\mathcal{K}(\mathcal{Z}_i, \mathcal{Z}_k)$, where $\mathcal{K}(\mathcal{Z}_i, \mathcal{Z}_k) = \int_a^b \mathcal{Z}_i^\top(z)\mathcal{Z}_k(z)w(z)dz$. Define $A_\mathcal{K}$ as $A_\mathcal{K}g(x) = \int_\infty^\infty \mathcal{K}(x,y)g(y)dF(y)$, where F is the distribution of $\mathcal{Z}_i$. Then we have the associated eigenvalues and eigenfunctions, denoted as $\{\lambda_k, \psi_k\}_{k=1}^\infty$. The remaining proof for D_n under the null hypothesis test is analogous to the sparse case in Theorem 2 and Corollary 1 in [114].

Secondly, we prove the asymptotic distribution for D_n under alternative hypothesis. Notice that $2\ell(z) = nh\tilde{\boldsymbol{\nu}}^\top\mathbf{R}^{-1}\tilde{\boldsymbol{\nu}} + o_p(1)$ as shown in Theorem 6.2 and by (E5) under local alternative,

$$\tilde{\boldsymbol{\nu}} = -\mathbf{R}\boldsymbol{\Sigma}_{23}\boldsymbol{\Sigma}_{12}^{-1}\left\{\frac{1}{n}\sum_{i=1}^{n} g_i(\boldsymbol{\beta}_0)\right\} + \mathbf{R}H(\boldsymbol{\beta}_0) + o_p\left(n^{-1/2}h^{-1/2}\right).$$

The remaining proof is the same as the sparse case in Theorem 3 in [114]. □