

Studies on semi-parametric varying coefficient models

by

Liying Jin

B.S., Chongqing University of Posts and Telecommunications, China, 2015

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2022

Abstract

In many economic and geographic studies, we may have spatially referenced covariates providing information about the spatial distribution that impacts the response variable. The spatial varying coefficient model (SVCМ) has been an effective tool for exploring such information by modeling spatial nonstationarity. In this thesis, we study the SVCМ and address several challenges in estimating the varying coefficient functions over complex domains in different scenarios.

In chapter 2, we consider a new class of semi-parametric regression models called the generalized partially linear spatially varying coefficient model (GPLSVCM). We propose using the bivariate penalized spline over triangulation (BPST) method to approximate the coefficient functions and employing a quasi-likelihood maximization to obtain model estimators. The proposed method can handle data distributed over arbitrarily shaped domains with complex boundaries and interior holes. We prove the consistency of the estimators under some regularity conditions. Additionally, we propose a model selection procedure via BIC that can accurately identify the covariates with constant and varying effects.

In chapter 3, we introduce a new R package **GPLSVCM**, which integrates model structure identification, variable selection, model fitting, and predictive inference for GPLSVCMs. To account for high-dimensional data, we propose a doubly penalized approach for simultaneous variable selection and model structure identification. The proposed method can efficiently remove irrelevant covariates while detecting constant and varying components of the coefficients. To quantify the uncertainty in a single prediction, we propose three resampling-based methods for constructing prediction intervals that attain target coverage probability. Compared with existing R packages for SVCМs, **GPLSVCM** is more flexible and computationally cheaper, so it can be widely applied in spatial data analysis over any arbitrarily shaped domain.

In chapter 4, we develop a new volatility model by allowing spatially varying coefficients in spatial GARCH models. This model captures volatility behaviors over space and investigates the relationship between some explanatory variables and the volatility at each location. A two-stage quasi-likelihood maximization via BPST is developed to estimate the model over a complicated domain. For each chapter, we conduct both simulation studies and real-data applications to demonstrate the performance of our approach.

Studies on semi-parametric varying coefficient models

by

Liying Jin

B.S., Chongqing University of Posts and Telecommunications, China, 2015

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2022

Approved by:

Major Professor
Jingru Mu

Copyright

© Liying Jin 2022.

Abstract

In many economic and geographic studies, we may have spatially referenced covariates providing information about the spatial distribution that impacts the response variable. The spatial varying coefficient model (SVCМ) has been an effective tool for exploring such information by modeling spatial nonstationarity. In this thesis, we study the SVCМ and address several challenges in estimating the varying coefficient functions over complex domains in different scenarios.

In chapter 2, we consider a new class of semi-parametric regression models called the generalized partially linear spatially varying coefficient model (GPLSVCM). We propose using the bivariate penalized spline over triangulation (BPST) method to approximate the coefficient functions and employing a quasi-likelihood maximization to obtain model estimators. The proposed method can handle data distributed over arbitrarily shaped domains with complex boundaries and interior holes. We prove the consistency of the estimators under some regularity conditions. Additionally, we propose a model selection procedure via BIC that can accurately identify the covariates with constant and varying effects.

In chapter 3, we introduce a new R package **GPLSVCM**, which integrates model structure identification, variable selection, model fitting, and predictive inference for GPLSVCMs. To account for high-dimensional data, we propose a doubly penalized approach for simultaneous variable selection and model structure identification. The proposed method can efficiently remove irrelevant covariates while detecting constant and varying components of the coefficients. To quantify the uncertainty in a single prediction, we propose three resampling-based methods for constructing prediction intervals that attain target coverage probability. Compared with existing R packages for SVCМs, **GPLSVCM** is more flexible and computationally cheaper, so it can be widely applied in spatial data analysis over any arbitrarily shaped domain.

In chapter 4, we develop a new volatility model by allowing spatially varying coefficients in spatial GARCH models. This model captures volatility behaviors over space and investigates the relationship between some explanatory variables and the volatility at each location. A two-stage quasi-likelihood maximization via BPST is developed to estimate the model over a complicated domain. For each chapter, we conduct both simulation studies and real-data applications to demonstrate the performance of our approach.

Table of Contents

List of Figures	xi
List of Tables	xiii
Acknowledgements	xiv
1 Introduction	1
2 Generalized Partially Linear Spatially Varying Coefficient Models	6
2.1 Introduction	6
2.2 The Estimation Method	9
2.2.1 Triangulations	10
2.2.2 Bivariate Splines	10
2.2.3 Penalized Quasi-likelihood Estimator	12
2.2.4 Asymptotic Results	16
2.3 Model Selection	19
2.3.1 The GPLSVCM-GIC Algorithm	19
2.4 Simulation Study	20
2.4.1 Simulation Study: Estimation	21
2.4.2 Simulation Study: Model Selection	27
2.5 Application to Texas Crash Data	27
2.6 Concluding Remarks	31
3 An R Package: GPLSVCM	32
3.1 Introduction	32

3.2	Model Fitting	35
3.2.1	Bivariate Penalized Spline over Triangulation	35
3.3	Predictive Inference	37
3.3.1	Prediction Intervals	38
3.3.2	Simulation Study 1 (Prediction Interval)	40
3.4	Model Selection	42
3.4.1	Simultaneous Variable and Model Structure Selection	42
3.4.2	Simulation Study 2 (Model Selection)	45
3.5	Application to NYC Crime Data	47
3.6	Concluding Remarks	55
4	Spatial Generalized Autoregressive Conditional Heteroskedasticity Varying Coefficient Models	56
4.1	Introduction	56
4.2	The Estimation Method	59
4.2.1	Model Representation	59
4.2.2	Bivariate Penalized Spline over Triangulation	60
4.2.3	Estimation of λ , ρ and $\beta(\cdot)$	63
4.2.4	Estimation of α	64
4.3	Simulation Studies	65
4.3.1	Simulation Study 1	66
4.3.2	Simulation Study 2	69
4.4	Application to Florida House Price Data	73
4.5	Concluding Remarks	78
5	Conclusion	79
	Bibliography	81

A	Theoretical Results with Details for Chapter 2	88
A.1	Properties of Penalized Quasi-likelihood Estimators	88
A.2	Consistency of Penalized Quasi-likelihood Estimators in the GPLSVCM . . .	96
B	R Code for Chapter 3	103

List of Figures

2.1	Contour plots for the true coefficient functions and plots of triangulation 1 and 2 on the horseshoe domain.	22
2.2	Contour plots of estimated coefficient functions of BPST method and GGWR method under Poisson and Negative Binomial (NB) model when $n=1000$. . .	24
2.3	Contour plots of estimated coefficient functions of BPST method and GGWR method under Poisson and Negative Binomial (NB) model when $n=2000$. . .	25
2.4	Contour plots of estimated coefficient functions of BPST method under Poisson and Negative Binomial (NB) model when $n=5000$	26
2.5	Plots of triangulation 1, 2 and 3 on the Texas domain.	30
2.6	Contour plots of the estimated coefficient surfaces on the Texas domain. . . .	30
3.1	Data analysis process of GPLSVCM	36
3.2	Simulation results showing the empirical coverage rates of the predictive interval for all methods based on different distributions and sample sizes. . . .	41
3.3	Felony Crimes within census block group in NYC.	49
3.4	Plot of triangulation on the domain of NYC.	50
3.5	Contor plots of the estimated coefficient functions on the domain of NYC . .	54
4.1	Simulation Study 1: the true coefficient function and two triangulations. . .	66
4.2	Simulation Study 1: contour plots for the estimated coefficient functions for $n = 1000$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$	68

4.3	Simulation Study 1: contour plots for the estimated coefficient functions for $n = 500$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$	69
4.4	Simulation Study 2: the true coefficient function and two triangulations. . .	70
4.5	Simulation Study 2: contour plots for the estimated coefficient functions for $n = 1000$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$	72
4.6	Simulation Study 2: contour plots for the estimated coefficient functions for $n = 500$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$	73
4.7	The estimated volatility of the log return of the median house price in 2019 and 202 using two-stage estimation based on BPST, referred to as “BPST” and the least square estimation based method, referred to as “LSE”.	76
4.8	The estimated coefficient functions of land area in 2019 and 2020 using two-stage estimation based on BPST; $\triangle$ is the triangulation over the domain of Florida.	77

List of Tables

2.1	Model Estimation for the simulation study.	23
2.2	Proportion of $\widehat{\mathcal{I}}_c$ (estimated set of constant coefficients) selected among 200 replications.	27
2.3	Description of the explanatory variables in the 2017 Texas crash dataset. . .	28
2.4	Model Selection, Estimation and Prediction for Texas Car Crash Data. . . .	29
3.1	Results of Model Structure Identification.	47
3.2	Results of CV-MSPE.	48
3.3	Description of the covariates in the 2016 NYC Felony Crime dataset.	49
3.4	Model Selection, Estimation, and Prediction for NYC Felony Crime Data. . .	52
4.1	Estimation results for simulation study 1.	67
4.2	Estimation results for simulation study 2.	71
4.3	Estimation results for Florida House Price Data.	75

Acknowledgments

First of all, I would like to offer my most heartfelt gratitude and thanks to my Ph.D. advisor, Dr. Jingru Mu, for all of her assistance and support throughout the duration of my doctoral studies. Dr. Jingru Mu is an inspiring mentor who has a lot of brilliant ideas. The communications that we had always provided me ideas to keep going forward in my research. Her knowledge, expertise, and enthusiasm toward research have set as a good example for me to follow in my future career.

I would also like to express my appreciation to the members of my Ph.D. committee, Dr. Christopher Vahl, Dr. Michael Higgins and Dr. Max Lu, for their support, advice, and insightful comments that were helpful throughout the course of my research. In addition, I would like to say thank you to the department's faculty members for providing such high-quality educational opportunities. I also want to express my gratitude to the whole department for offering me graduate assistantship while I was studying here. Without the financial support, I wouldn't be able to complete my Ph.D. study.

Moreover, great thanks to all my friends at K-State, who helped and accompanied me when I went through some difficulties, comforted and enlightened me when I was anxious. Last but not least, I would like to thank my family, particularly my parents and boyfriend, for the unconditional love and support they have shown me throughout my life.

Chapter 1

Introduction

In spatial statistics, situations where the response variable is not normally distributed arise frequently. [Nelder and Wedderburn \(1972\)](#) proposed the generalized linear model (GLM) that provides a regression relationship between the response variable and predictors for various response variables. However, in most georeferenced studies, the GLM cannot display the spatially varying component within the data. As a commonly used nonparametric regression tool, the varying coefficient model (VCM) enjoys much more flexibility by allowing the regression coefficients to change over different groups characterized by certain covariates. See [Hastie and Tibshirani \(1993\)](#), [Huang et al. \(2004\)](#), [Fan et al. \(2005\)](#), [Yang et al. \(2006\)](#), [Fan and Zhang \(2008\)](#) and [Park et al. \(2015\)](#). In the spatial context, [Brunsdon et al. \(1996\)](#) developed the spatial varying coefficient model (SVCMM), which assumes all the regression coefficients vary from location to location. The SVCMM is particularly well-suited for exploring the spatial nonstationarity inherent in the data, but two of its drawbacks are progressively becoming apparent. One is that it cannot be used with noncontinuous response variables, such as count and binary data, and another is that it cannot take into account scenarios in which some covariates have homogeneous effects while others have heterogeneous effects across locations. [Mu et al. \(2020\)](#) proposed the spatial autoregressive partially linear varying coefficient models (SAR-PLVCM) that include both constant and spatially varying coefficients. If the neighbor effects are ignored, the model becomes the PLSVCM.

PLSVCs are more efficient than SVCs when linear coefficients are present, but their applicability is severely limited by the requirement of a continuous response variable. [Kim and Wang \(2020\)](#) proposed the generalized spatial varying coefficient models (GSVC) based on SVCs, which apply to all response variables. However, it does not address the second issue with SVCs simultaneously.

To facilitate the development of a more flexible spatial regression model, we herein consider a new class of semiparametric regression models as a more generalized version of GSVC. Given observations $\{(\mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i, Y_i)\}_{i=1}^n$, we propose the generalized partially linear spatial varying coefficient models (GPLSVC):

$$g\{E(Y_i | \mathbf{Z} = \mathbf{Z}_i, \mathbf{X} = \mathbf{X}_i, \mathbf{U} = \mathbf{U}_i)\} = \sum_{l=1}^{p_1} X_{il}\beta_l + \sum_{k=1}^{p_2} Z_{ik}\alpha_k(\mathbf{U}_i), i = 1, \dots, n$$

where $g(\cdot)$ is the link function, $\mathbf{U}_i \in \mathbb{R}^2$, β_l 's are unknown constant coefficients and $\alpha_k(\cdot)$'s are unknown coefficient functions.

Early works in estimating the SVC include the Kernel-based approach, see [Brunsdon et al. \(1996\)](#), [Fotheringham et al. \(2003\)](#) and the Bayesian approach, see [Assunção \(2003\)](#), [Gelfand et al. \(2003\)](#). On the one hand, both approaches are extremely time-consuming for large datasets. On the other hand, they can only deal with rectangular domains. Researchers often encounter large datasets dispersed over irregularly shaped domains with complex boundaries and holes. Any Euclidean distance-based method would suffer from the 'leakage' problem over such a complicated domain.

Recently, [Mu et al. \(2018\)](#) has proposed to use the bivariate splines over triangulations (BPST) method to estimate SVCs over arbitrarily shaped spatial domains, whose computation just takes a few seconds for a sample size at the thousands level. Initially proposed by [Lai and Schumaker \(2007\)](#), the BPST method is proved to be computationally efficient and theoretically reliable in estimating spatially varying coefficient functions over arbitrarily shaped domains. Other applications of the BPST method in nonparametric and semi-parametric regression also confirm its strength in handling complicated domains, see [Lai and Wang \(2013\)](#), [Zhou and Pan \(2014\)](#), [Wang et al. \(2016\)](#) and [Yu et al. \(2020\)](#).

In this dissertation, we propose to approximate the coefficient functions in GPLSVCM by the BPST method and develop a maximum likelihood estimation approach to estimate the unknown parameters simultaneously. We offer to use BPST for two reasons : (i) it achieves remarkable computational efficiency while handling complex domains; (ii) it is a global smoothing method that allows for convenient implementation in a quasi-likelihood maximization procedure with flexible degrees and various smoothness. One obstacle with the implementation is that one usually does not know which covariate has spatially varying effects and which does not. We therefore have developed a backward elimination algorithm based on the Bayesian information criterion (BIC) to identify the covariates with constant and varying coefficients, respectively.

To advance the application of GPLSVCM in real data analysis, building an R package that integrates model selection, model estimation, and predictive inferences of GPLSVCMs could be very helpful. As there are increasing challenges from high-dimensional data, it would be necessary for this R package to provide a model selection procedure to select the model structure while identifying the irrelevant covariates. By removing a larger number of redundant covariates, the selected model can promote the efficiency of model estimation. In previous studies, the variable selection techniques for semiparametric regression models have been largely studied in the literature; see [Liang and Li \(2009\)](#), [Xie and Huang \(2009\)](#), [Liu et al. \(2011\)](#), [Lian \(2012\)](#) and [Lian et al. \(2014\)](#). Specifically for VCMs, [Li et al. \(2020\)](#) has proposed a penalized method to identify the model structure and significant covariates altogether. The penalized approach is usually high-speed and sufficiently efficient for model selection. However, no such penalized approach can be directly applied to GPLSVCM. Another convenience we aim to offer in this package is a prediction interval to evaluate the uncertainty associated with the point prediction of a future observation. The additional information from a prediction interval could help decision-makers develop effective policies.

We develop a doubly penalized approach based on adaptive group lasso to address the difficulties of achieving the aforementioned model selection. We also devised three algorithms, the Jackknife, Jackknife+, and CV+, respectively, to compute the prediction intervals for GPLSVCMs. Along with the model estimation, the model selection and the prediction

intervals are incorporated into a new R package **GPLSVCM** we have created.

Volatility is a terminology to describe the dispersion of a quantity's error term in statistics. It has significant implications for timely risk management in practice. There has been a considerable amount of research into modeling volatility, the majority of which has been devoted to determining if and how volatility varies over time, see [Schwert \(1989\)](#), [Duan \(1995\)](#), [Andersen et al. \(1999\)](#), [Jacquier et al. \(2002\)](#) and [Mishra et al. \(2010\)](#). For example, the GARCH family of models and Stochastic Volatility (SV) models have acquired widespread acceptance for assessing the volatility of time series data from financial markets and economic growth. Volatility clustering and spillovers over space, frequently observed in reality, such as in real estate markets and oil price markets, have recently attracted the attention of researchers. A class of spatial GARCH models and their extensions have been well developed to characterize those volatility behaviors over space. They serve as valuable tools to capture market interdependencies for practitioners. In addition to looking at how volatility is related spatially, it may also be useful to look at how some explanatory variables affect volatility across space, which remains a lack of exploration in the literature. In order to fill this gap, we develop a new semiparametric spatial model to simultaneously analyze spatial volatility and its association with certain covariates in space.

The rest of the dissertation is structured as follows. In Chapter [2](#), we demonstrate our quasi-likelihood maximization procedure based on bivariate penalized splines over triangulation (BPST) to estimate GPLSVCM and provide the asymptotic properties of the proposed estimators. We also show the backward elimination algorithm for model structure selection. Two simulation studies and analysis of a Texas car crash frequency dataset are conducted as illustrative examples of model estimation and selection. In Chapter [3](#), we introduce a new R package we create for the real data application of GPLSVCMs. We provide an explicit explanation of the methodology behind the main functions in the package, such as the algorithms for computing prediction intervals and the doubly penalized approach via adaptive group lasso for model selection. Simulation studies are conducted to evaluate each functionality's performance in this package. The sample R codes are offered for each evaluation as guidance for users of this package. Moreover, a New York City crime dataset is used to

illustrate an application of this package. In Chapter 4, we present a new model for analyzing spatial volatility called the spatial generalized autoregressive conditional heteroskedasticity varying coefficient (SGARCH-VCM) model. We show our proposed two-stage method based on two types of quasi-likelihood and BPST to obtain the estimated volatility and the coefficient functions. We conduct two simulation studies to examine our proposed method's performance over rectangular and horseshoe-shaped domains. Additionally, a Florida house price dataset is analyzed using our approach to compare the estimated volatility all over the area in different years. In chapter 5, we draw a general conclusion about our work with some remarks and future directions. Finally, the technical proof of the theoretical results in Chapter 2 and the sample R codes from Chapter 3 are given in the Appendix.

Chapter 2

Generalized Partially Linear Spatially Varying Coefficient Models

2.1 Introduction

In recent years, the spatially varying coefficient model (SVC) has been proved to be a powerful tool for investigating the nonstationarity of a regression relationship over space, see [Fotheringham et al. \(2003\)](#), [Mu et al. \(2018\)](#), [Kim and Wang \(2020\)](#), [Mu et al. \(2020\)](#). In SVCs, the regression coefficients are modeled as a bivariate function of spatial coordinates. As a result, it assumes all the regression coefficients change with locations. In practice, however, some covariates may have constant effects across the spatial domain while others have varying effects across space. If we mistakenly neglect the constant effects of some covariates, the efficiency of the model estimation will be compromised due to an unexpected higher variance of the estimators.

To accommodate for different data types and allow more flexibility at the same time, in this article, we propose a generalized partially linear spatially varying coefficient model (GPLSVC). For $i = 1, \dots, n$, let Y_i be the response variable, $\mathbf{X}_i = (X_{i1}, \dots, X_{ip_1})^\top$ be the explanatory variables that have constant effects on the response variable and $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{ip_2})^\top$ be the explanatory variables that have varying effects on the response vari-

ables at location $\mathbf{U}_i = (U_{i1}, U_{i2})^\top$ where $\mathbf{U}_i$ is the location of the i -th observation distributed on an arbitrarily shaped bounded domain $\Omega \subseteq \mathbb{R}^2$. $\{(Y_i, \mathbf{X}_i, \mathbf{Z}_i, \mathbf{U}_i)\}_{i=1}^n$ are independently and identically distributed samples of size n . For this model, we narrow our scope to the distributions from exponential families, for example, Poisson distribution and negative binomial distribution. To be specific, denote $\mathbf{T} = (\mathbf{X}, \mathbf{Z}, \mathbf{U})$ and $\mathbf{t} = (\mathbf{x}, \mathbf{z}, \mathbf{u})$, we assume the conditional density of Y given $\mathbf{T} = \mathbf{t}$ is:

$$f_{Y|\mathbf{T}}(y|\mathbf{t}) = \exp \left[\frac{1}{\sigma^2} \{y\xi(\mathbf{t}) - \mathcal{B}\{\xi(\mathbf{t})\}\} + \mathcal{C}(y, \sigma^2) \right] \quad (2.1)$$

where $\mathcal{B}(\cdot)$ and $\mathcal{C}(\cdot)$ are known functions, ξ and σ^2 are the canonical parameter and dispersion parameter, respectively. Let $\mu(\mathbf{t}) = \mathbb{E}[Y|\mathbf{T}]$, we model $\mu(\mathbf{t})$ through a link function $g(\cdot)$ by considering the following form:

$$g\{\mu(\mathbf{t})\} = \mathbf{x}^\top \boldsymbol{\beta} + \mathbf{z}^\top \boldsymbol{\alpha}(\mathbf{u}) = \sum_{l=1}^{p_1} x_l \beta_l + \sum_{k=1}^{p_2} z_k \alpha_k(\mathbf{u}) \quad (2.2)$$

where $\boldsymbol{\beta} = (\beta_1, \dots, \beta_{p_1})^\top$ is a vector of unknown constant coefficient parameters and $\boldsymbol{\alpha}(\mathbf{u}) = \{\alpha_1(\mathbf{u}), \dots, \alpha_{p_2}(\mathbf{u})\}^\top$ is a vector of bivariate smooth coefficient functions to be estimated. The conditional variance of Y given $\mathbf{T} = \mathbf{t}$ satisfies that $\text{Var}(Y|\mathbf{T} = \mathbf{t}) = \sigma^2 V\{\mu(\mathbf{t})\}$ for some known variance function V .

To estimate the proposed model in (2.1) and (2.2), one challenge is to estimate the coefficient functions $\alpha_k(\cdot)$'s computationally efficiently. There are mainly three approaches for estimating the varying coefficient functions in SVCMs. The first is the Bayesian approach, see [Gelfand et al. \(2003\)](#) and [Assunção \(2003\)](#). In the Bayesian framework, the coefficients' spatial dependency is modeled by a stochastic process. With each coefficient function being assigned a prior distribution, the varying coefficient estimators can be obtained by Gibbs sampling from a posterior distribution. While the Bayesian approach is straightforward and credible, it requires high computational cost, especially for models with many coefficient functions. Besides, people usually lack knowledge of the prior distribution, and an inappropriate one easily yields misleading results. Another popular approach is local smoothing.

For example, the geographically weighted regression (GWR) method proposed by [Fotheringham et al. \(2003\)](#) and the profile likelihood based estimation proposed by [Sun et al. \(2014\)](#). The GWR approach exploits a weighting mechanism for each data point that is dependent on its location in geographical space, and it requires a weighted least square estimation at each data point. Similarly, the profile likelihood-based approach is to maximize the local likelihood at each data point. Though the local smoothing method enjoys very nice theoretical properties, it could be computationally prohibitive for a sample size of thousands. A competing approach proposed by [Mu et al. \(2018\)](#) is the bivariate spline basis expansion, where they approximated the coefficient functions with bivariate splines. This approach is more computationally expedient since it inherits the benefits of spline-based techniques.

Another challenge in the estimation is handling the irregular spatial domain with complex shapes and interior holes, which is not uncommon in practice. The conventional smoothing tools, such as the local smoothing method, usually involve calculating the Euclidean distance between data points. Consequently, it will result in a poor estimation over difficult regions due to an inappropriate linking of the data points on the domain. This so-called ‘leakage’ problem across a complex domain has been extensively discussed in [Ramsay \(2002\)](#) and [Wood et al. \(2008a\)](#). For partly linear models, [He and Shi \(1996\)](#) proposed a bivariate tensor product of the B-spline method to estimate their functional components. This method is remarkable for its computational efficiency, but it requires data distributed on a rectangular domain.

In this article, to estimate GPLSVCMs for spatial datasets collected from irregularly shaped domains, we propose to use bivariate penalized splines over triangulations (BPST) method to approximate the coefficient functions $\alpha_k(\cdot)$ for $k = 1, 2, \dots, p_2$ and employ a quasi-likelihood maximization procedure to obtain the estimators of the coefficient functions as well as the constant coefficients. We show in this article that the proposed GPLSVCM estimators solve the ‘leakage’ problems across complicated domains and contribute to more accurate estimation and prediction without computational burdens compared with other competitors through numerical studies. To support our methodology, we prove the consistency of both the linear coefficient estimators and the coefficient function estimators under some regularity

conditions.

A critical obstacle when implementing GPLSVCs is identifying which coefficients are constant and which are spatially varying since people usually have little information about the true model structure. Model selection techniques are therefore imperative for real data analysis of GPLSVCs. [Zhang et al. \(2010\)](#) proposed employing the generalized information criterion (GIC) to select the regularization parameter for variable selection based on nonconcave penalized likelihood approach. In this article, we borrow the idea of GIC and propose a model selection procedure based on backward elimination via GIC to determine the functional and constant components of the coefficients in GPLSVCs. The excellent model selection and estimation accuracy that we see in our simulations lends credence to the validity of our approach.

The rest of the article is structured as follows. In [Section 2.2](#), assuming we know the true model structure, we introduce our estimation method and present the consistency of the proposed estimators. [Section 2.3](#) describes the algorithm for model selection and its implementation. In [Section 2.4](#), we conduct simulation studies to evaluate the finite sample performance of the proposed methods. Furthermore, in [Section 2.5](#), we illustrate an application of our method to a Texas crash dataset. Proof of the results is provided in the [Appendix A](#).

2.2 The Estimation Method

We estimate the model in [\(2.1\)](#) and [\(2.2\)](#) through a quasi-likelihood maximization based on the bivariate penalized splines over triangulations (BPST) method. In this section, we introduce the idea of triangulations, bivariate splines, and the penalized quasi-likelihood estimators.

2.2.1 Triangulations

Triangulation is accredited with handling spatial data distributed over irregular domains with complex boundaries and interior holes. A “good” triangulation usually refers to those with well-shaped triangles with no small or obtuse angles (Yu, Wang, Wang, Liu, and Yang, 2020). In the following, we summarize the idea of triangulation.

For an arbitrarily shaped polygon domain Ω , consider a collection of N triangles $\Delta = \{\tau_1, \dots, \tau_N\}$ that covers the domain Ω . According to Zhou and Pan (2014), this collection Δ is called a triangulation of $\Omega = \cup_{j=1}^N \tau_j$ provided that if a pair of triangles in Δ intersect, then their intersection is either a common vertex or a common edge. Let $|\tau|$ be the longest edge length of a triangle $\tau \in \Delta$ and denote the size of Δ as $|\Delta| := \max\{|\tau|, \tau \in \Delta\}$, that is, the length of the longest edge of Δ . Let ρ_τ be the radius of the largest disk that can be inscribed in τ and define the shape parameter τ as $\beta_\tau = |\tau|/\rho_\tau$. Note that when β_τ is small, the triangles are relatively uniform in the sense that all angles of triangles in the triangulation τ are relatively the same; see Lai and Schumaker (2007).

Based on the “max-min” criterion suggested by Lai and Schumaker (2007), one should maximize the minimum angle of all the triangles in the triangulation when generating the triangulation. In practice, many software packages are available for constructing a triangulation, most of which follow the Delaunay algorithm. For example, the `delaunay.m` and `distmesh2d.m` in MATLAB, and the R package **Triangulation** developed by Lai and Wang (2019).

2.2.2 Bivariate Splines

In this section, we present the main idea of bivariate splines. For detailed discussions, we refer to Lai and Schumaker (2007) and Zhou and Pan (2014).

For a non-degenerate triangle τ , write $\tau = \langle \mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3 \rangle$ where $\mathbf{u}_1, \mathbf{u}_2$, and $\mathbf{u}_3$ are its vertices with Cartesian coordinates $\mathbf{u}_i = \{u_{i1}, u_{i2}\}$ for $i = 1, 2, 3$. Then for any point $\mathbf{u} \in \mathbb{R}^2$, $\mathbf{u}$ can be uniquely represented in the form $\mathbf{u} = b_1 \mathbf{u}_1 + b_2 \mathbf{u}_2 + b_3 \mathbf{u}_3$ where b_1, b_2 and b_3 are the barycentric coordinates of the point $\mathbf{u}$ relative to the triangle τ with $b_1 + b_2 + b_3 = 1$. The

Bernstein polynomials of degree d relative to triangle τ is defined as $B_{ijk}^{\tau,d}(\mathbf{u}) = \frac{d!}{i!j!k!} b_1^i b_2^j b_3^k$.

Let $\mathbb{P}_d(\tau)$ be the space of Bernstein polynomials of degree less than or equal to d defined on the triangle τ . Then for any function $s \in \mathbb{P}_d(\tau)$, the B-form of s relative to the triangle τ is defined as $s|_{\tau} = \sum_{i+j+k=d} \gamma_{ijk}^{\tau} B_{ijk}^{\tau,d}$ with $i+j+k=d$, and $\boldsymbol{\gamma}_{\tau} = \{\gamma_{ijk}^{\tau}, i+j+k=d\}$ are called the B-coefficients of s . Given a triangulation Δ , define the space of piece-wise polynomials of degree d on Δ as

$$\mathbb{S}_d(\Delta) = \{s : s|_{\tau_i} \in \mathbb{P}_d(\tau_i), i = 1, \dots, N\}.$$

For a positive integer r , denote $\mathbb{C}^r(\Delta)$ as the collection of r th differentiable function on Δ , then the spline space of degree d and smoothness r on Δ is defined as

$$\mathbb{S}_d^r(\Delta) = \mathbb{S}_d(\Delta) \cap \mathbb{C}^r(\Delta) = \{s \in \mathbb{C}^r(\Delta) : s|_{\tau_i} \in \mathbb{P}_d(\tau_i), i = 1, \dots, N\}.$$

Let $\mathbb{S} = \mathbb{S}_{3r+2}^r(\Delta)$ for a fixed smoothness $r \geq 1$, then such a spline space has the optimal approximation order (rate of convergence) for noise-free datasets; see [Lai and Schumaker \(2007\)](#).

For $k = 1, \dots, p_2$, let $\mathcal{J}_k$ be the index set of all Bernstein basis polynomials for the triangulation Δ_k with cardinality $|\mathcal{J}_k| = N_k(d+1)(d+2)/2$ where N_k is the number of triangles in Δ_k . We write $\{B_{kj}\}_{j \in \mathcal{J}_k}$ as the set of bivariate Bernstein basis polynomials from $\mathbb{S}_d^r(\Delta_k)$ for $k = 1, \dots, p_2$. Then for a bivariate function $\psi_k(\mathbf{u}) \in \mathbb{S}_d^r(\Delta_k)$, we can represent it with the basis expansion as follows:

$$\psi_k(\mathbf{u}) = \sum_{j \in \mathcal{J}_k} B_{kj}(\mathbf{u}) \gamma_{kj} = \mathbf{B}_k(\mathbf{u})^{\top} \boldsymbol{\gamma}_k \quad (2.3)$$

where $\mathbf{B}_k(\mathbf{u}) = \{B_{kj}(\mathbf{u}), j \in \mathcal{J}_k\}^{\top}$ is the vector of bivariate basis functions evaluated at the location $\mathbf{u}$ and $\boldsymbol{\gamma}_k = \{\gamma_{kj}, j \in \mathcal{J}_k\}^{\top}$ is the corresponding vector of spline coefficients. To satisfy the smoothness conditions of two adjacent triangles in the triangulation Δ_k , we use a constraint matrix $\mathbf{H}_k$ that depends on the smoothness parameter r and the structure of Δ_k

to restrict the spline coefficient γ_k in (2.3) for $k = 1, \dots, p_2$, that is, we require γ_k satisfies $\mathbf{H}_k \gamma_k = 0$ where $\mathbf{H}_k$ expresses the information of the smoothness conditions across all the shared edges of triangles in the triangulation Δ_k for $k = 1, \dots, p_2$.

2.2.3 Penalized Quasi-likelihood Estimator

For the proposed model in (2.1) and (2.2), we use the BPST method to approximate $\alpha_k(\cdot)$, $k = 1, \dots, p_2$, in which the smoothness penalty term is defined as the energy function:

$$\mathcal{E}(\alpha) = \int_{\Omega} \left\{ (D_{u_1}^2 \alpha)^2 + 2 (D_{u_1} D_{u_2} \alpha)^2 + (D_{u_2}^2 \alpha)^2 \right\} du_1 du_2,$$

where $D_{u_j}^q \alpha(\mathbf{u})$ is the q th order derivative in the direction $u_j, j = 1, 2$ at location $\mathbf{u} = (u_1, u_2)^\top$. Denote $\ell_q(\mu, y)$ as the quasi-likelihood function of the GPLSVC, note that $\frac{\partial}{\partial \mu} \ell_q(\mu, y) = \frac{y - \mu}{\sigma^2 V(\mu)}$. Given a data set $\{(\mathbf{X}_i, \mathbf{Z}_i, \mathbf{U}_i, Y_i)\}_{i=1}^n$, to obtain the estimators for β and α in (2.2), we consider to maximize the penalized quasi-likelihood function as follows:

$$\sum_{i=1}^n \ell_q \left[g^{-1} \left\{ \sum_{l=1}^{p_1} X_{il} \beta_l + \sum_{k=1}^{p_2} Z_{ik} \psi_k(\mathbf{U}_i) \right\}, Y_i \right] - \frac{1}{2} \sum_{k=1}^{p_2} \lambda_k \mathcal{E}(\psi_k) \quad (2.4)$$

where $\psi_k \in \mathbb{S}_d^r(\Delta_k)$ and λ_k is the non-negative smoothness penalty parameter that regularize the spline fit of ψ_k for $k = 1, \dots, p_2$.

According to the introduction of B-form bivariate spline in Section 2.2.2, one can use different triangulations, sets of Bernstein basis polynomials and constraint matrices for the approximation of $\psi_k, k = 1, \dots, p_2$. Without loss of generality, we assume that $\Delta_k = \Delta$, $\mathbf{B}_k(\mathbf{U}_i) = \mathbf{B}(\mathbf{U}_i) = \{B_j(\mathbf{U}_i), j \in \mathcal{J}\}^\top$ and $\mathbf{H}_k = \mathbf{H}$ for $i = 1, \dots, n$ and $k = 1, \dots, p_2$. Denote $\mathbf{B}_k = \mathbf{B} = \{\mathbf{B}(\mathbf{U}_1), \dots, \mathbf{B}(\mathbf{U}_n)\}^\top$ as the evaluation matrix of Bernstein basis polynomials, then maximizing the penalized quasi-likelihood function in (2.4) is equivalent

to the following minimization problem:

$$\min_{\boldsymbol{\beta}, \boldsymbol{\gamma}} \left\{ -\sum_{i=1}^n \ell_q \left[g^{-1} \left\{ \sum_{l=1}^{p_1} X_{il} \beta_l + \sum_{k=1}^{p_2} Z_{ik} \mathbf{B}(\mathbf{U}_i)^\top \boldsymbol{\gamma}_k \right\}, Y_i \right] + \frac{1}{2} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^\top \mathbf{P} \boldsymbol{\gamma}_k \right\} \quad \text{subject to } \mathbf{H} \boldsymbol{\gamma}_k = \mathbf{0}. \quad (2.5)$$

where $\boldsymbol{\beta} = (\beta_1, \dots, \beta_{p_1})^\top$, $\boldsymbol{\gamma} = (\boldsymbol{\gamma}_1^\top, \dots, \boldsymbol{\gamma}_{p_2}^\top)^\top$ and $\mathbf{P}$ is the block diagonal penalty matrix that satisfies $\mathcal{E}(\mathbf{B}^\top \boldsymbol{\gamma}_k) = \boldsymbol{\gamma}_k^\top \mathbf{P} \boldsymbol{\gamma}_k$

To solve the minimization problem in (2.5), we first remove the constraint by a QR decomposition of $\mathbf{H}^\top$. To be specific, through a QR decomposition of $\mathbf{H}^\top$, one has:

$$\mathbf{H}^\top = \mathbf{Q} \mathbf{R} = (\mathbf{Q}_1 \ \mathbf{Q}_2) \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \end{pmatrix},$$

where $\mathbf{Q}$ is an orthogonal matrix with a sub-matrix $\mathbf{Q}_1$ consisting of the first r columns of $\mathbf{Q}$ where r is the rank of matrix $\mathbf{H}$, $\mathbf{R}$ is an upper triangle matrix and $\mathbf{R}_2$ is a matrix of zeros. Then $\mathbf{H} \boldsymbol{\gamma}_k = \mathbf{0}$ can be guaranteed by a reparameterization of $\boldsymbol{\gamma}_k = \mathbf{Q}_2 \boldsymbol{\gamma}_k^*$ for some $\boldsymbol{\gamma}_k^*$. In this way, the constrained minimization problem in (2.5) can be converted to the following unconstrained minimization problem:

$$\min_{\boldsymbol{\beta}, \boldsymbol{\gamma}^*} \left\{ -\sum_{i=1}^n \ell_q \left[g^{-1} \left\{ \sum_{l=1}^{p_1} X_{il} \beta_l + \sum_{k=1}^{p_2} Z_{ik} \mathbf{B}(\mathbf{U}_i)^\top \mathbf{Q}_2 \boldsymbol{\gamma}_k^* \right\}, Y_i \right] + \frac{1}{2} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^{*\top} \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2 \boldsymbol{\gamma}_k^* \right\} \quad (2.6)$$

where $\boldsymbol{\gamma}^* = (\boldsymbol{\gamma}_1^{*\top}, \dots, \boldsymbol{\gamma}_{p_2}^{*\top})^\top$.

To solve the minimization problem in (2.6), we develop a penalized iteratively reweighted least square algorithm (IRLS). Let $\mathbf{Y} = (Y_1, \dots, Y_n)^\top$ be the vector of the response variable, $\mathbf{X}_i = (X_{i1}, \dots, X_{ip_1})^\top$ be the vector of the linear covariates for i th observation and $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{ip_2})^\top$ be the vector of the nonlinear covariates for i th observation. Denote $\boldsymbol{\mathcal{X}} = (\mathbf{X}_1, \dots, \mathbf{X}_n)^\top$ and $\boldsymbol{\mathcal{Z}}_B = (\boldsymbol{\mathcal{Z}}_{B,1}, \dots, \boldsymbol{\mathcal{Z}}_{B,n})^\top$ where $\boldsymbol{\mathcal{Z}}_{B,i} = \mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)$ and $\mathbf{B}^*(\mathbf{U}_i) = \mathbf{Q}_2^\top \mathbf{B}(\mathbf{U}_i)$. Besides, denote $\boldsymbol{\eta} = (\eta_1, \dots, \eta_n)^\top$ where $\eta_i = \mathbf{X}_i^\top \boldsymbol{\beta} + \boldsymbol{\mathcal{Z}}_{B,i}^\top \boldsymbol{\gamma}^*$ and $\boldsymbol{\mu}(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) = (\mu_1, \dots, \mu_n)^\top$ where $\mu_i = g^{-1}(\eta_i)$. Furthermore, let $\mathbf{G}, \mathbf{V}$ be diagonal matrices with diagonal

elements $g'(\mu_i)$ and $V(\mu_i)$ for $i = 1, \dots, n$, respectively. Denote the weight matrix as $\mathbf{W}$ with diagonal elements $\{V(\mu_i) g'(\mu_i)^2\}^{-1}$ for $i = 1, \dots, n$.

Suppose we have $\mathbf{G}^{(j)}$, $\mathbf{V}^{(j)}$ and $\mathbf{W}^{(j)}$ at the j th iteration, we consider the following objective function at $(j+1)$ th iteration:

$$L_q^{(j+1)} = \left\| \{\mathbf{V}^{(j)}\}^{-1/2} \{\mathbf{Y} - \boldsymbol{\mu}(\boldsymbol{\beta}, \boldsymbol{\gamma}^*)\} \right\|^2 + \frac{1}{2} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^{*\top} \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2 \boldsymbol{\gamma}_k^*$$

Note that $\boldsymbol{\mu}(\boldsymbol{\beta}, \boldsymbol{\gamma}^*)$ can be approximated by its first order Taylor expansion around $(\boldsymbol{\beta}^{(j)}, \boldsymbol{\gamma}^{*(j)})$

$$\boldsymbol{\mu}(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) \approx \boldsymbol{\mu}^{(j)} - \{\mathbf{G}^{(j)}\}^{-1} (\boldsymbol{\mathcal{X}} \ \boldsymbol{\mathcal{Z}}_B) \left\{ \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\gamma}^* \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}^{(j)} \\ \boldsymbol{\gamma}^{*(j)} \end{pmatrix} \right\}$$

where $\boldsymbol{\mu}^{(j)} = \boldsymbol{\mu}(\boldsymbol{\beta}^{(j)}, \boldsymbol{\gamma}^{*(j)})$. Therefore,

$$\begin{aligned} L_q^{(j+1)} &\approx \left\| \{\mathbf{V}^{(j)}\}^{-1/2} \left[\mathbf{Y} - \boldsymbol{\mu}^{(j)} - \{\mathbf{G}^{(j)}\}^{-1} (\boldsymbol{\mathcal{X}} \ \boldsymbol{\mathcal{Z}}_B) \left\{ \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\gamma}^* \end{pmatrix} - \begin{pmatrix} \boldsymbol{\beta}^{(j)} \\ \boldsymbol{\gamma}^{*(j)} \end{pmatrix} \right\} \right] \right\|^2 \\ &\quad + \frac{1}{2} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^{*\top} \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2 \boldsymbol{\gamma}_k^* \\ &= \left\| \{\mathbf{V}^{(j)}\}^{-1/2} \{\mathbf{G}^{(j)}\}^{-1} [\mathbf{G}^{(j)} (\mathbf{Y} - \boldsymbol{\mu}^{(j)}) + \boldsymbol{\eta}^{(j)} - \boldsymbol{\mathcal{X}} \boldsymbol{\beta} - \boldsymbol{\mathcal{Z}}_B \boldsymbol{\gamma}^*] \right\|^2 \\ &\quad + \frac{1}{2} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^{*\top} \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2 \boldsymbol{\gamma}_k^* \\ &= \left\| \{\mathbf{W}^{(j)}\}^{1/2} (\tilde{\mathbf{Y}}^{(j)} - \boldsymbol{\mathcal{X}} \boldsymbol{\beta} - \boldsymbol{\mathcal{Z}}_B \boldsymbol{\gamma}^*) \right\|^2 + \frac{1}{2} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^{*\top} \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2 \boldsymbol{\gamma}_k^* \end{aligned} \quad (2.7)$$

where $\tilde{\mathbf{Y}}^{(j)} = (\tilde{Y}_1^{(j)}, \dots, \tilde{Y}_n^{(j)})^\top$ and $\tilde{Y}_i^{(j)} = g'(\mu_i^{(j)})(Y_i - \mu_i^{(j)}) + \eta_i^{(j)}$. We present the penalized iteratively reweighted least square algorithm (IRLS) in Algorithm 1, where the convergence threshold is based on δ and δ is set as $\left\| (\boldsymbol{\beta}^{(j)\top}, \boldsymbol{\gamma}^{*(j)\top})^\top - (\boldsymbol{\beta}^{(j-1)\top}, \boldsymbol{\gamma}^{*(j-1)\top})^\top \right\|_2$. In the numerical studies, we set $\mu_i^{(0)} = Y_i + 0.001$.

Algorithm 1 : Penalized IRLS

Data: Observations $\{\mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i, Y_i\}_{i=1}^n$

Result: $(\boldsymbol{\beta}^{(j)}, \boldsymbol{\gamma}^{*(j)})$

- 1 Initialize: Set starting values to $\boldsymbol{\mu}^{(0)} = \left\{ \mu_i^{(0)} \right\}_{i=1}^n$, then calculate $g'(\mu_i^{(0)})$ and $V(\mu_i^{(0)})$ for $i = 1, \dots, n$ to obtain $\mathbf{W}^{(0)} = \text{diag} \left[\left\{ V(\mu_i^{(0)}) g'(\mu_i^{(0)})^2 \right\}^{-1} \right]_{i=1}^n$ and $\tilde{\mathbf{Y}}^{(0)} = \left\{ g'(\mu_i^{(0)})(Y_i - \mu_i^{(0)}) + \eta_i^{(0)} \right\}_{i=1}^n$
 - 2 Set j to 0 and δ to 1. Minimize the objective function in (2.7) with respect to $(\boldsymbol{\beta}^\top, \boldsymbol{\gamma}^{*\top})^\top$ and set $(\boldsymbol{\beta}^{(0)\top}, \boldsymbol{\gamma}^{*(0)\top})^\top$ to the solution
 - 3 **while** $\delta > \text{threshold}$ **do**
 - 4 (i) set j to $j + 1$
 - 5 (ii) Update $\boldsymbol{\eta}^{(j)} = \boldsymbol{\eta}(\boldsymbol{\beta}^{(j-1)}, \boldsymbol{\gamma}^{*(j-1)})$ and $\boldsymbol{\mu}^{(j)} = g^{-1}(\boldsymbol{\eta}^{(j)})$ to obtain $g'(\mu_i^{(j)})$ and $V(\mu_i^{(j)})$ for $i = 1, \dots, n$, then calculate $\mathbf{W}^{(j)}$ and $\tilde{\mathbf{Y}}^{(j)}$
 - 6 (iii) Minimize the objective function in (2.7) with respect to $(\boldsymbol{\beta}^\top, \boldsymbol{\gamma}^{*\top})^\top$ and set $(\boldsymbol{\beta}^{(j)\top}, \boldsymbol{\gamma}^{*(j)\top})^\top$ to the solution
 - 7 (iv) calculate δ
-

To choose the penalty parameter $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_{p_2})^\top$, we use the generalized cross validation criterion (GCV). Denote $\mathbf{D}_\Lambda = \Lambda \otimes \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2$ where $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{p_2})$, and $\mathbf{S}(\boldsymbol{\lambda}) = (\boldsymbol{\mathcal{X}} \ \mathbf{Z}_B) \mathbf{M}^{-1} (\boldsymbol{\mathcal{X}} \ \mathbf{Z}_B)^\top \mathbf{W}$ as the hat matrix where

$$\mathbf{M} = \begin{pmatrix} \boldsymbol{\mathcal{X}}^\top \mathbf{W} \boldsymbol{\mathcal{X}} & \boldsymbol{\mathcal{X}}^\top \mathbf{W} \mathbf{Z}_B \\ \mathbf{Z}_B^\top \mathbf{W} \boldsymbol{\mathcal{X}} & \mathbf{Z}_B^\top \mathbf{W} \mathbf{Z}_B + \mathbf{D}_\Lambda \end{pmatrix}.$$

Let $\tilde{\mathbf{Y}} = (\tilde{Y}_1, \dots, \tilde{Y}_n)^\top$ where $\tilde{Y}_i = g'(\hat{\mu}_i)(Y_i - \hat{\mu}_i) + \hat{\eta}_i$. We select the penalty parameter $\boldsymbol{\lambda}$ by minimize the following GCV criterion:

$$\text{GCV}(\boldsymbol{\lambda}) = \frac{n^{-1} \left\| \mathbf{W}^{1/2} \{ \tilde{\mathbf{Y}} - \mathbf{S}(\boldsymbol{\lambda}) \tilde{\mathbf{Y}} \} \right\|^2}{[n^{-1} \text{tr}\{\mathbf{I} - \mathbf{S}(\boldsymbol{\lambda})\}]^2} \quad (2.8)$$

over a grid of values of $\boldsymbol{\lambda}$

Let $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\gamma}}^*$ be the minimizer of (2.6) where $\widehat{\boldsymbol{\beta}} = (\widehat{\beta}_1, \dots, \widehat{\beta}_{p_1})^\top$ and $\widehat{\boldsymbol{\gamma}}^* = (\widehat{\gamma}_1^{*\top}, \dots, \widehat{\gamma}_{p_2}^{*\top})^\top$, then the estimator of $\boldsymbol{\beta}$ is $\widehat{\boldsymbol{\beta}}$. Let $\widehat{\boldsymbol{\gamma}}_k = \mathbf{Q}_2 \widehat{\boldsymbol{\gamma}}_k^* = \{\widehat{\gamma}_{kj}, j \in \mathcal{J}\}^\top$ for $k = 1, \dots, p_2$, then the bivariate spline estimator of $\alpha_k(\mathbf{u})$ is

$$\widehat{\alpha}_k(\mathbf{u}) = \mathbf{B}(\mathbf{u})^\top \widehat{\boldsymbol{\gamma}}_k = \sum_{j \in \mathcal{J}} B_j(\mathbf{u}) \widehat{\gamma}_{kj}, \text{ for } k = 1, \dots, p_2.$$

2.2.4 Asymptotic Results

This section studies the asymptotic properties of the proposed estimators, assuming we know the true model structure before estimation.

Notations

We first introduce some general notations. Let $A(\Omega)$ be the area of the domain Ω . Without loss of generality, we assume $A(\Omega) = 1$ in the rest of the paper. For a real value vector $\mathbf{a} \in \mathbb{R}^n$, we define $\|\mathbf{a}\|^2 = \sum_{i=1}^n a_i^2$ and $|\mathbf{a}| = \max_{1 \leq i \leq n} |a_i|$ as its Euclidean norm and supremum norm, respectively. For symmetric matrix $\mathbf{A} = (a_{ij})_{i=1, j=1}^{m, n}$, we denote $\lambda_{\max}(\mathbf{A})$ and $\lambda_{\min}(\mathbf{A})$ as its largest and smallest eigenvalues, $\|\mathbf{A}\|_2 = \max_{\mathbf{a} \in \mathbb{R}^n, \mathbf{a} \neq \mathbf{0}} \|\mathbf{A}\mathbf{a}\|_2 \|\mathbf{a}\|_2^{-1}$ as its L_2 norm. For any Lebesgue measurable function $\psi(\mathbf{u})$ on a domain $\mathcal{D}$ where $\mathcal{D} = [0, 1], [0, 1]^p, \Omega \subseteq \mathbb{R}^2$ or $[0, 1]^p \times \Omega$, let $\|\psi\|_\infty = \sup_{\mathbf{u} \in \mathcal{D}} |\psi(\mathbf{u})|$ and $\|\psi\|_{L_2}^2 = \int_{\mathcal{D}} \psi^2(\mathbf{u}) d\mathbf{u}$.

For any bivariate function $f : \Omega \rightarrow \mathbb{R}$, denote $|f|_{v, \infty, \Omega} = \max_{i+j=v} \|D_{u_1}^i D_{u_2}^j f(\mathbf{u})\|_{\infty, \Omega}$ where v is a nonnegative integer. Let

$$\mathcal{W}^{d+1, \infty}(\Omega) = \{f : |f|_{k, \infty, \Omega} < \infty, 0 \leq k \leq d+1\}$$

be the standard Sobolev space. Define the model space as

$$\mathcal{M} = \left\{ \eta : \eta(\mathbf{t}) = \sum_{l=1}^{p_1} x_l \beta_l + \sum_{k=1}^{p_2} z_k \alpha_k(\mathbf{u}) : \beta_l \in \mathbb{R}, \mathbf{u} \in \Omega, \alpha_k \in \mathcal{W}^{d+1, \infty}(\Omega) \right\}.$$

For functions $\eta_1, \eta_2 \in \mathcal{M}$, we define their theoretical inner product and empirical inner product as $\langle \eta_1, \eta_2 \rangle = E\{\eta_1(\mathbf{T})\eta_2(\mathbf{T})\}$ and $\langle \eta_1, \eta_2 \rangle_n = \frac{1}{n} \sum_{i=1}^n \eta_1(\mathbf{T}_i)\eta_2(\mathbf{T}_i)$ respectively. Then for a function $\eta \in \mathcal{M}$, denote $\|\eta\| = \sqrt{\langle \eta, \eta \rangle}$ and $\|\eta\|_n = \sqrt{\langle \eta, \eta \rangle_n}$.

For a vector valued function $\boldsymbol{\psi} = (\psi_1, \dots, \psi_{p_2})^\top$, let $\|\boldsymbol{\psi}\|_{L_2} = \{\sum_{k=1}^{p_2} \|\psi_k\|_{L_2}^2\}^{1/2}$ and $\|\boldsymbol{\psi}\|_\infty = \max_{1 \leq k \leq p_2} \|\psi_k\|_\infty$. For $\boldsymbol{\psi}^{(1)}(\mathbf{u}) = \{\psi_1^{(1)}(\mathbf{u}), \dots, \psi_{p_2}^{(1)}(\mathbf{u})\}^\top$ and $\boldsymbol{\psi}^{(2)}(\mathbf{u}) = \{\psi_1^{(2)}(\mathbf{u}), \dots, \psi_{p_2}^{(2)}(\mathbf{u})\}^\top$, define the empirical inner product as

$$\langle \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(2)} \rangle_n = \frac{1}{n} \sum_{i=1}^n \left\{ \sum_{k=1}^{p_2} Z_{ik} \psi_k^{(1)}(\mathbf{U}_i) \right\} \left\{ \sum_{k=1}^{p_2} Z_{ik} \psi_k^{(2)}(\mathbf{U}_i) \right\},$$

and the theoretical inner product as

$$\langle \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(2)} \rangle = E \left\{ \sum_{k=1}^{p_2} Z_{ik} \psi_k^{(1)}(\mathbf{U}_i) \right\} \left\{ \sum_{k=1}^{p_2} Z_{ik} \psi_k^{(2)}(\mathbf{U}_i) \right\},$$

where $\mathbf{U}$ is the random location with density function $f_{\mathbf{U}}$, denote the corresponding empirical and theoretical norms as $\|\cdot\|_n$ and $\|\cdot\|$. Furthermore, let $\|\cdot\|_{\varepsilon_v}$ be the norm introduced by the inner product $\langle \cdot, \cdot \rangle_{\varepsilon_v}$, where, for $\boldsymbol{\psi}^{(1)}(\mathbf{u})$ and $\boldsymbol{\psi}^{(2)}(\mathbf{u})$ on Ω ,

$$\begin{aligned} \langle \boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(2)} \rangle_{\varepsilon_v} &= \sum_{k,k'=1}^{p_2} \int_{\Omega} \left\{ \sum_{i+j=v} \binom{v}{i} \left(D_{u_1}^i D_{u_2}^j \psi_k^{(1)} \right)^2 \right\}^{\frac{1}{2}} \\ &\quad \left\{ \sum_{i+j=v} \binom{v}{i} \left(D_{u_1}^i D_{u_2}^j \psi_{k'}^{(2)} \right)^2 \right\}^{\frac{1}{2}} du_1 du_2. \end{aligned}$$

Let $\eta_0(\mathbf{t}) = \sum_{l=1}^{p_1} x_l \beta_l + \sum_{k=1}^{p_2} z_k \alpha_k(\mathbf{u})$ and $\eta_i^0 = \eta(\mathbf{T}_i)$ and $\varepsilon_i = Y_i - g^{-1}(\eta_i^0)$ be the error term. For the quasi-likelihood function $\ell\{g^{-1}(x), y\}$, denote $q_1(x, y) = \frac{\partial}{\partial x} \ell\{g^{-1}(x), y\} = \{y - g^{-1}(x)\} \rho_1(x)$ and $q_2(x, y) = \frac{\partial^2}{\partial x^2} \ell\{g^{-1}(x), y\} = \{y - g^{-1}(x)\} \rho'_1(x) - \rho_2(x)$ where $\rho_j(x) = \{\dot{g}^{-1}(x)\}^j / [\sigma^2 V\{g^{-1}(x)\}]$, $j=1, 2$ and $\dot{g}^{-1}(x) = \{g^{-1}(x)\}'$. Denote $\rho_{2,k}(z) = E \rho_2\{\eta(\mathbf{T}) | Z_k = z\}$ for $k = 1, \dots, p_2$.

Assumptions

(A1) For any $k = 1, 2, \dots, p_2$, the bivariate function $\alpha_k \in \mathcal{W}^{d+1,\infty}(\Omega)$ for an integer $d \geq 1$.

(A2) The density function $f(\mathbf{u})$ of $\mathbf{U}$ is bounded away from zero and infinity on Ω , to be specific,

$$0 < c_f \leq \inf_{\mathbf{u} \in \Omega} f(\mathbf{u}) \leq \sup_{\mathbf{u} \in \Omega} f(\mathbf{u}) \leq C_f < \infty.$$

(A3) The eigenvalues of $E[\mathbf{X}_i^\top \mathbf{X}_i | \mathbf{U}_i = \mathbf{u}]$ are bounded away from zero. For $l = 1, 2, \dots, p_1$ and some $\tau > 0$, $E|X_{il}|^{4+\tau} < \infty$.

(A4) The function $q_2(x, y) < 0$ for $x \in \mathbb{R}$ and y in the range of the response variable. The functions $\eta^{(3)}(\cdot), V(\cdot), V''(\cdot), g^{-1}(\cdot), \{g^{-1}(x)\}'$ and $g^{(3)}(\cdot)$ are continuous, and there exist positive constants c_ρ and C_ρ such that $c_\rho \leq \rho_2(\cdot) \leq C_\rho$. For each $\mathbf{t}$, $\text{var}(Y | \mathbf{T} = \mathbf{t})$ and $g' \{\mu(\mathbf{t})\}$ are nonzero.

(A5) The error satisfy $E[\varepsilon_i | \mathbf{T}_i = \mathbf{t}] = 0$ and $E[|\varepsilon_i|^{2+\iota} | \mathbf{T}_i = \mathbf{t}] < \infty$ for some $\iota \in (1/2, \infty)$. $(\mathbf{X}_i, \mathbf{Z}_i, \mathbf{U}_i, \varepsilon_i)$ are independently and identically distributed.

(A6) For any $k = 1, 2, \dots, p_2$, there exists a positive constant C_k such that $|Z_k| \leq C_k$, the eigenvalues $\lambda_1(\mathbf{u}) \leq \lambda_2(\mathbf{u}) \leq \dots \leq \lambda_{p_2}(\mathbf{u})$ of $\boldsymbol{\Sigma}(\mathbf{u}) = E[(\mathbf{X}_i^\top \mathbf{Z}_i^\top)^\top (\mathbf{X}_i^\top \mathbf{Z}_i^\top) | \mathbf{U}_i = \mathbf{u}]$ are bounded away from zero and infinity uniformly for all $\mathbf{u} \in \Omega$; that is, there exist positive constants C_1 and C_2 such that $C_1 \leq \lambda_1(\mathbf{u}) \leq \lambda_2(\mathbf{u}) \leq \dots \leq \lambda_{p_2}(\mathbf{u}) \leq C_2$ for all $\mathbf{u} \in \Omega$.

(A7) The triangulation Δ is π -quasi-uniform, i.e., $|\Delta| \rho_\Delta^{-1} \leq \pi$ for some positive constant π where $\rho_\Delta = \min \{\rho_\tau, \tau \in \Delta\}$.

(A8) The triangulation size $|\Delta|$ satisfies that $|\Delta| \rightarrow 0$, $|\Delta|^{-1} \left(\frac{\log n}{n}\right)^{1/2} \rightarrow 0$ and the smoothness penalty parameters satisfy that $\lambda_{\max} n^{-1} |\Delta|^{-4} \rightarrow 0$ where $\lambda_{\max} = \max \{\lambda_1, \dots, \lambda_{p_2}\}$.

The following theorems establish the consistency of the proposed constant estimators and spline estimators for the GPLSVCM.

THEOREM 1. *Under Assumptions (A1)-(A8), the linear coefficient estimator $\widehat{\boldsymbol{\beta}}$ satisfies that*

$$\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\| = O_{\text{a.s.}} \left\{ \left(\frac{\log n}{n} \right)^{1/2} + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right\}$$

as $n \rightarrow \infty$ where $\lambda_{\max} = \max \{\lambda_1, \dots, \lambda_{p_2}\}$.

THEOREM 2. *Under Assumptions (A1)-(A8), the bivariate spline estimators $\widehat{\alpha}_k(\mathbf{u})$, $k = 1, \dots, p_2$ satisfy that*

$$\sum_{k=1}^{p_2} \|\widehat{\alpha}_k - \alpha_k\|_{L_2} = O_{\text{a.s.}} \left\{ |\Delta|^{-1} \left(\frac{\log n}{n} \right)^{1/2} + |\Delta|^{d+1} + \frac{\lambda_{\max}}{n|\Delta|^4} \right\}$$

as $n \rightarrow \infty$ where $\lambda_{\max} = \max \{\lambda_1, \dots, \lambda_{p_2}\}$.

2.3 Model Selection

2.3.1 The GPLSVCM-GIC Algorithm

When employing GPLSVCMs in reality, one must often determine whether there is a linear or varying effect for each explanatory variable. Therefore, we propose a backward elimination procedure via Generalized Information Criterion(GIC) proposed by [Zhang et al. \(2010\)](#). To deal with model selection in GLMs, [Zhang et al.](#) proposed that $\text{GIC}_{\kappa_n} = \frac{1}{n}D(\mathbf{y}; \hat{\boldsymbol{\mu}}) + \frac{1}{n}\kappa_n df$ where $D(\mathbf{y}; \hat{\boldsymbol{\mu}}) = 2 \{\ell(\mathbf{y}; \mathbf{y}) - \ell(\hat{\boldsymbol{\mu}}; \mathbf{y})\}$ is the scaled deviance, κ_n is a positive number that controls the model complexity, and df is the effective degree of freedom of the model. In [Algorithm 2](#), we demonstrate how to select the model structure for GPLSVCMs using GIC.

Algorithm 2 : GPLSVCM-GIC

Data: Observations $\{\mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i, Y_i\}_{i=1}^n$

Result: Index sets $\mathcal{I}_v$ and $\mathcal{I}_c$ that indicates the index of explanatory variables with varying and constant effect respectively

```
8 Initialization:  $\mathcal{I}_v = \{1, \dots, p\}; \mathcal{I}_c = \emptyset; \text{GIC}_{\text{old}} = \text{GIC}_{\text{new}} = \text{GIC}(\mathcal{I}_v, \mathcal{I}_c)$ 
9 while  $\text{GIC}_{\text{new}} \leq \text{GIC}_{\text{old}}$   $\&\& |\mathcal{I}_v| > 0$  do
10   (i) set  $\text{GIC}_{\text{old}} = \text{GIC}_{\text{new}}$ 
11   (ii) let  $k^* = \arg \min_{k \in \mathcal{I}_v} \text{GIC}(\mathcal{I}_v \setminus \{k\}, \mathcal{I}_c \cup \{k\})$ 
12   (iii) set  $\mathcal{I}_v = \mathcal{I}_v \setminus \{k^*\}; \mathcal{I}_c = \mathcal{I}_c \cup \{k^*\}$ 
13   (iv) calculate  $\text{GIC}_{\text{new}} = \text{GIC}(\mathcal{I}_v, \mathcal{I}_c)$ 
```

2.4 Simulation Study

In this section, we conduct simulation studies to evaluate the performance of the proposed model estimation and model selection method.

Following Ramsay (2002) and Wood et al. (2008a), we consider a modified horseshoe domain Ω in a $[-1, 3.5] \times [-1, 1]$ rectangle as in Figure 2.1. Firstly, we generate 226×101 grid points over the rectangle, and the grid points falling inside the horseshoe domain form a population with a size of $N = 16540$. Then, we simulate data over the population grid from a Poisson distribution and a negative binomial distribution, respectively, as in the following two cases:

Case I : $Y_i \sim \text{Poisson}(\mu_i)$, where

$$\log(\mu_i) = X_{i1}\beta_1 + X_{i2}\beta_2 + Z_{i1}\alpha_1(\mathbf{U}_i) + Z_{i2}\alpha_2(\mathbf{U}_i), i = 1, \dots, N;$$

Case II : $Y_i \sim \text{NB}\{\theta, \mu_i/(\theta + \mu_i)\}$ and $\theta = 6$, where

$$\log(\mu_i) = X_{i1}\beta_1 + X_{i2}\beta_2 + Z_{i1}\alpha_1(\mathbf{U}_i) + Z_{i2}\alpha_2(\mathbf{U}_i), i = 1, \dots, N;$$

Note that $E(Y_i) = \mu_i$, $\text{Var}(Y_i) = \mu_i + \mu_i^2/\theta$ in Case II. For the above models, Z_{i1} , Z_{i2} , X_{i1}

and X_{i2} are independently generated from $\text{Uniform}(-1,1)$. We set $\beta_1 = -1$ and $\beta_2 = 2$. $\alpha_1(\cdot)$ is the surface test function used in [Wood et al. \(2008a\)](#) and $\alpha_2(u_1, u_2) = 3.5 \sin(\frac{\pi}{4}u_1u_2)$. The contour plots of $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ are shown in Figure 2.1(a)-(b). For the following simulation studies, we conduct 200 Monte Carlo experiments for each case and randomly sample $n = 1000, 2000$ and 5000 points, respectively, from the population in each experiment.

To implement the BPST method in our experiments, we set $d = 2$ and $r = 1$ when generating the bivariate spline basis functions. We select the penalty parameter λ by minimizing the GCV over a 10-point grid where the values of $\ln(\lambda)$ are equally spaced between $\ln(0.0001)$ and $\ln(10)$. Two triangulations are considered: Δ_1 with 87 triangles and 73 vertices, and Δ_2 with 238 triangles and 161 vertices as shown in Figure 2.1 (c)-(d). In Case II, the estimator of θ is selected from $(2, 8)$, which ensures that the Pearson estimate of the scale parameter is as close to 1 as possible.

2.4.1 Simulation Study: Estimation

In this simulation study, we assess the performance of our estimation method in comparison to generalized geographically weighted regression (GGWR), assuming the true model structure is known. We only consider moderate sample sizes $n = 1000$ and 2000 for GGWR because it is extremely time-consuming for a large sample size. For its implementation, we use the functions `gwr` and `ggwr` in the R package `spgwr`.

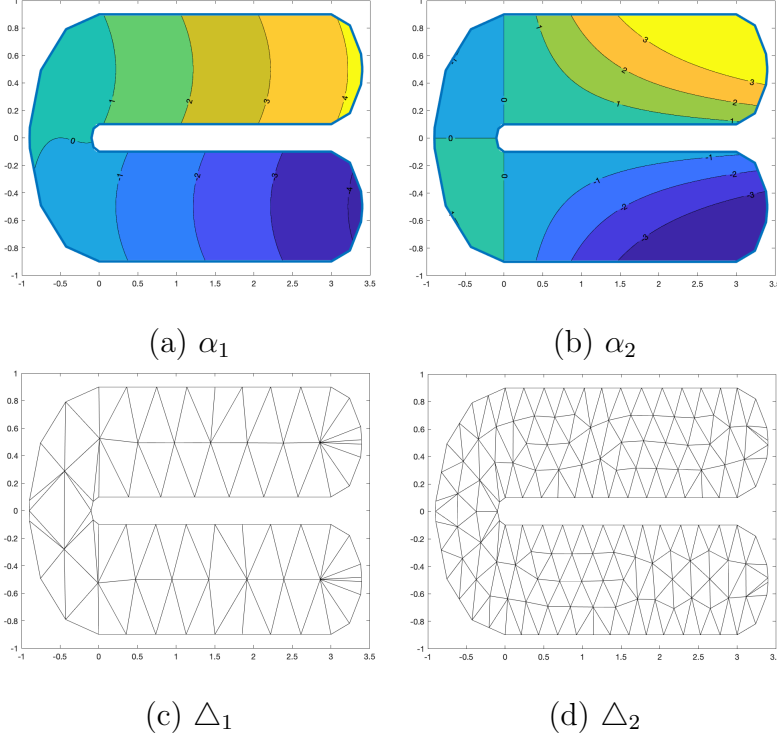


Figure 2.1: *Contour plots for the true coefficient functions and plots of triangulation 1 and 2 on the horseshoe domain.*

To evaluate the accuracy of the estimators, we calculate the mean squared error (MSE) of $\hat{\beta}_1$ and $\hat{\beta}_2$, and the mean integrated squared error (MISE) of $\hat{\alpha}_1(\cdot)$ and $\hat{\alpha}_2(\cdot)$ given as:

$$\text{MISE}(\hat{\alpha}_k) = \frac{1}{N} \sum_{i=1, \dots, N} \{\hat{\alpha}_k(\mathbf{u}_i) - \alpha_k(\mathbf{u}_i)\}^2, k = 1, 2$$

where $\mathbf{u}_i$, $i = 1, \dots, N$ are the grid points of the population over the domain. Table 2.1 summarizes the average MISE of $\hat{\alpha}_1(\cdot)$ and $\hat{\alpha}_2(\cdot)$, and average MSE of $\hat{\beta}_1$ and $\hat{\beta}_2$ based on 200 replications. From this table, we observe that our method significantly outperforms GGWR in terms of estimating both the linear coefficients and the varying coefficient functions. The estimation accuracy improves as the sample size increases. Moreover, the triangulation Δ_1 seems to be fine enough to reach good estimation accuracy as refining the triangulation does not reduce the MISEs or MSEs considerably. In addition, Figure 2.2 and 2.3 present the contour plots of the estimated coefficient functions using different methods for a typical

replication of $n=1000$ and 2000 . From those figures, one can visually tell that the surfaces of $\hat{\alpha}_1(\cdot)$ and $\hat{\alpha}_2(\cdot)$ from the BPST method are pretty similar to the true surfaces in Figure 2.1. By contrast, the estimated surfaces from the GGWR method are noticeably different from the true surfaces.

Furthermore, we report the computing time of one iteration for different methods. As is shown in Table 2.1, for $n = 1000$ and 2000 , our method only takes a few seconds for one iteration, while GGWR takes a few minutes or even hours. GGWR suffers from a heavy computational burden by requiring a weighted regression at every location. Compared to it, our method only requires solving a global maximization problem, which saves an enormous amount of computation time. This merit of our method also accounts for the computational efficiency of our model selection procedure in the following simulation study.

Table 2.1: *Model Estimation for the simulation study.*

Family	n	Method	MISE		MSE		Time (s)
			$\hat{\alpha}_1(\cdot)$	$\hat{\alpha}_2(\cdot)$	$\hat{\beta}_1$	$\hat{\beta}_2$	
Poisson	1000	BPST(Δ_1)	0.0302	0.0378	0.0006	0.0009	1.80
		BPST(Δ_2)	0.0281	0.0358	0.0006	0.0008	5.43
		GGWR	0.3937	0.2556	0.0261	0.0922	168.75
	2000	BPST(Δ_1)	0.0148	0.0184	0.0002	0.0002	2.94
		BPST(Δ_2)	0.0155	0.0175	0.0002	0.0003	6.95
		GGWR	0.1225	0.0902	0.0012	0.0138	413.00
	5000	BPST(Δ_1)	0.0065	0.0090	0.0001	0.0001	7.03
		BPST(Δ_2)	0.0075	0.0080	0.0001	0.0001	13.15
		GGWR	*	*	*	*	*
NB	1000	BPST(Δ_1)	0.0714	0.0808	0.0023	0.0021	10.50
		BPST(Δ_2)	0.0773	0.0844	0.0024	0.0022	27.33
		GGWR	0.1623	0.1050	0.0030	0.0039	1040.51
	2000	BPST(Δ_1)	0.0395	0.0459	0.0011	0.0010	13.72
		BPST(Δ_2)	0.0450	0.0492	0.0011	0.0011	31.59
		GGWR	0.0879	0.0653	0.0014	0.0015	5137.34
	5000	BPST(Δ_1)	0.0181	0.0207	0.0004	0.0003	37.63
		BPST(Δ_2)	0.0205	0.0213	0.0003	0.0003	76.20
		GGWR	*	*	*	*	*

Note: BPST(Δ_j) means the bivariate penalized spline estimation over triangulation Δ_j for $j = 1, 2$; MISE: mean integrated squared errors for varying coefficient estimators; MSE: mean squared errors for constant coefficient estimators.

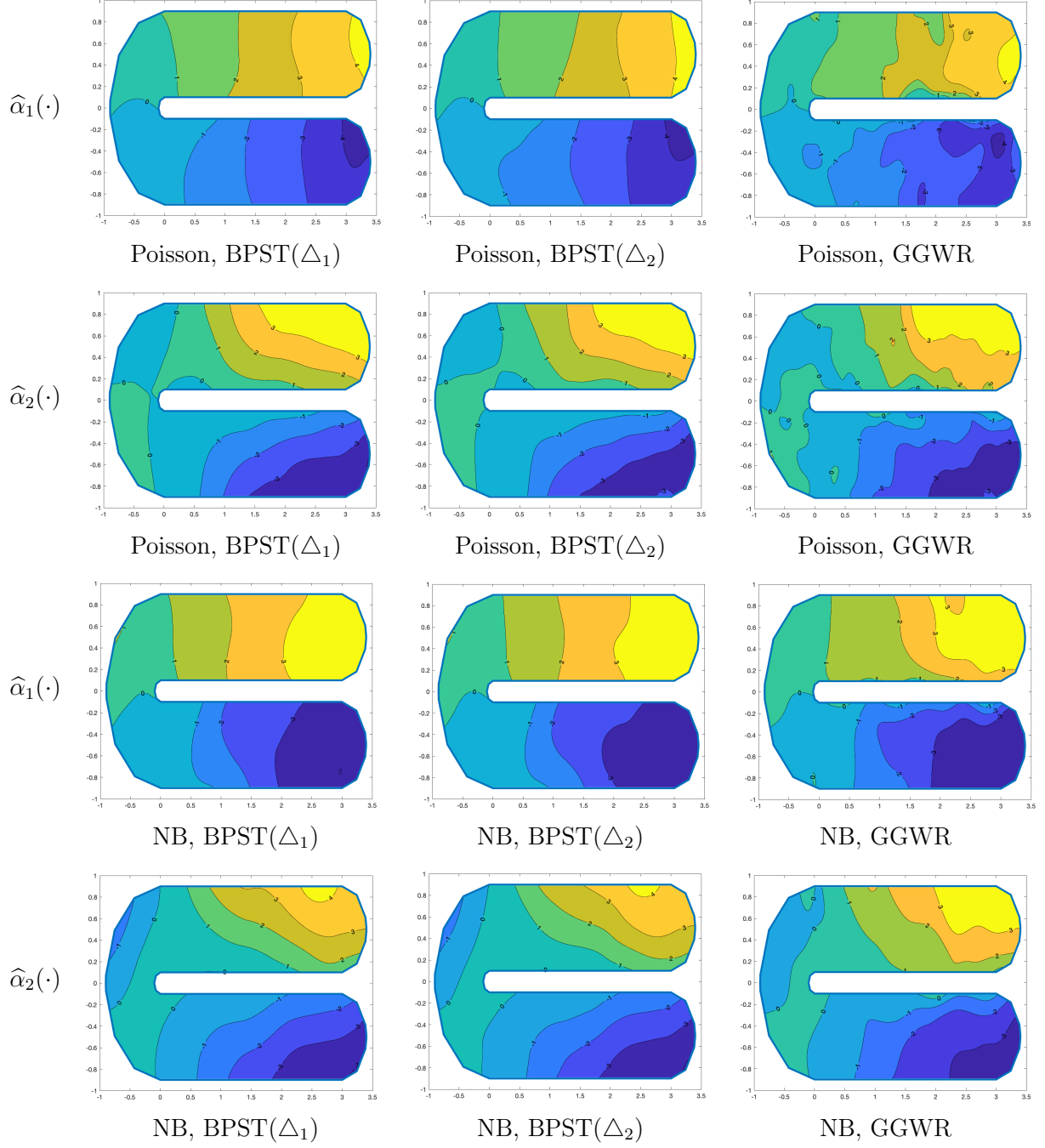


Figure 2.2: *Contour plots of estimated coefficient functions of BPST method and GGWR method under Poisson and Negative Binomial (NB) model when $n=1000$.*

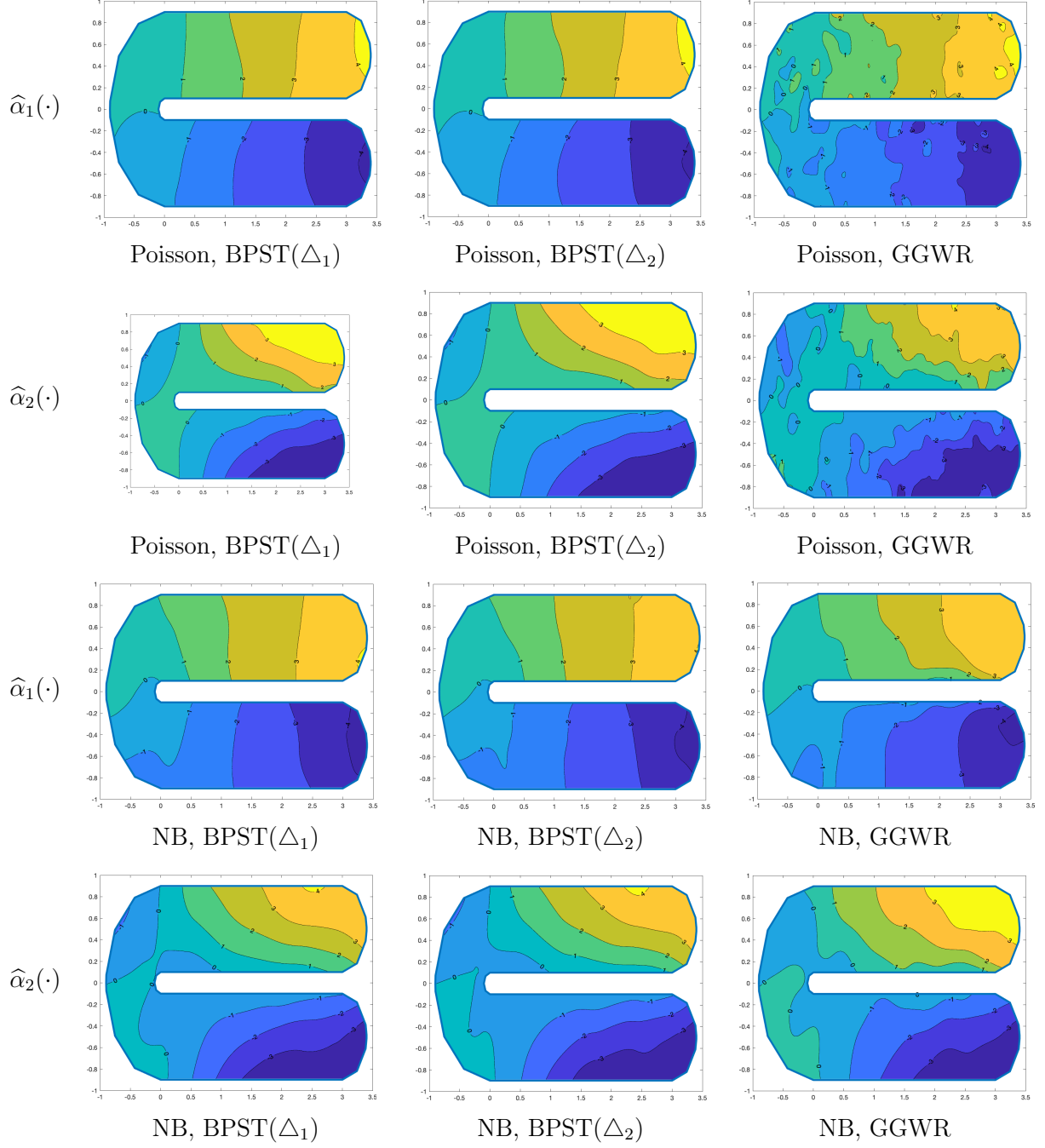


Figure 2.3: *Contour plots of estimated coefficient functions of BPST method and GGWR method under Poisson and Negative Binomial (NB) model when $n=2000$.*

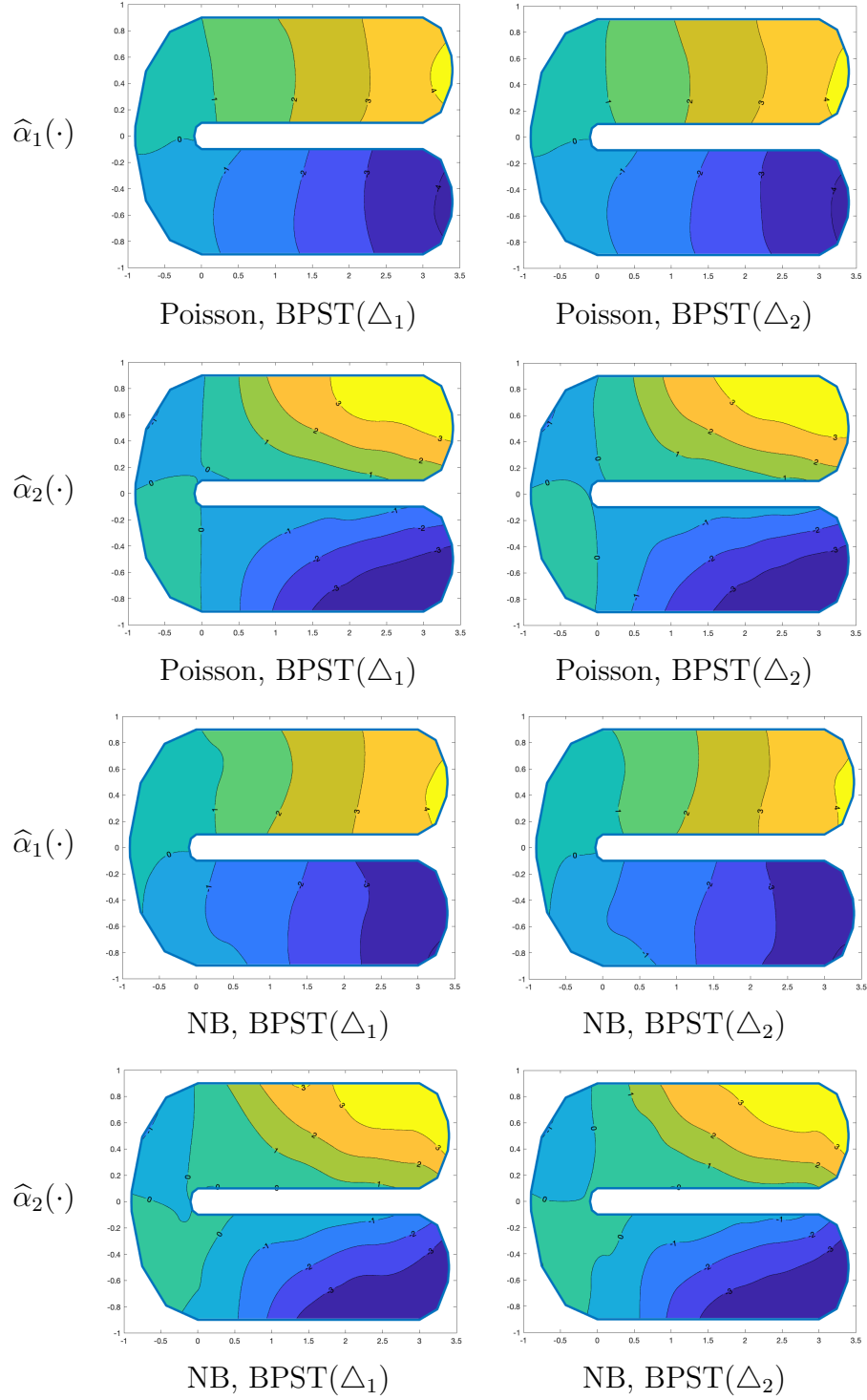


Figure 2.4: Contour plots of estimated coefficient functions of BPST method under Poisson and Negative Binomial (NB) model when $n=5000$.

2.4.2 Simulation Study: Model Selection

In the second simulation study, we examine the performance of our model selection algorithm regarding the accuracy of identifying the true model structure and estimation. We set $\kappa_n = \log(n)$ when calculating the GIC. For convenience, we refer to this setting as ‘BIC-type of selector’ hereafter. Note that in the true model, the set of varying coefficients is $\mathcal{I}_v = \{1, 2\}$ and the set of constant coefficients is $\mathcal{I}_c = \{3, 4\}$. In Table 2.2, we present the proportion of the estimated set of constant coefficients $\hat{\mathcal{I}}_c$ selected among 200 replications. The MISEs of $\hat{\alpha}_1(\cdot)$ and $\hat{\alpha}_2(\cdot)$, and MSEs of $\hat{\beta}_1$ and $\hat{\beta}_2$ are also summarized in the table. The results suggest that the BIC-type of selector can 100% correctly identify the true model structure regardless of the sample size and triangulation.

Table 2.2: *Proportion of $\hat{\mathcal{I}}_c$ (estimated set of constant coefficients) selected among 200 replications.*

Family	n	Tr	$\emptyset$	{3}	{4}	{3, 4}	MISE		MSE		Time (s)
							$\hat{\alpha}_1(\cdot)$	$\hat{\alpha}_2(\cdot)$	$\hat{\beta}_1$	$\hat{\beta}_2$	
Poisson	1000	Δ_1	0.000	0.000	0.000	1.000	0.0302	0.0378	0.0006	0.0009	18.95
		Δ_2	0.000	0.000	0.000	1.000	0.0281	0.0358	0.0006	0.0008	60.61
	2000	Δ_1	0.000	0.000	0.000	1.000	0.0148	0.0184	0.0002	0.0002	28.31
		Δ_2	0.000	0.000	0.000	1.000	0.0155	0.0175	0.0002	0.0003	76.36
	5000	Δ_1	0.000	0.000	0.000	1.000	0.0065	0.0090	0.0001	0.0001	54.24
		Δ_2	0.000	0.000	0.000	1.000	0.0075	0.0080	0.0001	0.0001	117.04
NB	1000	Δ_1	0.000	0.000	0.000	1.000	0.0714	0.0808	0.0023	0.0021	73.64
		Δ_2	0.000	0.000	0.000	1.000	0.0773	0.0844	0.0024	0.0022	239.51
	2000	Δ_1	0.000	0.000	0.000	1.000	0.0395	0.0459	0.0011	0.0010	100.20
		Δ_2	0.000	0.000	0.000	1.000	0.0450	0.0492	0.0011	0.0011	298.81
	5000	Δ_1	0.000	0.000	0.000	1.000	0.0181	0.0207	0.0004	0.0003	239.75
		Δ_2	0.000	0.000	0.000	1.000	0.0205	0.0213	0.0003	0.0003	667.49

Note: Δ_j : Triangulations; Proposed BIC is used for model selection; MISE: mean integrated squared errors for varying coefficient estimators; MSE: mean squared errors for constant coefficient estimators.

2.5 Application to Texas Crash Data

Traffic crashes are a severe problem in modern society. We can assist with road safety improvement by investigating the factors that determine the number of crashes. In this

section, we study the influence of some demographic characteristics on crash frequency using our methodology. We collect the 2017 crash frequency at 4771 census tracts from the Texas Department of Transportation, where a census tract is a small, relatively permanent statistical subdivisions of a county with about 4,000 inhabitants on average. Four explanatory variables are obtained from the Texas Department of Transportation and the U.S. Census Bureau; see Table 2.3 for the details. Note that all the explanatory variables are normalized in the analysis. In addition, we eliminate a few outliers and observations with no population from this study.

Table 2.3: *Description of the explanatory variables in the 2017 Texas crash dataset.*

Variable	Description
VMT*	Vehicle miles traveled
Population*	Total population
Old	Proportion of people age 65 and older
Hispanics	Proportion of Hispanics in population

Note: The variable with * are transformed from the original value by $X^* = \log(X + 0.01)$.

We consider a negative binomial regression model for this dataset, i.e., $Y_i \sim \text{NB}\left(\theta, \frac{\mu_i}{\theta + \mu_i}\right)$. For the spline basis functions, we set degree $d = 2, 3$ and 4 , respectively, each with smoothness $r = 1$. Figure 2.5 (a)-(c) shows three triangulations we adopt in the analysis: Δ_1 with 174 triangles and 119 vertices; Δ_2 with 202 triangles and 134 vertices; and Δ_3 with 272 triangles and 171 vertices. BIC-type of selector is used to select the model structure based on different triangulations and degrees of polynomials. Table 2.4 presents the selected linear covariates under different settings. Clearly, even if the triangulations and polynomial degrees change, our model selection procedure consistently chooses the same model:

$$\log(\mu_i) = \beta_0 + \beta_1 \text{VMT}_i + \beta_2 \text{Old}_i + \alpha_1(\mathbf{u}_i) \text{Population}_i + \alpha_2(\mathbf{u}_i) \text{Hispanics}_i,$$

$$\text{where } E(Y_i | x_i, z_i, u_i) = \mu_i \text{ and } \text{Var}(Y_i | x_i, z_i, u_i) = \mu_i + \mu_i^2 / \theta.$$

For comparison, we also fit a GLM on this dataset. To assess the estimation accuracy and predictive capability of different methods, we calculate the MSE and 10-fold cross-validation

mean squared prediction error (MSPE) for the selected GPLSVCM and GLM, respectively. The results are displayed in Table 2.4. We observe that our proposed method produces smaller MSEs and MSPEs than GLM, which emphasizes the significance of accounting for spatial nonstationarity in the study.

Moreover, in Figure 2.6, we show the estimated surfaces of the selected coefficient functions. According to Figure 2.6 (a), population has a positive impact on the crash frequency across Texas. This impact is increasing from south to north and west to east, with the highest level observed near the Texas-West Oklahoma border and Texas-Louisiana border. As suggested in Figure 2.6 (b), the proportion of Hispanics negatively affects the crash frequency in northern and eastern Texas. However, in western Texas, the proportion of Hispanics has a significantly positive impact on the crash frequency compared with other parts of the state.

Table 2.4: *Model Selection, Estimation and Prediction for Texas Car Crash Data.*

Model		MSE	MSPE	Linear Covariates
GLM		298.4822	298.9413	All
$d = 2$	BPST-BIC(Δ_1)	290.5389	296.5041	{ Intercept, VMT*, Old }
	BPST-BIC(Δ_2)	289.3179	296.7178	{ Intercept, VMT*, Old }
	BPST-BIC(Δ_3)	290.1385	296.4191	{ Intercept, VMT*, Old }
$d = 3$	BPST-BIC(Δ_1)	290.5910	296.4933	{ Intercept, VMT*, Old }
	BPST-BIC(Δ_2)	292.5110	296.8955	{ Intercept, VMT*, Old }
	BPST-BIC(Δ_3)	290.1709	296.4201	{ Intercept, VMT*, Old }
$d = 4$	BPST-BIC(Δ_1)	290.4451	296.5010	{ Intercept, VMT*, Old }
	BPST-BIC(Δ_2)	292.2836	296.6227	{ Intercept, VMT*, Old }
	BPST-BIC(Δ_3)	290.0437	296.3807	{ Intercept, VMT*, Old }

Note: BPST-BIC(Δ_j): bivariate penalized spline over triangulation Δ_j with BIC approach for model selection; GLM: generalized linear model is used as a benchmark; MSPE: mean square prediction error (MSPE) is calculated based on 10-fold cross-validation.

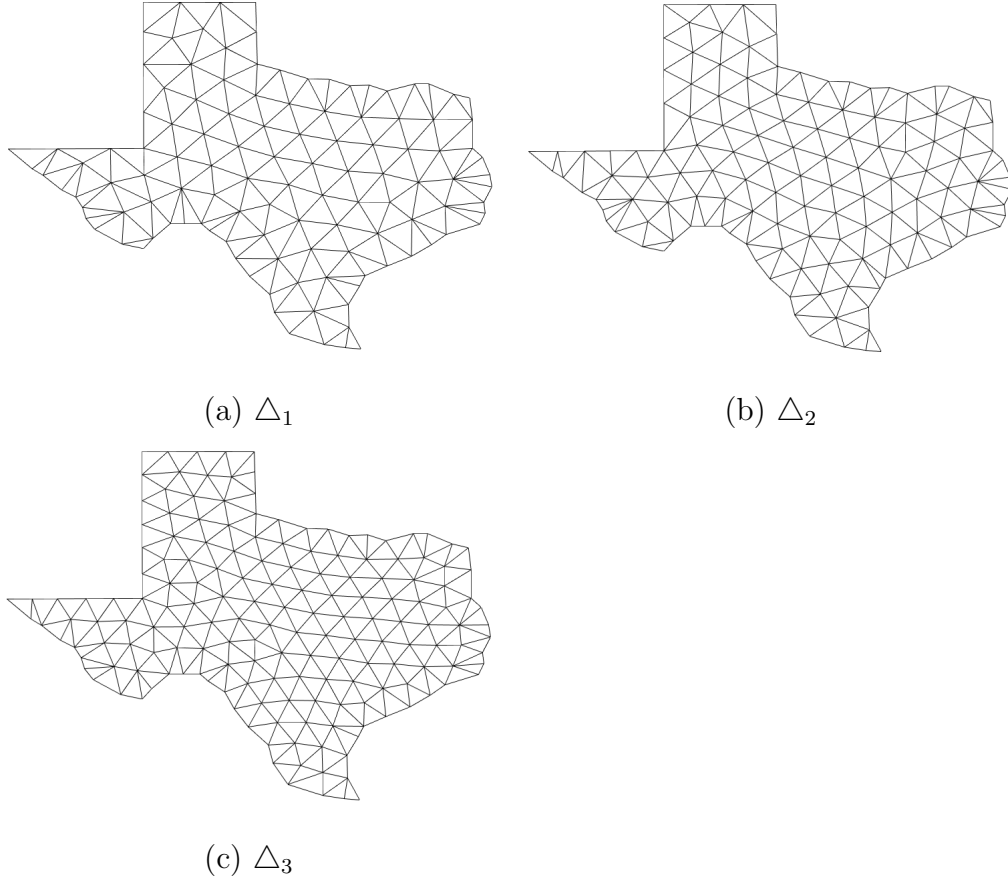


Figure 2.5: *Plots of triangulation 1, 2 and 3 on the Texas domain.*

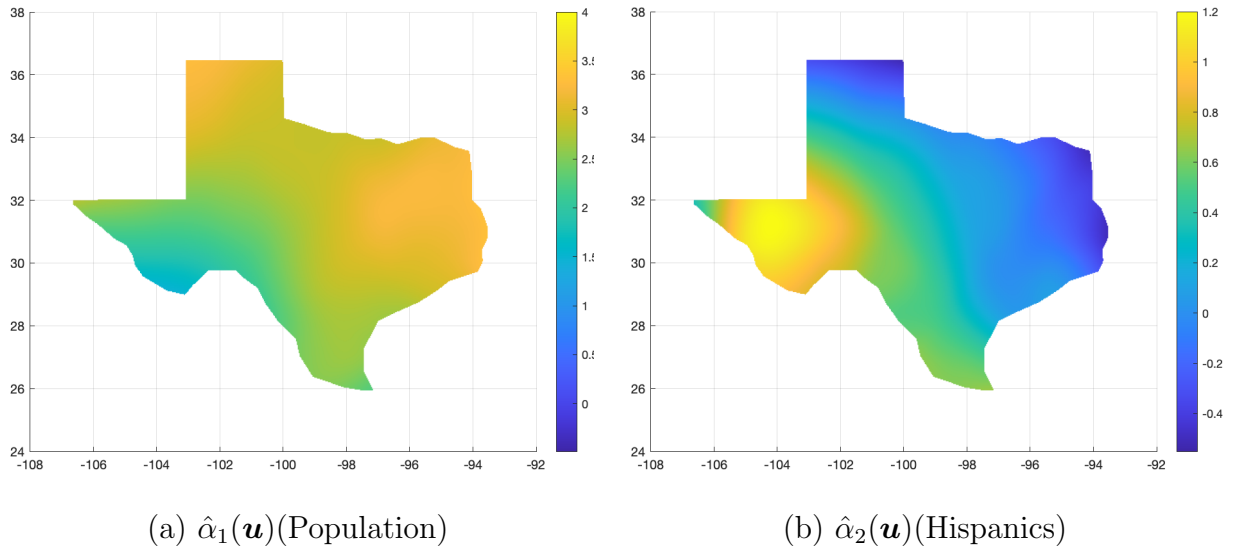


Figure 2.6: *Contour plots of the estimated coefficient surfaces on the Texas domain.*

2.6 Concluding Remarks

In this study, we consider a class of generalized semi-parametric regression models. This new model is very flexible in accommodating a wide range of existing models and dealing with various types of response variables, for example, discrete data and categorical data. For model estimation, we have proposed to use BPST method under a framework of penalized quasi-likelihood. The BPST method is advantageous compared with alternatives in some aspects. First, the BPST method is sufficiently fast in computation and convenient in implementation compared to GWR and the Bayesian approach. The proposed method is a global approach that allows various smoothness and degree parameters for different coefficient functions. Most importantly, the BPST method does not require the data to be evenly distributed on a rectangular domain as traditional smoothing tools do. To corroborate the method, we have established the consistency of the model estimators. Additionally, we have proposed a backward elimination approach via Generalized Information Criterion(GIC) for model structure selection that is proven to be able to accurately identify the constant and varying components of the coefficients in the simulation study. In general, our approach is balanced in terms of theory, computation, and interpretation, which significantly enhances its application to spatial data analysis.

Some extensions of this study warrant further research. In recent years, there has been a pervasive availability of data observed throughout time and space due to widespread network collecting and low-cost geographic information systems, which has prompted a wide range of new studies in various fields. It would be interesting to extend our approach to analyzing such spatiotemporal data. In addition, our backward elimination procedure for model selection is limited to the models with a small number of covariates. When increasing the dimensions of models to a certain degree, the procedure can be almost computationally inaccessible. However, high-dimensional data analysis has many applications in real life. It would be valuable to extend our work in a high-dimensional setting, and consider a unified approach via regularization methods for a simultaneous variable selection and model structure identification for GPLSVCs.

Chapter 3

An R Package: GPLSVCM

3.1 Introduction

In the era of big data, there is an increasing availability of large-scale data containing geographical information. For researchers' benefits, many R packages have been developed to promote the convenience of analyzing such spatially referenced data. Regarding fitting spatially varying coefficient model (SVCs) in R, the packages **spgwr** (Bivand and Yu, 2020), **gwrr** (Wheeler, 2013) and **GWmodel** (Gollini et al., 2015) have provided detailed implementations based on geographically weighted regression (GWR) as in Fotheringham et al. (2003). Moreover, a Bayesian-based package **spBayes** (Finley et al., 2007) recently offered new functionality for fitting SVCs using a MCMC algorithm, see Finley and Banerjee (2020). Although those packages have enabled easy access to fitting SVCs to a certain extent, their application can be easily hindered by a high dimensional dataset or a large sample size due to their huge computational expenses inherited from local regression or the Bayesian method. Another serious issue with those packages is that they fail to account for partially linear models as they treat all the covariates as having varying effects on the response.

In Chapter 2, we have proposed the generalized partially linear spatially varying coeffi-

cient model (GPLSVCM):

$$g\{E(Y_i | \mathbf{Z} = \mathbf{Z}_i, \mathbf{X} = \mathbf{X}_i, \mathbf{U} = \mathbf{U}_i)\} = \sum_{l=1}^{p_1} X_{il}\beta_l + \sum_{k=1}^{p_2} Z_{ik}\alpha_k(\mathbf{U}_i), i = 1, \dots, n, \quad (3.1)$$

where $g(\cdot)$ is the link function, $\mathbf{U}_i \in \Omega \subseteq \mathbb{R}^2$ with Ω being any arbitrarily shaped bounded domain, $\mathbf{X}_i = (X_{i1}, \dots, X_{ip_1})^\top$ is the vector of covariates having constant effects on the response, $\mathbf{Z}_i = (Z_{i1}, \dots, Z_{ip_2})^\top$ is the vector of covariates having varying effects on the response, β_l for $l = 1, \dots, p_1$ are unknown linear coefficients and $\alpha_k(\cdot)$ for $k = 1, \dots, p_2$ are unknown bivariate coefficient functions. For GPLSVCMs in (3.1), we proposed estimating the unknown parameters with bivariate penalized spline over triangulation (BPST). Based on BPST, we developed a backward elimination algorithm via GIC to identify the linear and nonlinear coefficients. The proposed model selection algorithm preserves oracle properties and computes relatively fast when there is a small number of covariates for a moderate sample size. However, it is almost computationally infeasible for many covariates or large sample sizes. On the other hand, in various fields such as genomics, bioinformatics, and economics, many covariates are usually available from a given sample, most of which may be noises. Removing irrelevant covariates through a variable selection procedure is imperative for reducing model complexity and thereby improving the performance of a model. Nevertheless, the proposed algorithm is incapable of selecting significant variables.

For variable selection of varying coefficient models (VCMs), shrinkage methods such as Lasso and SCAD are widely used in literature, see [Wang et al. \(2008\)](#), [Kai et al. \(2011\)](#), [Tang et al. \(2012\)](#) and [Wei et al. \(2011\)](#). In the context of partially linear varying coefficient models (PLVCMs), the model structure identification and variable selection need to be addressed simultaneously. Concerning this, [Wang and Lin \(2016\)](#) proposed a penalized method via adaptive lasso that consistently identifies the significant variables and parametric components along with estimation sparsity. In addition, [Wang and Kulasekera \(2012\)](#) proposed a method via penalized exponential squared loss that achieves simultaneous coefficient estimation and three selections (constant or varying effects and relevant variables) in one step. Nonetheless, their works are limited to PLVCMs in which the index variable of the coefficient function

is a scalar, such as time. [Li et al. \(2020\)](#) proposed a unified approach to simultaneously identify and estimate the nonzero spatially varying and constant coefficients in the presence of ultra-high-dimensional covariates. However, this approach only works for continuous responses. Unfortunately, there has been little discussion on simultaneous sparse estimation and model structure identification for GPLSVCs, and no packages with such functionality are available.

In many applications, a prediction interval, rather than a point prediction, is important for making inferences on a future observation. An example of this would be using the prediction interval of a blood sample as a reference range to determine whether or not a blood test is considered standard. In statistics, regression analysis is one of the most popular applications of prediction intervals. [Barber et al. \(2021\)](#) recently proposed a distribution-free method called jackknife+ to compute prediction intervals for a regression model and theoretically revealed that both jackknife and jackknife+ prediction intervals achieve target coverage probability under algorithmic stability assumptions. Moreover, [Barber et al. \(2021\)](#) extended the jackknife+ to K-fold cross-validation and similarly established rigorous coverage properties. Motivated by this work, we aim to provide a way for constructing prediction intervals for GPLSVCs.

To facilitate the application of GPLSVCs in spatial data analysis, we develop a new R package **GPLSVC** that embraces model structure identification, variable selection, model fitting, and predictive inference of GPLSVCs. To fit the model, we employ our proposed quasi-likelihood maximization procedure based on BPST since it is computationally efficient and provably reliable in tackling complex domains. For low dimensional models where variable selection is undesired, we offer the implementation of our proposed backward elimination algorithm via GIC. To fill the gap of simultaneous variable selection and model identification for GPLSVCs, we develop a doubly penalized algorithm via adaptive group lasso to automatically distinguish covariates with constant or varying effects while selecting contributive covariates. Additionally, to communicate the uncertainty of a prediction, we exploit the ideas in [Barber et al. \(2021\)](#) and develop three algorithms for constructing prediction intervals for GPLSVCs. The prediction intervals should be valuable for practitioners wishing to

assess the accuracy of their predictions. Taken together, our package **GPLSVCM** provides a powerful toolkit for regression analysis of various spatial data types.

Our package **GPLSVCM** contains five main functions. The `gplsvcm_aglasso` function performs model selection and fitting. The `compute_PIs` function constructs prediction intervals. Functions `gplsvcm_predict` and `cv_gplsvcm` compute point predictions and k-fold cross-validation mean squared prediction error respectively. And `gplsvcm_plot` function plots the surface of the estimated coefficient functions. To provide a general picture of our package, we demonstrate the overall process of **GPLSVCM** with a flow chart in Figure 3.1.

The rest of the article is organized as follows. Section 3.2 presents how to use our package to fit a GPLSVCM when the model structure is specified. Section 3.3 describes the algorithms for constructing prediction intervals and their simulation studies. In section 3.4, we illustrate the doubly penalized model selection approach and evaluate its performance through simulation studies. In section 3.5, we demonstrate the application of our package using a New York City crime dataset. The sample R code are presented in Appendix B.

3.2 Model Fitting

The model fitting procedure in **GPLSVCM** is performed via BPST method, which has been thoroughly discussed in Chapter 2. In this section, we detail how to fit a GPLSVCM using the `gplsvcm_fit` function in our package.

3.2.1 Bivariate Penalized Spline over Triangulation

In BPST method, the coefficient function is approximated with a basis expansion of bivariate Bernstein polynomials over Triangulation with a smoothness matrix constraining the spline coefficient vector. The construction of Bernstein polynomials basis and smoothness matrix are already embedded in the function `gplsvcm_fit` using the function `basis` in the R package **BPST** (Wang et al., 2020). Regarding construction of the triangulation for a given domain, we recommend using the function `TriMesh` from the R package **Triangulation** (Lai and Wang,

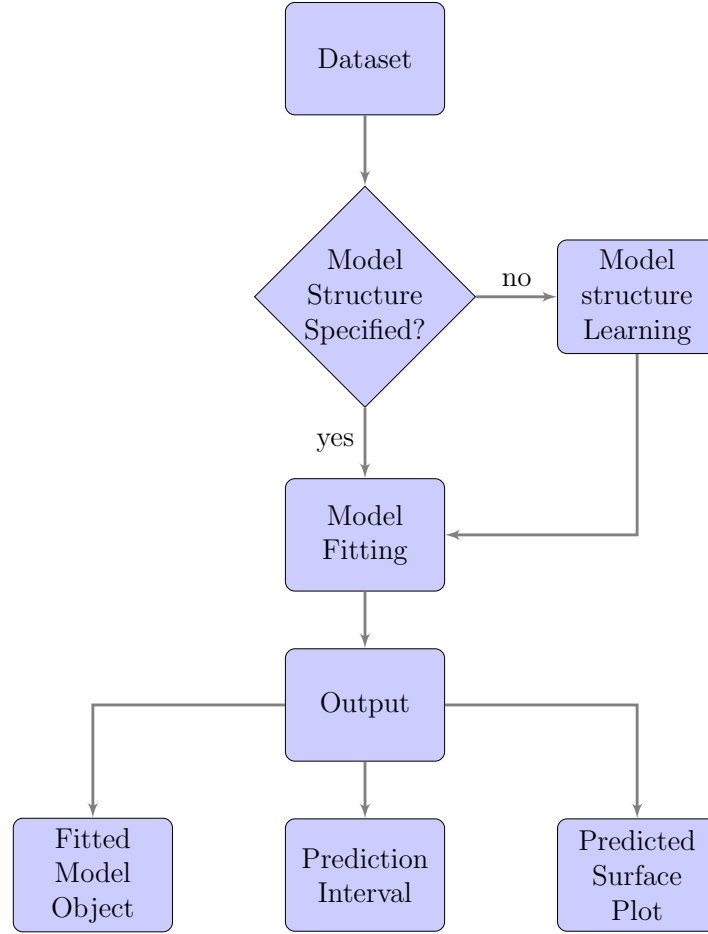


Figure 3.1: *Data analysis process of GPLSVCM*

2020). The `TriMesh` function is based on the Delaunay algorithm that follows the “max-min” criterion; therefore, it could guarantee the generation of a uniform triangulation.

To demonstrate how to fit a GPLSVCM, we first generate a dataset from a Poisson distribution over a horseshoe domain as in Wood et al. (2008b). In the following code, we use the function `Datagenerator` in **GPLSVCM** to generate this dataset:

```

R> family=poisson()
R> ngrid = 0.02
R> pop = Datagenerator(family, ngrid)
R> N=nrow(pop)
R> ind_s=sample(N,n,replace=FALSE)
R> data=as.matrix(pop[ind_s,])

```

```

R> Y=as.matrix(data[,1])
R> X=cbind(rep(1,length(Y)),data[,c(6:9)])
R> U=data[,c(10:11)]
R> ind_l=c(1,4,5)
R> ind_nl=c(2,3)

```

Note that the object `ind_l` and `ind_nl` save the indexes of covariates with constant and varying effects respectively. Next, we call the function `TriMesh` in **Triangulation** to construct the triangulation for the horseshoe domain. Inside the function, the first argument is the boundary of the domain, and the second argument `n` is for controlling the fineness of the triangulation; see the code as follows for the details:

```

R> library(Triangulation)
R> data('boundary.horseshoe')
R> Tr=TriMesh(boundary.horseshoe, n = 8)$Tr
R> V=TriMesh(boundary.horseshoe, n = 8)$V

```

Given a sequence of penalty parameters, we then call the `gplsvcm_fit` function to fit the model for a specified degree of polynomial spline d and smoothness r . In the code below, we illustrate an example for $d = 2$ and $r = 1$:

```

R> lambda=exp(seq(log(0.0001),log(10),length.out=10))
R> mfit = gplsvcm_fit(Y=Y,X=X,ind_l=ind_l,ind_nl=ind_nl,U=U,V=V,Tr=Tr,d=2,r=1,
+   lambda=lambda,family=family)

```

The object `mfit` contains the estimated varying coefficients and constant coefficients as well as all the other information related to BPST method, for example, the spline basis matrix `B`, the spline coefficient estimator `Qtheta`, `d` and `r`.

3.3 Predictive Inference

Predictive inference can be a key component in statistical analysis because it usually provides more information about a future observation than a point prediction. In this section, we introduce the technical details of computing prediction intervals for GPLSVCMs underlying

the function `compute_PIs` in **GPLSVCM**.

3.3.1 Prediction Intervals

Our function `compute_PIs` builds on the work in [Barber et al. \(2021\)](#). We offer three methods to compute prediction intervals in this function: `jackknife`, `jackknife+`, and `K-fold cross-validation (CV+)`. Before moving forward, we first introduce some notations. Suppose we fit a GPLSVCM on an i.i.d. training dataset $\{(Y_i, \mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i)\}_{i=1}^n$, then we denote $\hat{\mu}$ as the fitted regression function and $\hat{\mu}_{-i}$ as the fitted regression function with the i th data point removed from the training dataset. Given a new data point $(Y_{n+1}, \mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$ from the same population as the training dataset, we denote $\hat{\mu}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$ as the prediction of Y_{n+1} . Similarly, for a subset of the training dataset S , we denote $\hat{\mu}_{-S}$ as the fitted regression function with the subset removed. Moreover, for any sequence of values $\{(e_i)\}_{i=1}^n$, define

$$\hat{q}_{n,\alpha}^+ \{e_i\} = \text{the } \lceil (1 - \alpha)(n + 1) \rceil - \text{th smallest value of } e_1, \dots, e_n.$$

as the $1 - \alpha$ quantile of the empirical distribution of the sequence. Likewise, define the α quantile of the sequence as

$$\hat{q}_{n,\alpha}^- \{e_i\} = \text{the } \lfloor \alpha(n + 1) \rfloor - \text{th smallest value of } e_1, \dots, e_n.$$

We present the procedures for constructing `jackknife`, `jackknife+` and `CV+` prediction intervals in Algorithm 3, 4 and 5. In **GPLSVCM**, we call the function `compute_PIs` to compute a prediction interval:

```
R> set.seed(123)
R> compute_PIs( Y_train, X_train, ind_l, ind_nl, U_train, X_pred, U_pred, V, Tr,
d = 2, r = 1, lambda, family, off = 0, r_theta = c(2, 8), eps = 0.01, method =
"CV+", cp = 0.95, nfold = 10)
```

Algorithm 3 : Jackknife

Step 1. Fit a GPLSVCM on the training dataset $\{(Y_i, \mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i)\}_{i=1}^n$ to obtain the fitted regression function $\hat{\mu}$.

Step 2. Fit a GPLSVCM on the training data with the i th data point removed to obtain $\hat{\mu}_{-i}$ for each $i = 1, \dots, n$.

Step 3. Compute the leave-one-out residual $R_i^{\text{LOO}} = |Y_i - \hat{\mu}_{-i}(\mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i)|$ for $i = 1, \dots, n$.

Step 4. For any new data point $(Y_{n+1}, \mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$, compute $\hat{\mu}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$.

Step 5. Compute the jackknife prediction interval for the new data point

$$[\hat{q}_{n,\alpha}^- \{ \hat{\mu}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1}) - R_i^{\text{LOO}} \}, \hat{q}_{n,\alpha}^+ \{ \hat{\mu}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1}) + R_i^{\text{LOO}} \}].$$

Algorithm 4 : Jackknife+

Step 1. Fit a GPLSVCM on the training dataset $\{(Y_i, \mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i)\}_{i=1}^n$ with the i th data point removed to obtain $\hat{\mu}_{-i}$ for each $i = 1, \dots, n$.

Step 2. Compute the leave-one-out residual $R_i^{\text{LOO}} = |Y_i - \hat{\mu}_{-i}(\mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i)|$ for $i = 1, \dots, n$.

Step 3. For any new data point $(Y_{n+1}, \mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$, compute $\hat{\mu}_{-i}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$ for $i = 1, \dots, n$.

Step 4. Compute the jackknife+ prediction interval for the new data point

$$[\hat{q}_{n,\alpha}^- \{ \hat{\mu}_{-i}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1}) - R_i^{\text{LOO}} \}, \hat{q}_{n,\alpha}^+ \{ \hat{\mu}_{-i}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1}) + R_i^{\text{LOO}} \}].$$

Algorithm 5 : CV+

Step 1. Evenly split the training dataset $\{(Y_i, \mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i)\}_{i=1}^n$ into K disjoint subsets $S_1, \dots, S_K$.

Step 2. Fit a GPLSVCM on the training data with the subset $S_{k(i)}$ removed to obtain $\hat{\mu}_{-S_{k(i)}}$ for each $i = 1, \dots, n$, where $k(i) \in \{1, \dots, K\}$ identifies the subset that contains i .

Step 3. Compute the K -fold residual $R_i^{\text{CV}} = |Y_i - \hat{\mu}_{-S_{k(i)}}(\mathbf{U}_i, \mathbf{X}_i, \mathbf{Z}_i)|$ for $i = 1, \dots, n$.

Step 4. For any new data point $(Y_{n+1}, \mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$, compute

$\hat{\mu}_{-S_{k(i)}}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1})$ for $i = 1, \dots, n$.

Step 5. Compute the CV+ prediction interval for the new data point

$$[\hat{q}_{n,\alpha}^- \{ \hat{\mu}_{-S_{k(i)}}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1}) - R_i^{\text{CV}} \}, \hat{q}_{n,\alpha}^+ \{ \hat{\mu}_{-S_{k(i)}}(\mathbf{U}_{n+1}, \mathbf{X}_{n+1}, \mathbf{Z}_{n+1}) + R_i^{\text{CV}} \}].$$

Empirically, the jackknife+ prediction interval can approximately achieve the target coverage probability, while the CV+ prediction interval usually leads to a slightly larger coverage probability for a regression model. As a comparison, the jackknife prediction interval might have an under-coverage problem for an unstable regression algorithm in practice. Despite that, it is indicated in [Barber et al. \(2021\)](#) that without making any assumptions on the distribution of $(\mathbf{Y}, \mathbf{U}, \mathbf{X}, \mathbf{Z})$ or our penalized iteratively reweighted least square algorithm (IRLS), the jackknife and jackknife+ prediction intervals for GPLSVCMs are theoretically guaranteed to obtain the target coverage probability. Regarding the computational cost, the CV+ prediction interval is the most efficient. However, it comes at the price of having a prediction interval with a larger width than the other two. In practice, we recommend using the jackknife or jackknife+ method for small sample sizes and the CV+ method for large sample sizes in `compute_PIs`.

3.3.2 Simulation Study 1 (Prediction Interval)

In this simulation study, we investigate the predictive performance of the jackknife, jackknife+, and CV+ in **GPLSVCM**. We still consider simulating population grid of size $N = 16540$ from the modified horseshoe domain in a $[-1, 3.5] \times [-1, 1]$ rectangle as in [Ramsay \(2002\)](#), [Sangalli et al. \(2013\)](#) and [Wood et al. \(2008a\)](#). To generate the response variable Y_i , we consider the following three GPLSVCM models:

Case I : $Y_i \sim N(\mu_i, \sigma^2)$ and $\sigma^2 = 0.5$, where

$$\mu_i = X_{i1}\beta_1 + X_{i2}\beta_2 + Z_{i1}\alpha_1(\mathbf{U}_i) + Z_{i2}\alpha_2(\mathbf{U}_i), i = 1, \dots, N;$$

Case II : $Y_i \sim \text{Poisson}(\mu_i)$, where

$$\log(\mu_i) = X_{i1}\beta_1 + X_{i2}\beta_2 + Z_{i1}\alpha_1(\mathbf{U}_i) + Z_{i2}\alpha_2(\mathbf{U}_i), i = 1, \dots, N;$$

Case III : $Y_i \sim \text{NB}(\theta, \mu_i/(\theta + \mu_i))$ and $\theta = 6$, where

$$\log(\mu_i) = X_{i1}\beta_1 + X_{i2}\beta_2 + Z_{i1}\alpha_1(\mathbf{U}_i) + Z_{i2}\alpha_2(\mathbf{U}_i), i = 1, \dots, N;$$

We generate X_{i1} , X_{i2} , Z_{i1} and Z_{i2} independently from $\text{Uniform}(-1, 1)$. The true constant

coefficients β_1, β_2 are equal to -1 and 2 respectively. The true varying coefficient function $\alpha_1(\cdot)$ is the surface test function given in Wood et al. (2008a) while the other one is $\alpha_2(u_1, u_2) = 3.5 \sin(\frac{\pi}{4}u_1u_2)$. The contour plots of these two coefficient functions are illustrated in Figure 2.1(a)-(b) of chapter 2. We randomly select $n = 1000, 2000, 5000$ samples from each population as the training data set and independently sample 200 test data points from each corresponding population. In the part of model fitting, we consider using polynomial splines of degree $d = 2$ with smoothness $r = 1$ and a triangulation with 87 triangles, 73 vertices as in Figure 3.2. To examine the performance of each prediction interval in our package, we calculate the proportion of test points for which the computed prediction interval contains the observed response (i.e., the empirical probability of coverage). Figure 3.2 (a)-(c) displays the averaged empirical coverage probability of the 95% prediction intervals for each method over 100 Monte Carlo experiments. The results clearly show satisfactory coverage rates of the proposed prediction intervals, regardless of the distribution and sample sizes for all three methods.

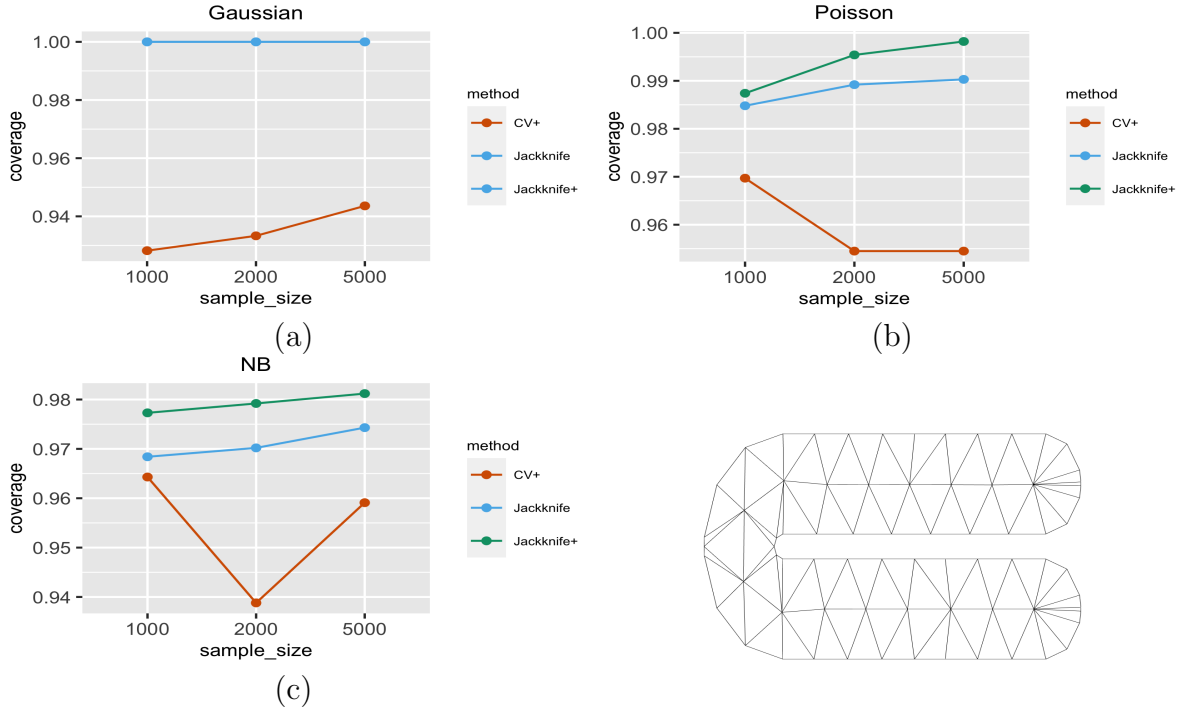


Figure 3.2: Simulation results showing the empirical coverage rates of the predictive interval for all methods based on different distributions and sample sizes.

3.4 Model Selection

This section explains the methodology of identifying the nonzero constant and varying coefficients behind the function `gplsvcm_aglasso` in **GPLSVCM**.

3.4.1 Simultaneous Variable and Model Structure Selection

In what follows, we assume the functional form of the coefficient (constant or varying) for each covariate is unknown in model 3.1. That is, we consider the following model

$$g\{E(Y_i | \mathbf{X}_i, \mathbf{U}_i)\} = \sum_{k=1}^p X_{ik} f_k(\mathbf{U}_i), \quad \mathbf{U}_i \in \Omega \text{ and } i = 1, \dots, n \quad (3.2)$$

where $\mathbf{X}_i = (X_{i1}, \dots, X_{ip})^\top$ is the vector of covariates and $f_k(\mathbf{U}_i) = \beta_k^* + \alpha_k^*(\mathbf{U}_i)$ with β_k^* being some unknown constant and α_k^* being some unknown bivariate smooth function for $k = 1, \dots, p$. On account of model identifiability, we assume that $\int_{\Omega} \alpha_k^*(\mathbf{U}) d\mathbf{U} = 0$ since it guarantees the nonexistence of constant component in the bivariate function $\alpha_k^*(\mathbf{U})$. Therefore, model 3.2 can be represented as follows:

$$g\{E(Y_i | \mathbf{X}_i, \mathbf{U}_i)\} = \sum_{k=1}^p X_{ik} \beta_k^* + \sum_{k=1}^p X_{ik} \alpha_k^*(\mathbf{U}_i), \quad \mathbf{U}_i \in \Omega \text{ and } i = 1, \dots, n \quad (3.3)$$

For above model, we say X_k has spatially varying effect on the response variable if $\alpha_k^*(\mathbf{U}) \neq 0$ for some $\mathbf{U} \in \Omega$, and X_k has a constant effect on the response variable if $\beta_k^* \neq 0$ and $\alpha_k^*(\mathbf{U}) = 0$ for all $\mathbf{U} \in \Omega$. As we intend to simultaneously identify the insignificant covariates, we say X_k has no relationship (zero effect) with the response variable if both $\beta_k^* = 0$ and $\alpha_k^*(\mathbf{U}) = 0$ for all $\mathbf{U} \in \Omega$. Specifically, we define the following index sets for $\mathbf{X}$:

index set of nonzero constant effect for $\mathbf{X}$: $\mathcal{I}_{x,C} = \{k = 1, \dots, p : \beta_k^* \neq 0, \alpha_k^*(\cdot) \equiv 0\}$,

index set of varying effect for $\mathbf{X}$: $\mathcal{I}_{x,V} = \{k = 1, \dots, p : \alpha_k^*(\cdot) \neq 0\}$,

index set of zero effect for $\mathbf{X}$: $\mathcal{I}_{x,0} = \{k = 1, \dots, p : \beta_k^* = 0 \text{ and } \alpha_k^*(\cdot) \equiv 0\}$.

In this way, the sparse model selection problem in model 3.3 is equivalent to the problem of identifying $\mathcal{I}_{x,C}$, $\mathcal{I}_{x,V}$ and $\mathcal{I}_{x,0}$. To solve it, we propose a doubly penalized approach by

minimizing the objective function as below:

$$-\sum_{i=1}^n \ell \left[g^{-1} \left\{ \sum_{k=1}^p X_{ik} \beta_k^* + \sum_{k=1}^p X_{ik} \alpha_k^*(\mathbf{U}_i) \right\}, Y_i \right]^2 + \sum_{k=1}^p p_{\rho,1}(|\beta_k^*|) + \sum_{k=1}^p p_{\rho,2}(\|\alpha_k^*\|) \quad (3.4)$$

where $\|\alpha_k^*\| = [E \{\alpha_k^*(\mathbf{U})\}]^{1/2}$, $p_{\rho,1}(\cdot)$ and $p_{\rho,2}(\cdot)$ are the penalty functions imposed on the constant and varying components of the coefficients, which are indexed by a regularization parameter ρ that controls model complexity.

In order to satisfy the assumption $\int_{\Omega} \alpha_k^*(\mathbf{U}) d\mathbf{U} = 0$, we use the normalized spline basis of $\mathbf{B}(\mathbf{U}) = \{B_1(\mathbf{U}), \dots, B_{M_n}(\mathbf{U})\}$ to approximate α_k^* . More concretely, we define the centralized bivariate spline basis function as $B_m^0(\cdot) = B_m(\cdot) - B_1(\cdot) \int_{\Omega} B_m(\mathbf{U}) d\mathbf{U} / \int_{\Omega} B_1(\mathbf{U}) d\mathbf{U}$ for $m = 1, \dots, M_n$. The collection of all the normalized bivariate basis can be then written as $\mathbf{B}^c(\mathbf{U}) = \{B_m^c(\mathbf{U}), m = 1, \dots, M_n\}$ where $B_m^c(\cdot) = B_m^0(\cdot) / \|B_m^0(\cdot)\|_{L^2(\Omega)}$ for $m = 1, \dots, M_n$ and $\|g\|_{L^2(\Omega)} = [\int_{\Omega} g^2(\mathbf{U}) d\mathbf{U}]^{1/2}$ for any bivariate function g . On this basis, the problem in (3.4) can be converted to minimizing the function:

$$-\sum_{i=1}^n \ell \left[g^{-1} \left\{ \sum_{k=1}^p X_{ik} \beta_k^* + \sum_{k=1}^p X_{ik} \mathbf{B}^{c\top}(\mathbf{U}_i) \boldsymbol{\theta}_k \right\}, Y_i \right]^2 + \sum_{k=1}^p p_{\rho,1}(|\beta_k^*|) + \sum_{k=1}^p p_{\rho,2}(\|\boldsymbol{\theta}_k\|) \quad (3.5)$$

where $\|\boldsymbol{\theta}_k\|$ is the Euclidean norm of $\boldsymbol{\theta}_k$.

To tackle the above penalized problem, we propose using the adaptive group lasso. To be exact, we first obtain the group lasso estimator $(\tilde{\boldsymbol{\beta}}^*, \tilde{\boldsymbol{\theta}})$ by solving the following regularized problem:

$$\begin{aligned} (\tilde{\boldsymbol{\beta}}^*, \tilde{\boldsymbol{\theta}}) = \arg \min_{\boldsymbol{\beta}^*, \boldsymbol{\theta}} \{ & -\sum_{i=1}^n \ell [g^{-1} \{ \sum_{k=1}^p X_{ik} \beta_k^* + \sum_{k=1}^p X_{ik} \mathbf{B}^{c\top}(\mathbf{U}_i) \boldsymbol{\theta}_k \}, Y_i]^2 \\ & + \sum_{k=1}^p \tilde{\rho} |\beta_k^*| + \sum_{k=1}^p \tilde{\rho} \|\boldsymbol{\theta}_k\| \} \end{aligned} \quad (3.6)$$

where $\boldsymbol{\beta}^* = (\beta_1^*, \dots, \beta_p^*)^\top$, $\boldsymbol{\theta} = (\boldsymbol{\theta}_1^\top, \dots, \boldsymbol{\theta}_p^\top)^\top$, and $\tilde{\rho}$ is a penalty parameter. Denote $w_k^c = |\tilde{\beta}_k^*|^{-1}$ and $w_k^v = \|\tilde{\boldsymbol{\theta}}_k\|^{-1}$, we set $p_{\rho,1}(|\beta_k^*|) = \rho w_k^c |\beta_k|$ and $p_{\rho,2}(\|\boldsymbol{\theta}_k\|) = \rho w_k^v \|\boldsymbol{\theta}_k\|$. Then, we

formulate the following adaptive group lasso objective function:

$$\begin{aligned} \mathcal{L}(\boldsymbol{\beta}^*, \boldsymbol{\theta}; \rho) = & - \sum_{i=1}^n \ell \left[g^{-1} \left\{ \sum_{k=1}^p X_{ik} \beta_k^* + \sum_{k=1}^p X_{ik} \mathbf{B}^{c\top} (\mathbf{U}_i) \boldsymbol{\theta}_k \right\}, Y_i \right]^2 \\ & + \sum_{k=1}^p \rho w_k^c |\beta_k| + \sum_{k=1}^p \rho w_k^v \|\boldsymbol{\theta}_k\| \end{aligned} \quad (3.7)$$

By minimizing (3.7), we obtain the adaptive group LASSO estimators as below:

$$(\hat{\boldsymbol{\beta}}^*, \hat{\boldsymbol{\theta}}) = \arg \min_{\boldsymbol{\beta}^*, \boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\beta}^*, \boldsymbol{\theta}; \rho)$$

As a result, the selected model can be denoted as:

$$\hat{\mathcal{I}}_{x,C} = \left\{ k : \left| \hat{\beta}_k^* \right| > 0, \left\| \hat{\boldsymbol{\theta}}_k \right\|_2 = 0, 1 \leq k \leq p \right\}$$

$$\hat{\mathcal{I}}_{x,V} = \left\{ k : \left\| \hat{\boldsymbol{\theta}}_k \right\|_2 \neq 0, 1 \leq k \leq p \right\}$$

Breheny and Huang (2015) built up an R package **grpreg** to solve nonconvex penalized regression problems with grouped predictors through group coordinate descent algorithms. We utilize **grpreg** in our R package **GPLSVCM** to implement the adaptive group lasso. In terms of choosing the regularization parameter ρ , traditional methods such as K-fold cross-validation (CV) and generalized cross-validation (GCV) can be used. In addition to those, we consider the Bayesian information criteria (BIC) that is widely applied in model selection for high-dimensional data, see Wang et al. (2007) and Lee et al. (2014). We also consider a modified version of BIC (mBIC) to accommodate our model structure. Denote $\hat{\mathcal{I}}_{\rho,c}$ and $\hat{\mathcal{I}}_{\rho,v}$ as the identified index set of constant and varying coefficients that corresponds to ρ , we define the following two criterion:

$$\text{BIC}(\rho) = LL_\rho + \left\{ \left| \hat{\mathcal{I}}_{\rho,c} \right| + \left| \hat{\mathcal{I}}_{\rho,v} \right| M_n \right\} \times \log(n) \quad (3.8)$$

$$\text{mBIC}(\rho) = LL_\rho + \left\{ \left| \hat{\mathcal{I}}_{\rho,c} \right| \log(p) + \left| \hat{\mathcal{I}}_{\rho,v} \right| M_n \log(p M_n) \right\} \times \frac{\log(n)}{2} \quad (3.9)$$

where LL_ρ is the deviance for a given ρ .

3.4.2 Simulation Study 2 (Model Selection)

In this simulation study, we investigate the performance of our proposed model selection method with respect to model identification and prediction accuracy. The oracle model is considered as a benchmark for comparing the prediction accuracy. Note that we know the true model structure and irrelevant covariates in the oracle model. We generate simulated datasets from the following three model structures:

$$\text{Model I : } Y_i = \beta_0 + \sum_{l=1}^{p_1} x_{il}\beta_l + \sum_{k=1}^{p_2} z_{ik}\alpha_k(\mathbf{u}_i) + \epsilon_i, \quad \epsilon_i \sim N(0, 0.5), i = 1, \dots, n$$

$$\text{Model II : } Y_i \sim \text{Poisson}(\mu_i), \log(\mu_i) = \beta_0 + \sum_{l=1}^{p_1} x_{il}\beta_l + \sum_{k=1}^{p_2} z_{ik}\alpha_k(\mathbf{u}_i), i = 1, \dots, n$$

$$\text{Model III : } Y_i \sim \text{Bernoulli}(\mu_i), \text{logit}(\mu_i) = \beta_0 + \sum_{l=1}^{p_1} x_{il}\beta_l + \sum_{k=1}^{p_2} z_{ik}\alpha_k(\mathbf{u}_i), i = 1, \dots, n$$

In above models, z_{ik} and x_{il} are independently simulated from $\text{Unif}[-1, 1]$ for $i = 1, \dots, n$, $l = 1, \dots, p_1$ and $k = 1, \dots, p_2$. $\alpha_1(\cdot)$ is the the surface test function used in [Wood et al. \(2008b\)](#) and $\alpha_2(u_1, u_2) = 1.5 \sin\left(\frac{\pi}{4}u_1u_2\right)$. Fixing $p_2 = 2$ and $\beta_0 = -2$, we consider the following four scenarios for p_1 :

- (i) $p = 4$ with $p_1 = 2$: $\beta_1 = -1$ and $\beta_2 = 2$.
- (ii) $p = 10$ with $p_1 = 8$: $\beta_1 = -1, \beta_2 = 2, \beta_3$ to $\beta_8 = 0$.
- (iii) $p = 20$ with $p_1 = 18$: $\beta_1 = -1, \beta_2 = 2, \beta_3$ to $\beta_{18} = 0$.
- (iv) $p = 100$ with $p_1 = 98$: $\beta_1 = -1, \beta_2 = 2, \beta_3$ to $\beta_{98} = 0$.

For each scenario in Model I and II, we take sample size $n = 2000$ and 5000 , and select the regularization parameter ρ through the BIC in (3.8). For Model III, we consider sample size $n = 5000$ and use GCV to select ρ . To construct the basis evaluation matrix, we use Bernstein polynomials of $d = 2$ and $r = 1$ over three triangulations: Δ_1 with 87 triangles and 73 vertices, Δ_2 with 165 triangles and 120 vertices, and Δ_3 with 238 triangles and 161 vertices. To evaluate our method, we adopt the following measures based on 100 replicates:

(c-i) The percentage of the selected model structures that are correctly identified among all the replications (“Corr”).

(c-ii) 10-fold cross-validation mean squared prediction error (CV-MSPE).

Note that (c-i) measures the accuracy of model selection, (c-ii) measures the prediction accuracy of the selected model.

In **GPLSVCM**, the function `gplsvcm_aglasso` implements the proposed model selection and corresponding model fitting at one time. The following code demonstrate the model fitting with this function:

```
R> uvpop=cbind(pop$u, pop$v)
R> # set up for shrinkage parameters in the penalty term:
R> lambda_start=0.1; lambda_end=0.2; nlambdas=50;
R> lambda1=exp(seq(log(lambda_start), log(lambda_end), length.out=nlambdas))
R> lambda2=exp(seq(log(lambda_start), log(lambda_end), length.out=nlambdas))
R> mfit=gplsvcm_aglasso(Y=Y, X=X, uvpop=uvpop, U=U, index=ind_s, V=V, Tr=Tr,
+ d=2, r=1, penalty="agrLasso", criteria="BIC", lambda1=lambda1, lambda2=lambda2,
+ family=family, lambda=lambda)
```

Table 3.1 summarizes the model selection results for both adaptive group lasso estimators and group lasso estimators. Table 3.2 compares our doubly penalized approach to the oracle model in terms of prediction accuracy. As demonstrated in Table 3.1, the model selection accuracy increases as the sample size increases for both adaptive group lasso estimators and group lasso estimators. However, we observe that group lasso estimators can identify the true model structure only for Gaussian and Poisson models that are devoid of irrelevant variables. When extraneous covariates are present, adaptive group lasso estimators significantly improve the inferior performance of group lasso estimators. Particularly for the continuous response, adaptive group lasso estimators can identify the underlying model structure with 100% accuracy regardless of triangulations or sample sizes. As for the noncontinuous response, adaptive group lasso estimators can nearly perfectly identify the true model for sample size $n = 5000$. While for sample size $n = 2000$, refining the triangulation might

reduce the model selection accuracy of adaptive group lasso estimators. In terms of the prediction accuracy, Table 3.2 suggests that the CV-MSPE obtained using our proposed approach is almost identical to those obtained from oracle model in all the model settings, and the prediction accuracy improves with increasing sample size.

Table 3.1: *Results of Model Structure Identification.*

Family	n	Dimension	Corr (agL)			Corr (gL)		
			Δ_1	Δ_2	Δ_3	Δ_1	Δ_2	Δ_3
Gaussian	2000	$p = 4$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		$p = 10$	1.0000	1.0000	1.0000	0.1900	0.1300	0.1400
		$p = 20$	1.0000	1.0000	1.0000	0.0200	0.0300	0.0200
		$p = 100$	1.0000	1.0000	1.0000	0.0000	0.0000	0.0000
	5000	$p = 4$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		$p = 10$	1.0000	1.0000	1.0000	0.4900	0.4600	0.4700
		$p = 20$	1.0000	1.0000	1.0000	0.1100	0.0900	0.1000
		$p = 100$	1.0000	1.0000	1.0000	0.0000	0.0000	0.0000
Poisson	2000	$p = 4$	1.0000	1.0000	0.9900	0.9900	0.9800	0.9600
		$p = 10$	1.0000	0.9600	0.9000	0.2100	0.0200	0.0100
		$p = 20$	1.0000	0.9500	0.8600	0.0000	0.0000	0.0000
		$p = 100$	1.0000	0.9300	0.8300	0.0000	0.0000	0.0000
	5000	$p = 4$	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		$p = 10$	1.0000	1.0000	1.0000	0.3100	0.2600	0.3400
		$p = 20$	1.0000	1.0000	0.9900	0.0500	0.0300	0.0400
		$p = 100$	1.0000	1.0000	1.0000	0.0000	0.0000	0.0000
Bernoulli	5000	$p = 4$	1.0000	1.0000	1.0000	0.3800	0.4100	0.4000
		$p = 10$	1.0000	1.0000	1.0000	0.0200	0.0400	0.0200
		$p = 20$	0.9900	0.9800	0.9600	0.0000	0.0000	0.0000
		$p = 100$	0.9800	0.9700	0.9700	0.0000	0.0000	0.0000

Note: Corr (agL): the percentage of the selected model structures that are correctly identified among all the replications by adaptive group lasso; Corr (gL): the percentage of the selected model structures that are correctly identified among all the replications by group lasso.

3.5 Application to NYC Crime Data

In this section, we use a New York City crime dataset to demonstrate the application of our proposed methodology in **GPLSVCM**.

Table 3.2: *Results of CV-MSPE.*

Family	n	Dimension	MSPE (agL)			MSPE (Oracle)		
			Δ_1	Δ_2	Δ_3	Δ_1	Δ_2	Δ_3
Gaussian	2000	$p = 4$	0.2614	0.2629	0.2618	0.2614	0.2629	0.2618
		$p = 10$	0.2614	0.2629	0.2618	0.2614	0.2629	0.2618
		$p = 20$	0.2614	0.2629	0.2618	0.2614	0.2629	0.2618
		$p = 100$	0.2614	0.2629	0.2618	0.2614	0.2629	0.2618
	5000	$p = 4$	0.2552	0.2564	0.2556	0.2552	0.2564	0.2556
		$p = 10$	0.2552	0.2564	0.2556	0.2552	0.2564	0.2556
		$p = 20$	0.2552	0.2564	0.2556	0.2552	0.2564	0.2556
		$p = 100$	0.2552	0.2564	0.2556	0.2552	0.2564	0.2556
Poisson	2000	$p = 4$	1.9398	2.1413	2.1534	1.9398	2.1413	2.1271
		$p = 10$	1.9398	1.6777	2.2968	1.9398	2.1413	2.1271
		$p = 20$	1.9398	2.0649	2.8050	1.9398	2.1413	2.1271
		$p = 100$	1.9398	1.9986	2.8009	1.9398	2.1413	2.1271
	5000	$p = 4$	1.3039	1.3128	1.2932	1.3039	1.3128	1.2932
		$p = 10$	1.3039	1.3128	1.2932	1.3039	1.3128	1.2932
		$p = 20$	1.3039	1.3128	1.3272	1.3039	1.3128	1.2932
		$p = 100$	1.3039	1.3128	1.2932	1.3039	1.3128	1.2932
Bernoulli	5000	$p = 4$	0.1146	0.1148	0.1147	0.1146	0.1148	0.1147
		$p = 10$	0.1146	0.1148	0.1147	0.1146	0.1148	0.1147
		$p = 20$	0.1173	0.1174	0.1173	0.1146	0.1148	0.1147
		$p = 100$	0.1139	0.1140	0.1141	0.1146	0.1148	0.1147

Note: MSPE (agL): the mean square prediction error calculated based on 10-fold cross-validation for the selected model by using adaptive group lasso; MSPE (Oracle): the mean square prediction error calculated based on 10-fold cross-validation for true model

Crimes are always a serious problem in society. In recent years, there has been an increase in the news of murder, theft, and sensational trials. The analysis of crime data has evolved into a critical component of law enforcement's attempts to improve public safety, recognize emerging patterns, allocate resources, and develop crime-prevention tactics. Felonies are the most severe kinds of crimes. Generally, a crime is considered a felony when it is punishable by more than a year in state prison (also called a penitentiary). Examples of felonies are murder, rape, burglary, and the sale of illegal drugs ([Bergman and Berman, 2022](#)). In this section, we aim to investigate how some demographic and economic factors impact felony frequency when considering the spatial effect.

In this study, we collect the felony frequency within each census block group of New York City in 2016 from the official website of the city of New York, see Figure 3.3. A census block group is the smallest geographical unit used by the United States Census Bureau to publish sample data. Typically, a block group has a population of 600 to 3000 people. Additionally, we acquire some demographic and socioeconomic data as the explanatory covariates from U.S. Census Bureau. See Table 3.3 for detailed information of the covariates. We remove a few outliers and areas with no population from the analysis because the frequencies of felony crimes in those areas are generally zero. The dataset contains 5779 census block groups with the felony frequency as the response variable and eight covariates as in Table 3.3.

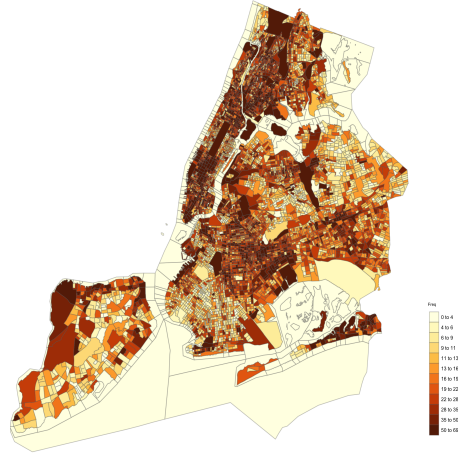


Figure 3.3: *Felony Crimes within census block group in NYC.*

Table 3.3: *Description of the covariates in the 2016 NYC Felony Crime dataset.*

Variable	Description
Population*	Total population
Poverty	Proportion of people whose income is under poverty level in population
Marriage	Proportion of people in a marriage in population
Youth	Proportion of people aged 14-25 years old in population
Unemployed	Proportion of unemployed people in population
Education	Proportion of people with a higher degree than high school in population
Household	Proportion of people in a household in population
Male	Proportion of males in population

Note: The covariates with * are transformed from the original value by $X^* = \log(X + 0.01)$.

Let Y be the response variable, we assume the felony frequency follows a Poisson distri-

bution, i.e. $Y_i \sim \text{Poisson}(\mu_i)$ for $i = 1, \dots, 5779$. We first implement our proposed doubly penalized approach to identify the covariates with spatially varying effects and constant effects, respectively, and then obtain the coefficient estimates of the selected model. In our analysis, we adopt a triangulation with 496 triangles and 304 vertices as in Figure 3.4, and we choose Bernstein polynomial splines of degree $d = 2$ and smoothness $r = 1$. All of the covariates are normalized in the analysis.

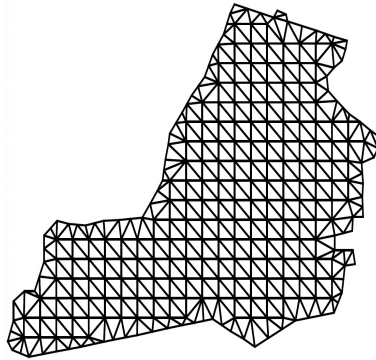


Figure 3.4: *Plot of triangulation on the domain of NYC.*

In the following code, we demonstrate the application of **GPLSVCM** to the crime dataset. Note that the matrix of the boundary of New York City and our crime dataset is named as **NYCBoundary** and **NYCfelony** in R, respectively. The column names of **NYCBoundary** are **lon** and **lat**, which correspond to the longitudes and latitudes of the boundary locations in New York City. Moreover, the triangulation for New York City is constructed using the **TriMesh** function in **Triangulation**:

```
R> library(GPLSVCM)
R> # Generate a uniform triangulation:
R> TriNYC<- TriMesh(as.matrix(NYCBoundary), n = 18)
R> Tr<-TriNYC$Tr; V<-TriNYC$V
R> # Set up candidate vector of smoothness penalty parameter:
R> lambda<- exp(seq(log(0.001),log(10),length.out=10))
R> Y<- NYCfelony$Freq
R> X<- as.matrix(NYCfelony[,c("Population","Poverty","Marriage","Youth",
```

```

+   "Unemployed", "Household", "Education", "Male"]])
R> U<- as.matrix(NYCfelony[,c("lon", "lat")])
R> MIDFNYCfelony<- gplsvcm.aglasso(Y,X,U,V,Tr,d=2,r=1,family=poisson(),
+   lambda=lambda)
R> MIDFNYCfelony$constant_varname
R> [1] "intercept" "Population"
R> MIDFNYCfelony$varying_varname
R> [1] "Poverty" "Marriage" "Youth" "Unemployed" "Education" "Household"
+   "Male"

```

The object `constant_varname` and `varying_varname` return the identified covariate names that has constant and spatially varying effect on the response variable, respectively. The `intercept` and covariate `Population` are identified as being linearly associated with the felony frequency, while all the other covariates are identified as having a spatially varying impact on the felony frequency. As a result, our proposed model selection procedure selects the following model for our dataset:

$$\begin{aligned}
Y_i &\sim \text{Poisson}(\mu_i), i = 1, \dots, 5779 \\
\log(\mu_i) &= \beta_0 + \beta_1 \text{Population}_i^* + \alpha_1(\mathbf{u}_i) \text{Poverty}_i + \alpha_2(\mathbf{u}_i) \text{Marriage}_i \\
&+ \alpha_3(\mathbf{u}_i) \text{Youth}_i + \alpha_4(\mathbf{u}_i) \text{Unemployed}_i + \alpha_5(\mathbf{u}_i) \text{Household}_i \\
&+ \alpha_6(\mathbf{u}_i) \text{Education}_i + \alpha_7(\mathbf{u}_i) \text{Male}_i
\end{aligned}$$

where β_0 and β_1 are the unknown linear coefficients and $\alpha_k(\cdot)$ are the unknown coefficient functions for $k = 1, \dots, 7$.

As the `gplsvcm.aglasso` function integrates model structure identification and model fitting, we can extract the estimation of the selected model from `MIDFNYCfelony`:

```
R> fitNYCfelony<- MIDFNYCfelony$mfit
```

Then, the estimation of linear coefficients, coefficient functions, and spline coefficients can be extracted from `fitNYCfelony` accordingly:

```
R> fitNYCfelony$beta_hat
```

```
R> fitNYCfelony$alpha_hat
```

```
R> fitNYCfelony$theta_hat
```

On this basis, the MSE of the response variable can be obtained from `fitNYCfelony`:

```
R> mean((fitNYCfelony$fitted_values-fitNYCfelony$Y)^2,na.rm = TRUE)
```

```
R> [1] 107.9877
```

To assess the prediction performance of the fitted model, we use the function `cv_gplsvcm` in **GPLSVCM** to compute the 10-fold cross validation prediction error:

```
R> cv_gplsvcm(Y,X,fitNYCfelony$ind_l,fitNYCfelony$ind_nl,U,V,Tr,d=2,r=1,
```

```
R> + lambda=lambda,family=poisson())
```

```
R> [1] 122.9637
```

As a comparison, we also calculate the MSE and CV-MSPE for the GLM fitted to the dataset. We show the results in Table 3.4. It can be observed that BPST-agL exceeds GLM remarkably in both model estimation and prediction accuracy. The inferior performance of GLM may be due to the lack of accounting for the spatial nonstationarity in the data.

Table 3.4: *Model Selection, Estimation, and Prediction for NYC Felony Crime Data.*

Model	MSE	MSPE
BPST-agL	107.9877	122.9637
GLM	150.2400	151.1791

Note: BPST-agL: bivariate penalized spline over triangulation with adaptive group lasso approach for model selection; GLM: generalized linear model is used as a benchmark; MSPE: mean square prediction error is calculated based on 10-fold cross-validation.

At last, we use the `gplsvcm_plot` function in **GPLSVCM** to obtain the estimated coefficient functions:

```
R> gplsvcm_plot(fitNYCfelony,gridnumber=500,xlab=rep("lon",7),
```

```
R> + ylab=rep("lat",7),main=rep(expression(paste("The Estimated Surface for"
```

```
R> + , " ",hat(alpha))),7))
```

In this function, the `gridnumber` controls the fineness of plots, `xlab` and `ylab` are for specifying the vectors of labels of the x and y axis, respectively, for all the estimated coefficient functions, and `main` is for providing the vector of titles of all the estimated coefficient func-

tions.

The estimated surfaces for bivariate functions α_k , $k = 1, \dots, 7$ are depicted in Figure 3.5. “Poverty” generally positively affects felony frequency except in some areas of western Brooklyn, a small area of Bronx near Long Island Sound, and areas near Nassau County in Queens. Especially for the very southwestern part of Staten Island and areas near Atlantic Ocean in Queens, the impact of “Poverty” is the largest. Compared to other areas, western Staten Island and a tiny area along the Bronx’s northern border have a substantially larger positive effect of “Marriage” on felony frequency, while the southeastern part of NYC has a relatively higher negative effect. For “Youth”, we observe its largest impact in southwestern Staten Island. Additionally, in the eastern Bronx and western Staten Island, it seems that the unemployment rate plays a more prominent role in felony crimes. It is also indicated that the effect of “Household” on felony frequency is between the levels $\exp(-2)$ and $\exp(1.5)$ in most areas. Expectedly, a higher educational attainment ratio mainly associates with fewer felonies in NYC, particularly in Tottenville and Conference House Park. Moreover, a larger number of males seems to relate to more felonies in most areas. A comparatively larger positive impact of “Male” is found in Manhattan, the border between Staten Island and Brooklyn near Upper New York Bay, and Arthur Kill, whereas a strikingly larger negative impact of “Male” happens in southern Staten Island.

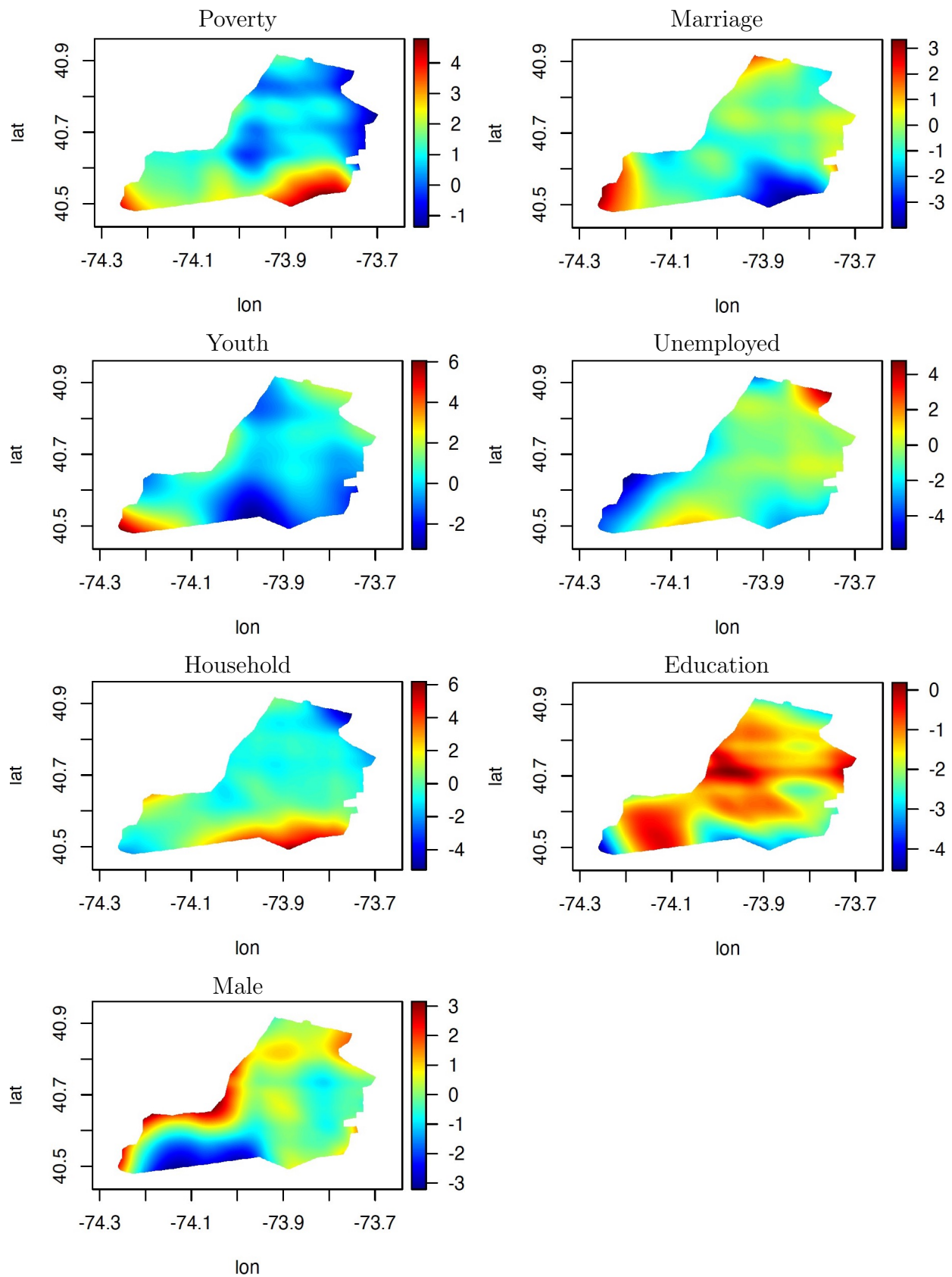


Figure 3.5: Contour plots of the estimated coefficient functions on the domain of NYC

3.6 Concluding Remarks

In this article, we have introduced a new R package **GPLSVCM** which is established for integrating variable selection, model structure selection, model fitting, and predictive inference of the proposed GPLSVCMs. We show how to use the package to implement the BPST method to obtain the estimators for varying coefficient functions and linear coefficients. To adapt to high dimensional data, we proposed a unified approach via adaptive group lasso for simultaneously selecting significant predictors and distinguishing between the predictors with functional coefficients and constant coefficients. To obtain information about the variability of the prediction of observation in the future, we proposed jackknife, jackknife+, and CV+ methods, respectively, for computing the prediction interval of a data point with the desired coverage level. The proposed prediction intervals are theoretically reliable with an assumption-free guarantee and practically feasible.

We illustrate both functionalities through a series of simulation studies and an application to the 2016 NYC crime dataset coupled with code demonstrations. Both our simulation and application results confirm that users can obtain a fitted model with excellent estimation and prediction performance by using our package compared with alternatives. Moreover, **GPLSVCM** enables convenient and user-friendly implementations of GPLSVCMs in real-world data analyses. In future work, our package can be further developed to incorporate the temporal effect in the model.

Chapter 4

Spatial Generalized Autoregressive Conditional Heteroskedasticity Varying Coefficient Models

4.1 Introduction

Volatility was initially introduced to measure the variation of time series data, for example, stock market values, oil prices, exchange rates, and GDP. Modelling volatility can help decision-makers estimate the risk associated with a product in the financial and economic markets. The autoregressive conditional heteroskedasticity model (ARCH) introduced by [Engle \(1982\)](#) established the foundation for modelling the volatility of time series data. Subsequent to this pioneering work, [Bollerslev \(1986\)](#) proposed a generalized version of ARCH models known as the generalized autoregressive conditional heteroskedasticity model (GARCH). Both models have broad applications, such as investment selection, option pricing, and risk management.

Recent advances in spatial econometrics have emphasized on spatial volatility, which is defined as the conditional variance given the data in adjacent regions. Forecasting locally varying risk in geographically constrained markets such as real estate, labor, and stocks

focuses substantially on modelling spatial volatility. However, only a few models have been developed to model volatility over space. For example, [Yan \(2007\)](#) introduced a spatial stochastic volatility (SSV) component into the conditional autoregressive model (CAR) to capture the spatial heteroscedasticity for lattice data. [Robinson \(2009\)](#) proposed a spatial extension of time series volatility models in terms of ARCH model. As a more direct extension of ARCH models in the spatial context, [Sato and Matsuda \(2017\)](#) introduced the spatial ARCH model (S-ARCH). Later, the S-ARCH model was further extended to the spatial GARCH model (S-GARCH) in [Sato and Matsuda \(2021\)](#) analogous to the extension of ARCH models to GARCH models. The S-GARCH model is effective in understanding volatility behavior over space because it can capture the characteristics of volatility, such as volatility clustering and spillover effect, which refers to an increase in volatility in one region leading to an increase or decrease in neighboring regions. Despite those encouraging works, study on nonstationary relationship between some explanatory variables and the volatility over space had been a largely underexplored yet prospective domain.

In this article, we fill this research gap by considering a spatial generalized autoregressive conditional heteroskedasticity varying coefficient model (SGARCH-VCM). For $i = 1, \dots, n$, let y_i represent the observed spatial data without level components, $\mathbf{X}_i = (X_{i1}, \dots, X_{ip})^\top$ be the vector of explanatory variables, and $\mathbf{U}_i = (U_{i1}, U_{i2})^\top$ be the location of the i -th observation distributed over a spatial domain Ω with an arbitrary shape. Denote h_i as the unobservable spatial volatility of our interest, the SGARCH-VCM is then defined as:

$$\begin{aligned} y_i &= \sqrt{h_i} \varepsilon_i, \quad i = 1, \dots, n, \\ \log h_i &= \lambda \sum_{j=1}^n w_{ij} \log h_j + \rho \sum_{j=1}^n w_{ij} \log y_j^2 + \alpha + \sum_{k=1}^p X_{ik} \beta_k(\mathbf{U}_i), \end{aligned} \quad (4.1)$$

where w_{ij} is the weight of the neighboring effects that satisfies $w_{ii} = 0$ and $\sum_{j \neq i} w_{ij} \geq 0$ for any $i, j = 1, \dots, n$, ε_i 's are i.i.d. random variable with $E(\varepsilon_i) = 0$ and $\text{Var}(\varepsilon_i) = 1$. (λ, ρ, α) are the unknown parameters while $\beta_k(\cdot)$'s are unknown varying coefficient functions to be estimated. This model would assist with investigating how some observable exogenous

variables influence the volatility behavior over a spatial domain while modelling spatial volatility.

Estimation of the model in 4.1 can be challenging for a complicated domain. On the one hand, the determination of weight matrix $\mathbf{W} \equiv \{w_{ij}\}_{i,j=1}^n$ should be carefully handled, because conventional methods based on calculating the Euclidean distance between different locations are not appropriate for a domain that is irregularly shaped and typically excludes the line segment that connects two points. On the other hand, estimating the varying coefficient functions for an increasing volume of spatial data requires a computationally efficient approximation method while overcoming the “leakage” problem over a complicated domain. For SVCs, three major approaches have been widely used to estimate the coefficient functions: kernel based approach (Brunsdon, Fotheringham, and Charlton, 1996; Fotheringham, Brunsdon, and Charlton, 2003), Bayesian method (Gelfand, Kim, Sirmans, and Banerjee, 2003; Assunção, 2003), and spline smoothing technique (Mu, Wang, and Wang, 2018). The first two approaches only work well for rectangular domains. For large-scale data, they are almost computationally impossible on a regular computer. By contrast, the third approach involves a low computational cost and is favorable to complicated domains. To estimate SGARCH-VCs, we propose to approximate the varying coefficient functions by the bivariate penalized spline over triangulation (BPST) method and calculate the weight matrix by approximated geodesic distances via a triangulation technique based algorithm developed in Mu et al. (2020). More specifically, we have developed a global two-stage quasi-likelihood maximization approach based on the BPST method to estimate the unknown global parameters and coefficient functions.

The rest of the article is organized as follows. In Section 4.2, we demonstrate our two-stage estimation procedure. In Section 4.3, we show simulation studies comparing our method with its competitors over both rectangular and horseshoe domains. In Section 4.4, we illustrate our model and estimation method using a Florida median housing price dataset.

4.2 The Estimation Method

4.2.1 Model Representation

Denote $\log \mathbf{y}^2 = (\log y_1^2, \dots, \log y_n^2)'$, $\log \mathbf{h} = (\log h_1, \dots, \log h_n)'$, $\log \boldsymbol{\varepsilon}^2 = (\log \varepsilon_1^2, \dots, \log \varepsilon_n^2)'$ and $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^\top$ where $\mu_i = \alpha + \mathbf{X}_i^\top \boldsymbol{\beta}(\mathbf{U}_i)$ and $\boldsymbol{\beta}(\mathbf{u}) = (\beta_1(\mathbf{u}), \dots, \beta_p(\mathbf{u}))^\top$. Denote $\mathbf{W} = (w_{ij})$ as the $n \times n$ spatial weight matrix, the model in 4.1 has the following matrix representation:

$$\begin{aligned} \log \mathbf{y}^2 &= \log \mathbf{h} + \log \boldsymbol{\varepsilon}^2, \\ \log \mathbf{h} &= \lambda \mathbf{W} \log \mathbf{h} + \rho \mathbf{W} \log \mathbf{y}^2 + \boldsymbol{\mu}. \end{aligned} \tag{4.2}$$

Given the common assumption that $|\lambda| < 1$, one can easily obtain that:

$$\log \mathbf{y}^2 = (\lambda + \rho) \mathbf{W} \log \mathbf{y}^2 + \boldsymbol{\mu} + (\mathbf{I}_n - \lambda \mathbf{W}) \log \boldsymbol{\varepsilon}^2. \tag{4.3}$$

Note that for SGARCH-VCMS, we require $|\lambda + \rho| < 1$ because it is a sufficient condition that guarantees the model identifiability.

To estimate the model in 4.3, traditional maximum likelihood estimation is inappropriate here because $\log \varepsilon_i^2$ may not follow a normal distribution with a zero mean. The estimation of α may be inconsistent if we assume $E(\log \varepsilon_i^2) = 0$. To address this issue, we develop a two-stage procedure where we estimate $(\lambda, \rho)'$ and $(\beta_1(\cdot), \dots, \beta_p(\cdot))'$ using a quasi-likelihood at the first stage, and then estimate α using another quasi-likelihood at the second stage. Denote $a = E \log \varepsilon_i^2$, $\mathbf{Y} = \log \mathbf{y}^2$ and $\mathbf{V} = \log \boldsymbol{\varepsilon}^2 - a \mathbf{1}_n$ where $\mathbf{1}_n = (1, \dots, 1)'$, then the model in 4.3 can be represented as:

$$\mathbf{Y} = (\lambda + \rho) \mathbf{W} \mathbf{Y} + \boldsymbol{\mu} + (\mathbf{I}_n - \lambda \mathbf{W}) \mathbf{V}. \tag{4.4}$$

At the estimation of the first stage, we treat $\mathbf{V}$ as a random variable following a multivariate normal distribution with mean vector $\mathbf{0}$ and variance-covariance matrix $\sigma^2 \mathbf{I}_n$. Denote $\mathbf{T}(\lambda) =$

$\mathbf{I}_n - \lambda \mathbf{W}$ and $\mathbf{T}(\lambda, \rho) = \mathbf{I}_n - (\lambda + \rho) \mathbf{W}$, then

$$\mathbf{T}(\lambda, \rho) \mathbf{Y} = \boldsymbol{\mu} + \mathbf{T}(\lambda) \mathbf{V}. \quad (4.5)$$

We further denote

$$\mathbf{Y}(\lambda, \rho) = (Y_1(\lambda, \rho), \dots, Y_n(\lambda, \rho))^\top = \mathbf{T}^{-1}(\lambda) \mathbf{T}(\lambda, \rho) \mathbf{Y},$$

then 4.5 is equivalent to:

$$\mathbf{Y}(\lambda, \rho) = \mathbf{T}^{-1}(\lambda) \boldsymbol{\mu} + \mathbf{V}. \quad (4.6)$$

4.2.2 Bivariate Penalized Spline over Triangulation

As the first stage involves estimating the unknown coefficient functions, we consider a smoothing technique to approximate the coefficient functions, that is, penalized bivariate spline over triangulations. In the following, we introduce the idea and implementation of BPST in the first stage estimation of the model.

Triangulation is a useful technique for dealing with data dispersed over complex domains with irregular boundaries and inner holes, as shown in [Lai and Wang \(2013\)](#). Denote τ as a triangle over a hyperplane, then a collection of $\Delta = \{\tau_1, \dots, \tau_K\}$ of K triangles is defined as a triangulation of a domain $\Omega = \cup_{j=1}^K \tau_j$ if a pair of intersected triangles in Δ satisfies that either their intersection is a common vertex or a common edge. Then we define for a triangulation Δ and nonnegative integers d, r , $\mathcal{S}_d^r(\Delta) = \{s \in \mathbb{C}^r(\Omega) : s|_\tau \in \mathbb{P}_d(\tau), \tau \in \Delta\}$ where $\mathbb{C}^r(\Omega)$ is the collection of all r -th continuously differentiable functions over Ω , $s|_\tau$ is the polynomial piece of spline s restricted on triangle τ , and $\mathbb{P}_d$ is the space of all polynomials of degree less than or equal to d . To achieve the optimal approximation rate, we choose the spline space with $d = 3r + 2$ as advised in [Lai and Schumaker \(2007\)](#). For notation simplicity, we denote the spline space $\mathcal{S}_{3r+2}^r(\Delta)$ as $\mathcal{S}$.

Denote $\mathbf{T}^{-1}(\lambda) = \{T_{ij}^{-1}(\lambda)\}_{i,j=1}^n$ and $T_i = \sum_{j=1}^n T_{ij}^{-1}(\lambda)$. Let $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_n)^\top$, write $\mathbf{T}^{-1}(\lambda) \mathbf{X} = (\mathbf{X}_1(\lambda), \dots, \mathbf{X}_n(\lambda))^\top$ where $\mathbf{X}_i(\lambda) = (X_{i1}(\lambda), \dots, X_{ip}(\lambda))^\top$ for $i = 1, \dots, n$.

Given $\{(\mathbf{U}_i, T_i, \mathbf{X}_i(\lambda), Y_i(\lambda, \rho))\}_{i=1}^n$ for a fixed λ and ρ , we consider the optimization problem as follows:

$$\min_{\substack{c_0 \in \mathcal{R} \\ g_k \in \mathcal{S}, k=1, \dots, p}} \sum_{i=1}^n \left\{ Y_i(\lambda, \rho) - T_i c_0 - \sum_{k=1}^p X_{ik}(\lambda) g_k(\mathbf{U}_i) \right\}^2 + \sum_{k=1}^{p_2} \theta_k \mathcal{E}(g_k),$$

where $g_k(\cdot), k = 1, \dots, p$, can be approximated by bivariate splines and $\theta_k > 0, k = 1, \dots, p$ are the penalty parameters that regularize how smooth the function g_k is and $\mathcal{E}(g_k) = \int_{\Omega} \left\{ (D_{u_1}^2 g_k)^2 + 2(D_{u_1} D_{u_2} g_k)^2 + (D_{u_2}^2 g_k)^2 \right\} du_1 du_2$ where $D_{u_j}^q f(\mathbf{x})$ is the q -th order derivative in the direction u_j at the point $\mathbf{u}$ for directions u_1 and u_2 . For any $k = 1, \dots, p$, denote $\{B_j\}_{j \in \mathcal{J}_k}$ the set of bivariate Bernstein basis polynomials for $\mathcal{S}$. Then $g_k(\mathbf{u})$ can be represented as the linear combination of Bernstein basis: $g_k(\mathbf{u}) = \sum_{j \in \mathcal{J}_k} B_{kj}(\mathbf{u}) c_{kj} = \mathbf{B}_k(\mathbf{u})^\top \mathbf{c}_k$, where $\mathbf{c}_k = (c_{kj}, j \in \mathcal{J}_k)^\top$ is the spline coefficient vector. In order to guarantee the smoothness of the spine fit, we impose some constraints on the spline coefficients using a smoothness matrix $\mathbf{H}_k$ on each coefficient $\mathbf{c}_k$. The smoothness matrix is related to the triangulation structure and choice of d and r . For the details of constructing $\mathbf{H}_k$, we refer to [Zhou and Pan \(2014\)](#).

Without loss of generality, we assume $\mathbf{B}(\mathbf{u}) = \mathbf{B}_1(\mathbf{u}) = \dots = \mathbf{B}_p(\mathbf{u}) = \{B_j(\mathbf{u})\}_{j \in \mathcal{J}}$ and $\mathbf{H} = \mathbf{H}_1 = \dots = \mathbf{H}_p$ for a convenient and explicit explanation. Accordingly, we can consider the following constrained optimization problem:

$$\min_{\substack{c_0 \in \mathcal{R} \\ \mathbf{c}_k \in \mathcal{R}^{\mathcal{J}}, k=1, \dots, p}} \sum_{i=1}^n \left\{ Y_i(\lambda, \rho) - T_i c_0 - \sum_{k=1}^p X_{ik}(\lambda) \mathbf{B}(\mathbf{U}_i)^\top \mathbf{c}_k \right\}^2 + \sum_{k=1}^{p_2} \theta_k \mathbf{c}_k^\top \mathbf{P} \mathbf{c}_k,$$

subject to $\mathbf{H} \mathbf{c}_k = \mathbf{0}, k = 1, \dots, p.$

where $\mathbf{P}$ is the diagonal block penalty matrix satisfying that $\mathbf{c}_k^\top \mathbf{P} \mathbf{c}_k = \mathcal{E}(\mathbf{B}^\top \mathbf{c}_k)$. To solve the above constrained minimization problem, we consider a reparameterization of the spline coefficient $\mathbf{c}_k$. That is, we perform a QR decomposition on the smoothness matrix $\mathbf{H}$ so that we can obtain $\mathbf{H}^\top = \mathbf{Q} \mathbf{R} = (\mathbf{Q}_1 \quad \mathbf{Q}_2) \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \end{pmatrix}$ where $\mathbf{Q}$ is an orthogonal matrix, $\mathbf{Q}_1$ is the

first r columns of $\mathbf{Q}$ where r is the rank of $\mathbf{H}$, $\mathbf{R}_1$ is an upper triangle matrix, and $\mathbf{R}_2$ is a matrix of zeros. In light of this, the constrain can be removed by reparameterizing $\mathbf{c}_k$ as $\mathbf{c}_k = \mathbf{Q}_2 \mathbf{c}_k^*$, and we then obtain the following minimization problem in the form of a simple penalized regression problem:

$$\min_{\substack{c_0, \mathbf{c}_k^*, \\ k=1, \dots, p}} \sum_{i=1}^n \left\{ Y_i(\lambda, \rho) - T_i c_0 - \sum_{k=1}^p X_{ik}(\lambda) \mathbf{B}^\top(\mathbf{U}_i) \mathbf{Q}_2 \mathbf{c}_k^* \right\}^2 + \sum_{k=1}^p \theta_k \mathbf{c}_k^{*\top} \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2 \mathbf{c}_k^*.$$

Let $\mathbf{B}^*(\mathbf{U}_i) = \mathbf{Q}_2^\top \mathbf{B}(\mathbf{U}_i)$ and $\mathbf{c}^* = (\mathbf{c}_1^{*\top}, \dots, \mathbf{c}_p^{*\top})^\top$, denote $\mathcal{X}_B(\lambda) = (\mathbf{X}_1(\lambda) \otimes \mathbf{B}^*(\mathbf{U}_1), \dots, \mathbf{X}_n(\lambda) \otimes \mathbf{B}^*(\mathbf{U}_n))^\top$, $\mathcal{T} = (T_1, \dots, T_n)^\top$ and $\Phi = (\mathcal{T}, \mathcal{X}_B(\lambda))$. Moreover, denote $\Theta = \text{diag}(\theta_1, \dots, \theta_p)$, and $\mathbf{D}_\Theta = \text{diag}\{0, \Theta \otimes (\mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2)\}$, we obtain that

$$\begin{pmatrix} \hat{c}_0(\lambda, \rho) \\ \hat{\mathbf{c}}^*(\lambda, \rho) \end{pmatrix} = (\Phi^\top \Phi + \mathbf{D}_\Theta)^{-1} \Phi^\top \mathbf{Y}(\lambda, \rho) \quad (4.7)$$

As a result, we obtain the bivariate spline estimator of $\mathbf{c}_k$ and $\beta_k(\cdot)$ as $\hat{\mathbf{c}}_k(\lambda, \rho) = \mathbf{Q}_2 \hat{\mathbf{c}}_k^*(\lambda, \rho)$ and $\hat{\beta}_k(\mathbf{u}; \lambda, \rho) = \mathbf{B}(\mathbf{u})^\top \hat{\mathbf{c}}_k(\lambda, \rho)$ respectively.

Regarding the choice of penalty parameters, we use the generalized cross-validation criterion (GCV) given as:

$$\text{GCV} = n(n - m)^{-2} \|\mathbf{Y}(\lambda, \rho) - \mathbf{S}_\Theta \mathbf{Y}(\lambda, \rho)\|^2,$$

where $\mathbf{S}_\Theta = \Phi (\Phi^\top \Phi + \mathbf{D}_\Theta)^{-1} \Phi^\top$ is the projection matrix and $m = \text{tr}(\mathbf{S}_\Theta)$ is the effective degrees of freedom (edf) for the model. The optimal combination of $(\theta_1, \dots, \theta_p)$ is obtained by minimizing the GCV over a set of grid points.

4.2.3 Estimation of λ , ρ and $\beta(\cdot)$

We now proceed with the first stage estimation. As we are treating each element in $\mathbf{V}$ as i.i.d. random noise following $N(0, \sigma^2)$, the loglikelihood of the model 4.4 is then

$$L_n(\lambda, \rho, \boldsymbol{\mu}, \sigma^2) = -\frac{n}{2} \log(2\pi\sigma^2) + \log(|\mathbf{T}(\lambda, \rho)|) - \log(|\mathbf{T}(\lambda)|) - \frac{1}{2\sigma^2} \{\mathbf{Y}(\lambda, \rho) - \boldsymbol{\mu}\}^\top \{\mathbf{Y}(\lambda, \rho) - \boldsymbol{\mu}\}.$$

Plugging in $\hat{c}_0(\lambda, \rho)$ and $\hat{\mathbf{c}}^*(\lambda, \rho)$ in 4.7 into $L_n(\lambda, \rho, c_0, \mathbf{c}_k, \sigma^2)$, we have

$$\begin{aligned} L_n(\lambda, \rho, \boldsymbol{\mu}, \sigma^2) = & -\frac{n}{2} \log(2\pi\sigma^2) + \log(|\mathbf{T}(\lambda, \rho)|) - \log(|\mathbf{T}(\lambda)|) \\ & - \frac{1}{2\sigma^2} \{\mathbf{Y}(\lambda, \rho) - \hat{\boldsymbol{\mu}}(\lambda, \rho)\}^\top \{\mathbf{Y}(\lambda, \rho) - \hat{\boldsymbol{\mu}}(\lambda, \rho)\}, \end{aligned} \quad (4.8)$$

where $\hat{\boldsymbol{\mu}}(\lambda, \rho) = \boldsymbol{\Phi} (\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \mathbf{D}_\Theta)^{-1} \boldsymbol{\Phi}^\top \mathbf{Y}(\lambda, \rho)$. By taking the first order derivative of 4.8 with respect to σ^2 , we have

$$\hat{\sigma}^2(\lambda, \rho) = \frac{1}{n} \{\mathbf{Y}(\lambda, \rho) - \hat{\boldsymbol{\mu}}(\lambda, \rho)\}^\top \{\mathbf{Y}(\lambda, \rho) - \hat{\boldsymbol{\mu}}(\lambda, \rho)\}.$$

Plugging $\hat{\sigma}^2(\lambda, \rho)$ in 4.8, we can then obtain the concentrated loglikelihood function of λ and ρ :

$$L_n(\lambda, \rho) = -\frac{n}{2} (\log(2\pi) + 1) - \frac{n}{2} \log \hat{\sigma}^2(\lambda, \rho) + \log(|\mathbf{T}(\lambda, \rho)|) - \log(|\mathbf{T}(\lambda)|).$$

Then the maximum quasi-likelihood estimator of (λ, ρ) is:

$$(\hat{\lambda}, \hat{\rho}) = \arg \max_{\lambda, \rho \in [0, 1]} \left\{ -\frac{n}{2} \log \hat{\sigma}^2(\lambda, \rho) + \log(|\mathbf{T}(\lambda, \rho)|) - \log(|\mathbf{T}(\lambda)|) \right\}.$$

As the end of the first stage, we obtain the estimator of $\beta(\cdot)$ by substituting (λ, ρ) with $(\hat{\lambda}, \hat{\rho})$.

4.2.4 Estimation of α

We now move to the second stage to estimate α . At this stage, we consider a new likelihood function by directly treating ε_i in 4.3 as i.i.d. standard normal random variable. Let $\eta = (\lambda, \rho, \boldsymbol{\beta}(\cdot))'$ and $\mathbf{R} = \log \boldsymbol{\varepsilon}^2$, then

$$\begin{aligned} \mathbf{R}(\eta, \alpha) &= \mathbf{T}^{-1}(\lambda) (\mathbf{T}(\lambda, \rho) \mathbf{Y} - \boldsymbol{\mu}) \\ &= \mathbf{T}^{-1}(\lambda) (\mathbf{T}(\lambda, \rho) \mathbf{Y} - \boldsymbol{\mu}_\beta) - \frac{\alpha}{1 - \lambda} \mathbf{1}_n \\ &= \mathbf{C}(\eta) - \frac{\alpha}{1 - \lambda} \mathbf{1}_n, \end{aligned} \tag{4.9}$$

where $\boldsymbol{\mu}_\beta = (\mathbf{X}_1^\top \boldsymbol{\beta}(\mathbf{U}_1), \dots, \mathbf{X}_n^\top \boldsymbol{\beta}(\mathbf{U}_n))^\top$ and $\mathbf{C}(\eta) = \mathbf{T}^{-1}(\lambda) (\mathbf{T}(\lambda, \rho) \mathbf{Y} - \boldsymbol{\mu}_\beta)$. As a direct result of $\varepsilon_i \sim N(0, 1)$, it follows that the probability density function of $R_i = \log \varepsilon_i$ is:

$$f(R_i) = \frac{1}{\sqrt{2\pi}} \exp \left(-\frac{1}{2} \exp(R_i) + \frac{1}{2} R_i \right).$$

Therefore, one can obtain the following log quasi-likelihood function:

$$\begin{aligned} L_n(\eta, \alpha) &= -\frac{n}{2} \log 2\pi - \sum_{i=1}^n \left\{ -\frac{1}{2} \frac{\exp(C_i(\eta))}{\exp\left(\frac{\alpha}{1-\lambda}\right)} + \frac{1}{2} \left(C_i(\eta) - \frac{\alpha}{1-\lambda} \right) \right\} \\ &\quad + \log(|\mathbf{T}(\lambda, \rho)|) - \log(|\mathbf{T}(\lambda)|), \end{aligned} \tag{4.10}$$

where C_i is the i -th element of $\mathbf{C}(\eta)$. Setting the partial derivative of $L_n(\eta, \alpha)$ with respect to α to zero and solving the equation, we obtain the estimator of α given as:

$$\hat{\alpha} = (1 - \hat{\lambda}) \log \left(\frac{1}{n} \sum_{i=1}^n \exp(C_i(\hat{\eta})) \right),$$

where $\hat{\eta} = (\hat{\lambda}, \hat{\rho}, \hat{\boldsymbol{\beta}}(\cdot))'$ is the estimator obtained in the first stage estimation.

4.3 Simulation Studies

In this section, we conduct simulation studies to evaluate the finite-sample performance of the proposed two-stage estimation method for SGARCH-VCMs. We perform two simulation studies based on different spatial domains, one is a rectangular domain common in literature, and the other is a horseshoe domain with an irregular boundary. Besides our proposed method, we also implement the least square estimation based approach in [Sato and Matsuda \(2021\)](#) where all the covariates are assumed to have a constant effect on the spatial volatility. In what follows, we refer to our method as “BPST” and the method in [Sato and Matsuda \(2021\)](#) as “LSE”. We take LSE as a benchmark in the simulation studies.

In both simulation studies, we calculate the weight matrix using the exponential function, that is, $w_{ij} = \exp(-10d_{ij}) / \sum_{k \neq i} \exp(-10d_{ik})$ where d_{ij} measures the distance between two locations i and j . In the first simulation study, we consider a rectangular domain, the d_{ij} is therefore simply the Euclidean distance. In the second simulation study, we consider the horseshoe domain. Because it has a complex boundary so that Euclidean distance does not apply here, we adopt the algorithm in [Mu et al. \(2020\)](#) to calculate the geodesic distance based on the triangulation technique. Whereas for “LSE” in both examples, d_{ij} is the Euclidean distance.

To generate data from a SGARCH-VCM, instead of directly simulating y_i in [4.1](#), we simulate $\log y_i^2$ for $i = 1, \dots, n$ by an equivalent model as below:

$$\log y_i^2 = (\lambda + \rho) \sum_{j=1}^n w_{ij} \log y_j^2 + \alpha + X_i \beta(\mathbf{U}_i) + \log \varepsilon_i^2 - \lambda \sum_{j=1}^n w_{ij} \log \varepsilon_j^2. \quad (4.11)$$

In both simulation studies as follows, the explanatory variable X_i and the random noise ε_i are independently and identically generated from a standard normal distribution. To implement the bivariate spline approximation, we use polynomial splines of $d = 2$ and $r = 1$. In order to study the performance of our method when λ is larger than, equal to, and smaller than ρ respectively, we consider three cases for the parameter $\phi = (\lambda, \rho, \alpha)'$: $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$ and $\phi_{0,3} = (0.05, 0.9, 0.5)'$.

4.3.1 Simulation Study 1

In the first example, we generate 100×100 grid points as a population over the squared domain $\Omega = [0, 1]^2$. Then we randomly sample $n = 500, 1000$ grid points respectively from the population, each with 100 replications. We set $\beta(\mathbf{u}) = \sin(\|\mathbf{u}\|^2 \pi)$, its contour plot is shown in Figure 4.1 (c). We consider two triangulations on the squared domain: Δ_1 with 8 triangles and 9 vertices; Δ_2 with 18 triangles and 16 vertices, see Figure 4.1 (a)-(b).

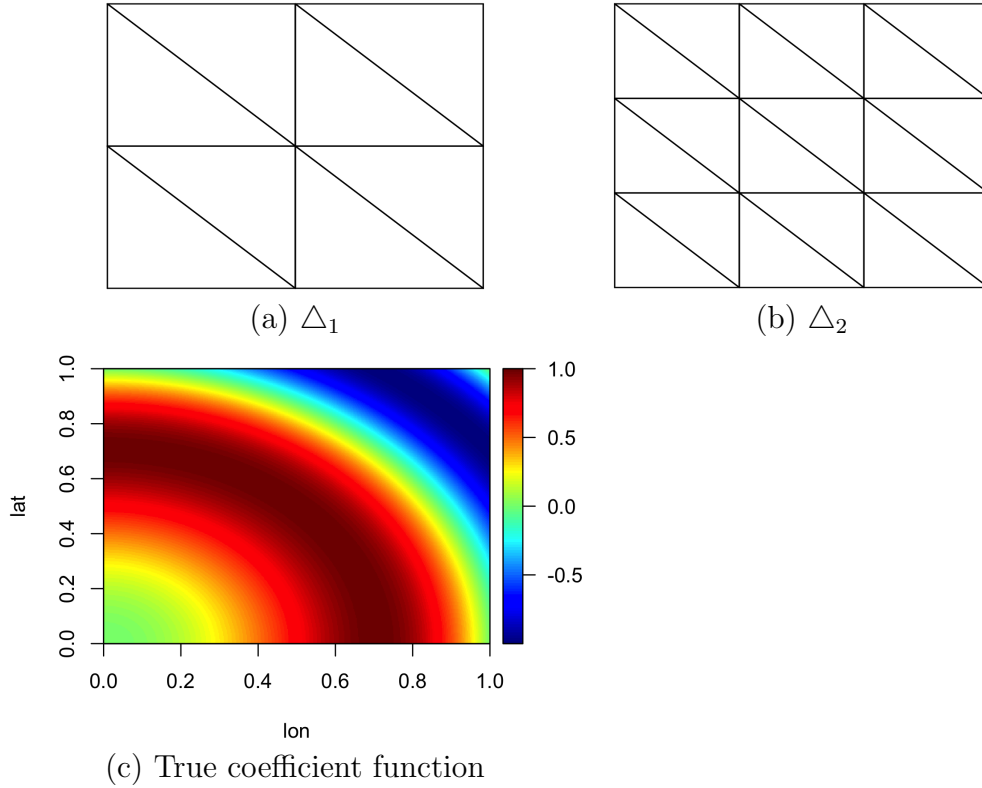


Figure 4.1: *Simulation Study 1: the true coefficient function and two triangulations.*

We calculate the mean squared error (MSE) of λ , ρ , α and the mean integrated squared error (MISE) of the coefficient function under various simulation settings. In Table 4.1, we report the average MSE and MISE over 100 Monte Carlo experiments. The results clearly show that our proposed estimation method produces much smaller MSEs and MISEs than the LSE method when λ is larger than ρ . Interestingly, when λ is equal to or smaller than ρ , the MSEs of λ and ρ are close to those of LSE, while the MSE of α and MISE of $\hat{\beta}(\cdot)$ are

still obviously smaller than those of LSE. On the other hand, we observe that the estimation accuracy of our method decreases as we increase the fineness of the triangulation regardless of the sample size and combinations of (λ, ρ) . In Figure 4.2 and 4.3, we show the contour plot of the estimated coefficient function for one simulation run based on $n = 500$ and 1000, respectively. One can easily tell that the estimated surface of the coefficient function is very similar to the true surface in Figure 4.1. Overall, it is suggested that our method performs reasonably well for a rectangular domain.

Table 4.1: *Estimation results for simulation study 1.*

n	ϕ	Method	MSE			MISE
			$\hat{\lambda}$	$\hat{\rho}$	$\hat{\alpha}$	$\hat{\beta}(\cdot)$
500	$\phi_{0,1}$	BPST(Δ_1)	0.0009	0.0005	0.0210	0.0594
		BPST(Δ_2)	0.0009	0.0006	0.0281	0.0724
		LSE	0.0224	0.0187	0.1695	0.3534
	$\phi_{0,2}$	BPST(Δ_1)	0.0274	0.0158	0.0227	0.1034
		BPST(Δ_2)	0.0278	0.0173	0.0236	0.1128
		LSE	0.0270	0.0158	0.0557	0.3532
	$\phi_{0,3}$	BPST(Δ_1)	0.0080	0.0068	0.0443	0.1066
		BPST(Δ_2)	0.0080	0.0071	0.0407	0.1121
		LSE	0.0068	0.0061	0.0533	0.3533
1000	$\phi_{0,1}$	BPST(Δ_1)	0.0006	0.0004	0.0092	0.0309
		BPST(Δ_2)	0.0009	0.0005	0.0199	0.0409
		LSE	0.0197	0.0178	0.1110	0.3503
	$\phi_{0,2}$	BPST(Δ_1)	0.0157	0.0092	0.0114	0.0584
		BPST(Δ_2)	0.0157	0.0094	0.0119	0.0631
		LSE	0.0161	0.0094	0.0363	0.3507
	$\phi_{0,3}$	BPST(Δ_1)	0.0074	0.0062	0.0220	0.0608
		BPST(Δ_2)	0.0078	0.0065	0.0226	0.0657
		LSE	0.0047	0.0039	0.0267	0.3507

Note: BPST(Δ_1): the two-stage estimation method based on triangulation Δ_1 ; BPST(Δ_2): the two-stage estimation method based on triangulation Δ_2 ; LSE: the least square estimation based method; $\phi_{0,1}$, $\phi_{0,2}$, $\phi_{0,3}$ correspond to the true value of $(\lambda, \rho, \alpha)'$: $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$.

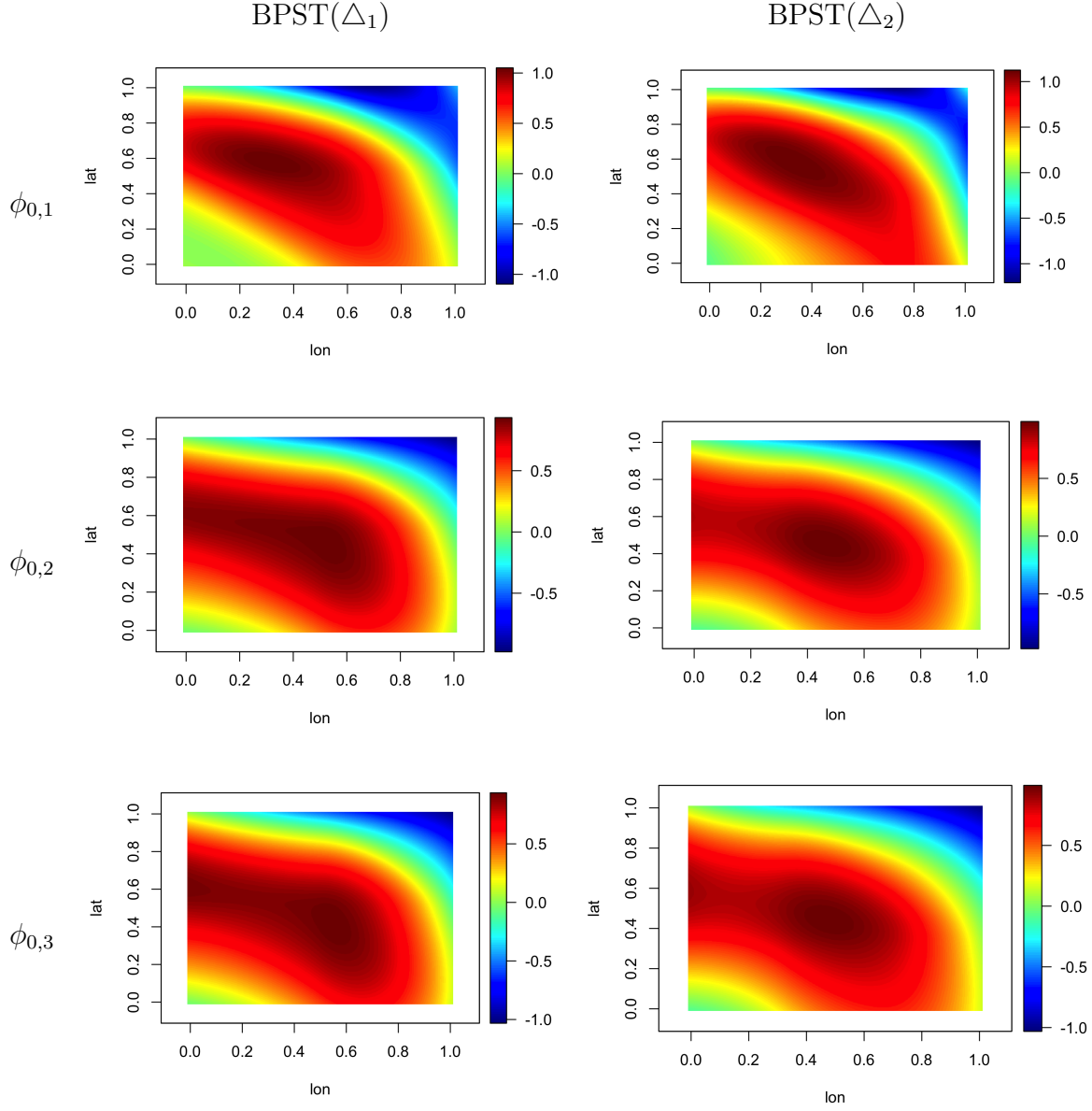


Figure 4.2: *Simulation Study 1: contour plots for the estimated coefficient functions for $n = 1000$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$.*

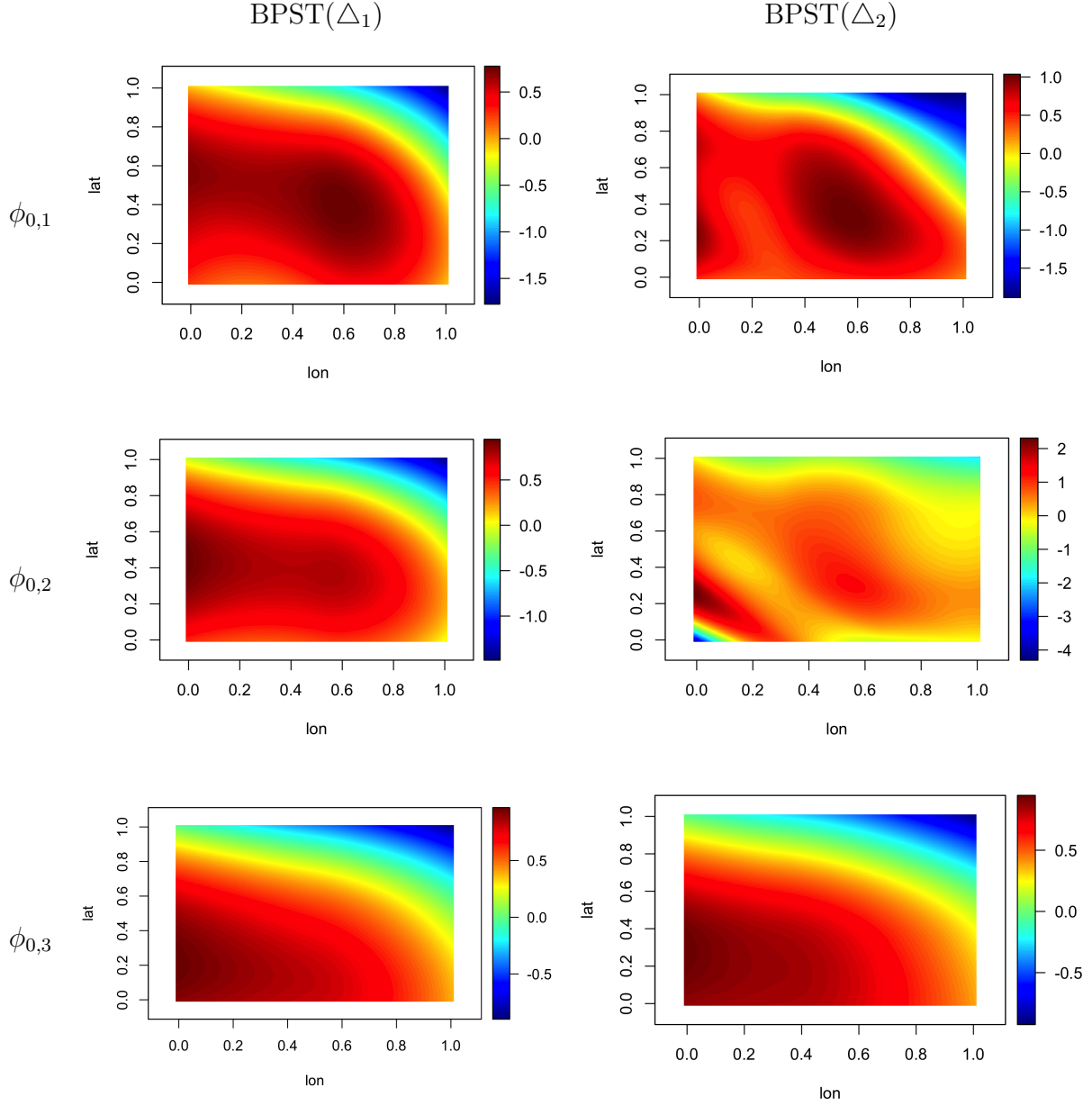


Figure 4.3: *Simulation Study 1: contour plots for the estimated coefficient functions for $n = 500$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$.*

4.3.2 Simulation Study 2

In the second example, we consider a modified horseshoe shaped domain Ω in [Wood et al. \(2008a\)](#), and the surface test function in [Wood \(2003\)](#) as $\beta(\cdot)$. Similar to the first example, we evenly generate $N = 80 \times 180 = 14400$ population grid points over the domain and then

sample n points randomly from the population where $n = 500$ and 1000 , respectively. To implement our method, we also adopt two triangulations. The first triangulation Δ_1 consists of 89 triangles and 73 vertices, and the second triangulation Δ_2 consists of 165 triangles and 120 vertices. The contour plot of the coefficient function and the two triangulations are depicted in Figure 4.4 (a)-(c).

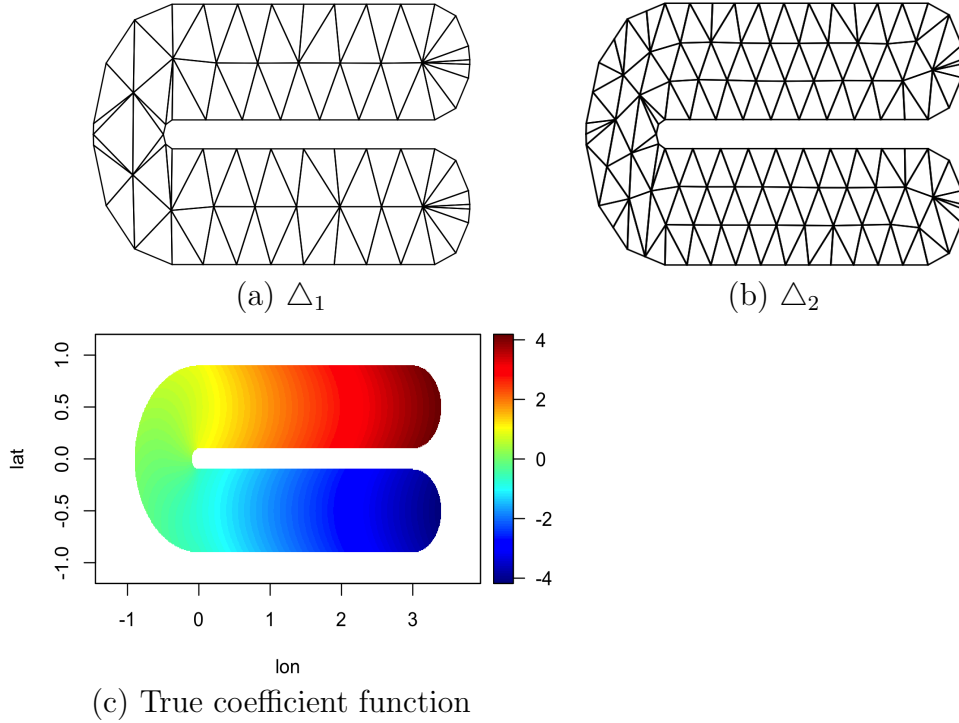


Figure 4.4: *Simulation Study 2: the true coefficient function and two triangulations.*

Table 4.2 summarizes the simulation results over 100 Monte Carlo replications. The numbers in the table represent the average MSEs and MISEs for both our method and LSE. First of all, it is quite obvious that LSE performs extremely worse than our method in estimating α and $\beta(\cdot)$. As same as the first simulation study, when λ is larger than ρ , the estimation accuracy of λ and ρ via our method is much higher than that via LSE. However, in the case that λ is equal to or smaller than ρ , the advantage of our method over LSE regarding estimating λ and ρ is not noticeable. Furthermore, the table also indicates that a finer triangulation for the horseshoe domain does not lead to higher model estimation accuracy.

Additionally, the estimated surface plots of the coefficient function for one replication at different settings are shown in Figure 4.5 and 4.6. Jointly from the plots and table, one can conclude that our method delivers distinguished performance in model estimation for complex domains.

Table 4.2: *Estimation results for simulation study 2.*

n	ϕ	Method	MSE			MISE
			$\hat{\lambda}$	$\hat{\rho}$	$\hat{\alpha}$	$\hat{\beta}(\cdot)$
500	$\phi_{0,1}$	BPST(Δ_1)	0.0007	0.0006	0.0139	0.2049
		BPST(Δ_2)	0.0010	0.0006	0.0305	0.2905
		LSE	0.3045	0.3008	11.0845	5.6612
	$\phi_{0,2}$	BPST(Δ_1)	0.0778	0.0672	0.0843	0.4590
		BPST(Δ_2)	0.1432	0.1250	0.1563	0.8114
		LSE	0.0740	0.0643	11.2512	5.6606
	$\phi_{0,3}$	BPST(Δ_1)	0.0280	0.0261	0.0435	0.2087
		BPST(Δ_2)	0.0481	0.0452	0.0603	0.2286
		LSE	0.0071	0.0062	11.6476	5.6609
1000	$\phi_{0,1}$	BPST(Δ_1)	0.0006	0.0006	0.0009	0.0415
		BPST(Δ_2)	0.0006	0.0006	0.0047	0.0664
		LSE	0.2925	0.2949	10.6725	5.6369
	$\phi_{0,2}$	BPST(Δ_1)	0.0098	0.0083	0.0106	0.0798
		BPST(Δ_2)	0.0225	0.0192	0.0231	0.1121
		LSE	0.0696	0.0640	11.8303	5.6379
	$\phi_{0,3}$	BPST(Δ_1)	0.0056	0.0052	0.0131	0.0760
		BPST(Δ_2)	0.0063	0.0057	0.0121	0.0813
		LSE	0.0050	0.0038	12.9191	5.6387

Note: BPST(Δ_1): the two-stage estimation method based on triangulation Δ_1 ; BPST(Δ_2): the two-stage estimation method based on triangulation Δ_2 ; LSE: the least square estimation based method; $\phi_{0,1}$, $\phi_{0,2}$, $\phi_{0,3}$ correspond to the true value of $(\lambda, \rho, \alpha)'$, where $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_0^2 = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$.

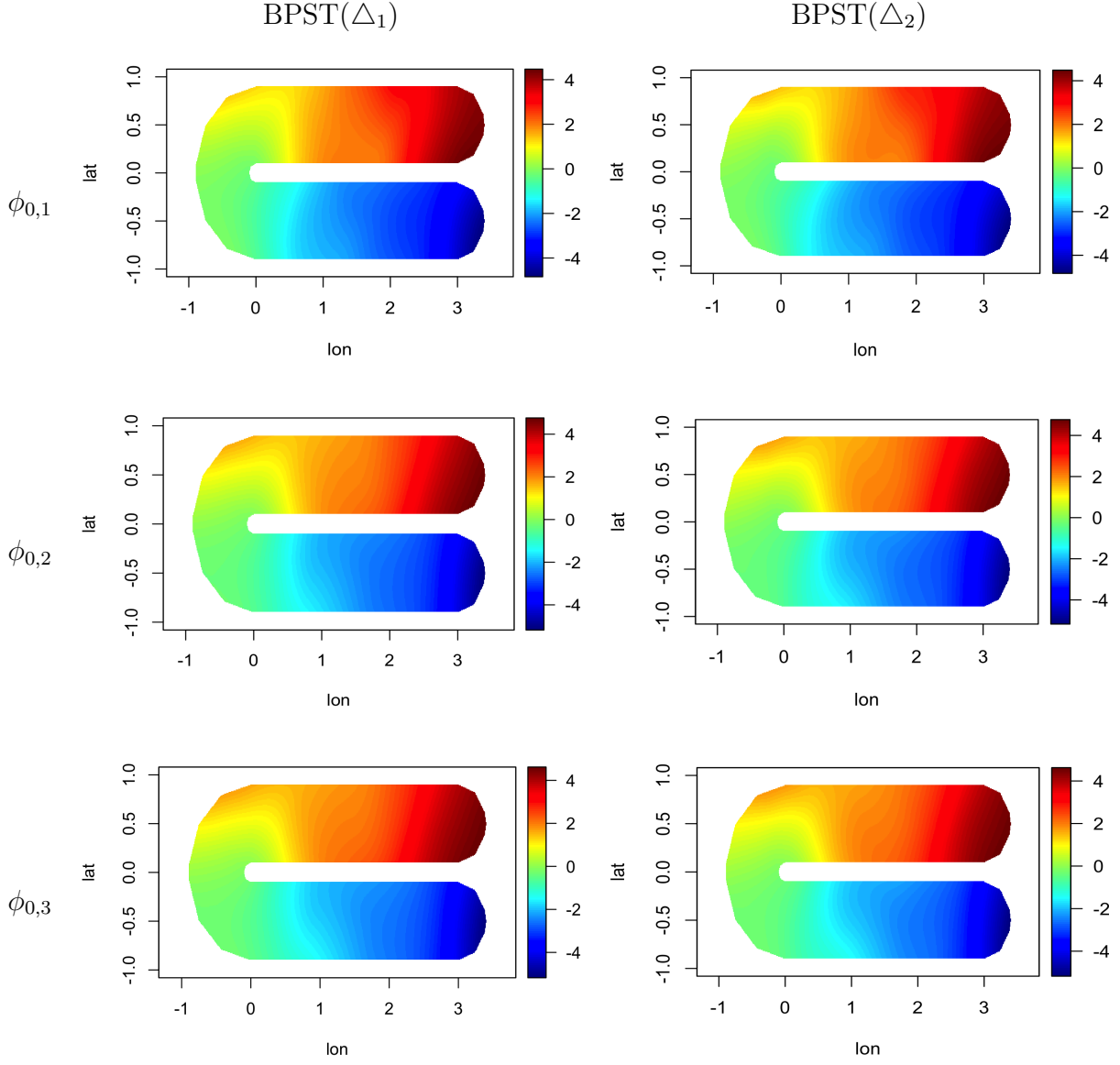


Figure 4.5: *Simulation Study 2: contour plots for the estimated coefficient functions for $n = 1000$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$.*

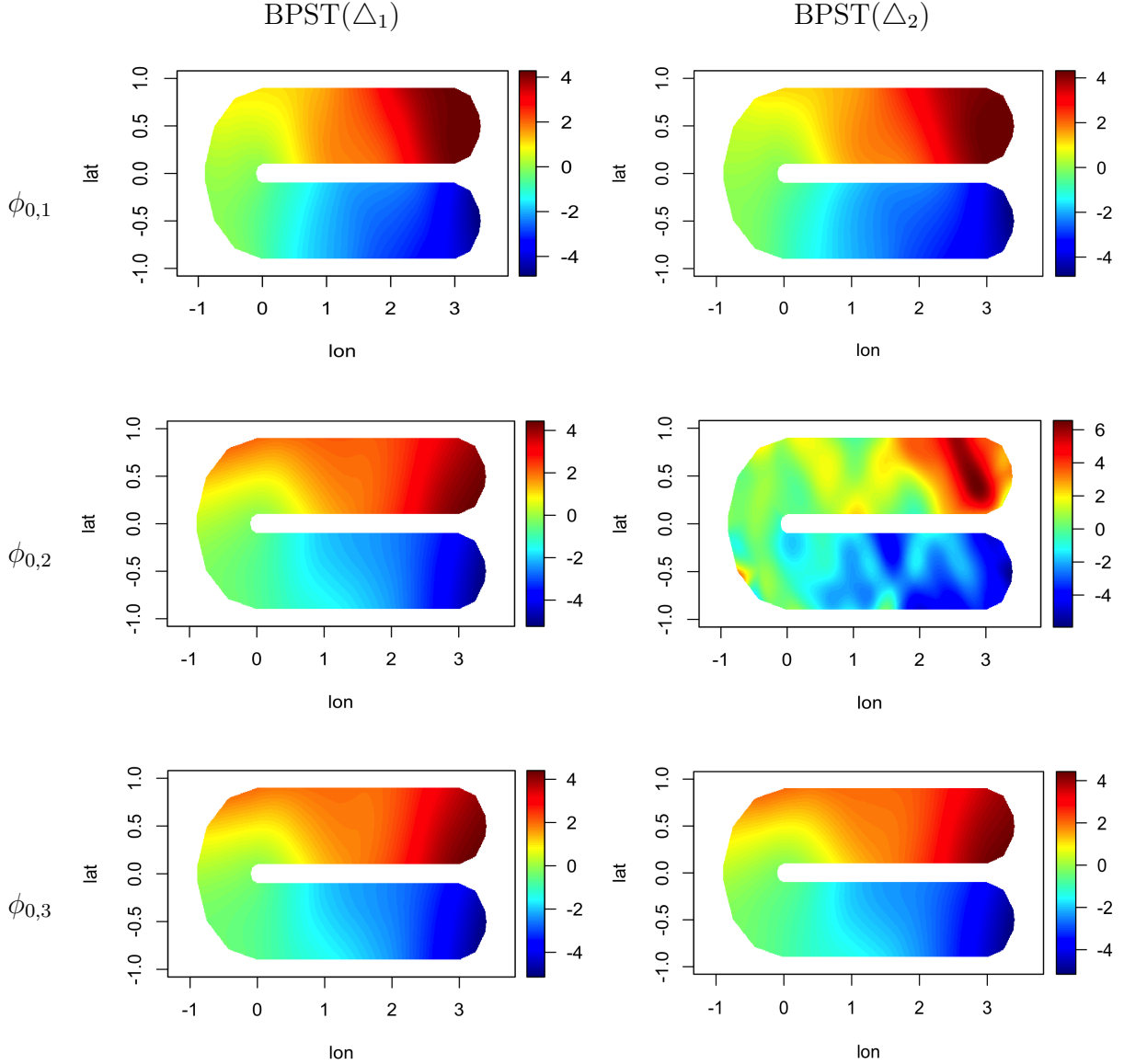


Figure 4.6: *Simulation Study 2: contour plots for the estimated coefficient functions for $n = 500$ when $\phi_{0,1} = (0.9, 0.05, 0.5)'$, $\phi_{0,2} = (0.45, 0.45, 0.5)'$, $\phi_{0,3} = (0.05, 0.9, 0.5)'$.*

4.4 Application to Florida House Price Data

In this section, we apply our proposed method to the median house price data in Florida to analyze the volatility behavior and how the land area influences the volatility over space, especially before and after the 2020 COVID-19 pandemic.

The dataset is based on the census tract level, where a census tract is a small, relatively permanent statistical subdivision of a county with about 4,000 inhabitants on average. The variable of our interest is the median house price in dollars at each census tract in Florida. We collect this data over the years of 2018 to 2020. The explanatory covariate in the dataset is the land area in square meters at each census tract in Florida. Both the median house price and land area are obtained from the official website of United States Census Bureau. After removing some locations with no records in any of the three years, this dataset includes 3191 census tracts in Florida. In this study, we focus on the log return of the house price that is transformed from the original house price given as:

$$y_{i,t} = \log p_{i,t} - \log p_{i,t-1}$$

where $p_{i,t}$ is the median house price at census tract i in the year t for $i = 1, \dots, 3191$ and $t = 2019, 2020$. As a result, the log return is the response variable in our analysis.

In studies of time series data, the volatility models are commonly applied to the residuals obtained by fitting a time-series model to the data, for example, the autoregressive model (AR). Similarly, prior to fitting the SGARCH-VCM, we fit a spatial autoregressive model (SAR) to the log return data to eliminate its level components.

Specifically, denote $y_t = (y_{1,t}, \dots, y_{3191,t})^\top$, we fit the following SAR model:

$$y_t = \zeta_t \mathbf{1} + k_t \mathbf{W} y_t + \varepsilon_t,$$

where $\mathbf{W}$ is the spatial weight matrix that represents neighboring effect and ε_t is a vector of i.i.d. random noises with zero mean and a constant variance. After fitting the SAR model, we compute the model residues μ_{it} for $i = 1, \dots, 3191$ and $t = 2019, 2020$. Taking the land area in each census tract as the explanatory variable, we proceed to fit an SGARCH-VCM to μ_{it} for each t . In other words, we consider fitting the following model for $t=2019$ and

2020, respectively:

$$\mu_{it} = \sqrt{h_{it}}\varepsilon_{it}, \quad i = 1, \dots, 3191,$$

$$\log h_{it} = \lambda_t \sum_{j=1}^n w_{ij} \log h_{jt} + \rho_t \sum_{j=1}^n w_{ij} \log \mu_{jt}^2 + \alpha_t + \text{area}_i \beta_t(\mathbf{u}_i),$$

where h_{it} is the volatility at year t that we aim to estimate and area_i is the land area value at census tract i .

For model estimation, we use a triangulation with 174 triangles and 119 vertices, as illustrated in Figure 4.8 (c). We also implement the LSE method to compare with our approach. Table 4.3 presents the estimated values of λ_t , ρ_t and α_t as well as the AICs corresponding to different methods for $t = 2019$ and 2020. Our method is validated to be better than LSE by the smaller AIC values. We also notice that the estimated λ decreased slightly in 2020 compared with 2019, which implies a decrease in the spatial dependence among the volatility over Florida in 2020. The estimated ρ is much smaller than λ , on the other hand, revealing that the spatial volatility at one location in Florida is influenced more by the volatility in its neighboring areas than the response.

Table 4.3: *Estimation results for Florida House Price Data.*

Year	Method	$\hat{\lambda}$	$\hat{\rho}$	$\hat{\alpha}$	AIC
2019	BPST	0.7500	0.0500	-0.9059	-2643.1130
	LSE	0.8250	0.0500	-0.5407	-2622.0220
2020	BPST	0.7250	0.0750	-0.7586	-2963.1170
	LSE	0.7250	0.0750	-0.7589	-2956.3920

Note: BPST: the two-stage estimation method based on BPST method; LSE: the least square estimation based method.

Figure 4.7 describes the estimated volatility of the house price all over Florida in 2019 and 2020 from the two methods. Evidently, the volatility in 2020 is generally larger than

that in 2019 except for a small area over the counties of Charlotte, Desoto, Hardee and Highlands. This is expected to be a result of the affected real estate market in Florida due to the COVID-19 pandemic in 2020. Besides, the figure manifests that the estimated volatility from our method is generally larger than that from LSE, indicating that the LSE method might give rise to an underestimation problem.

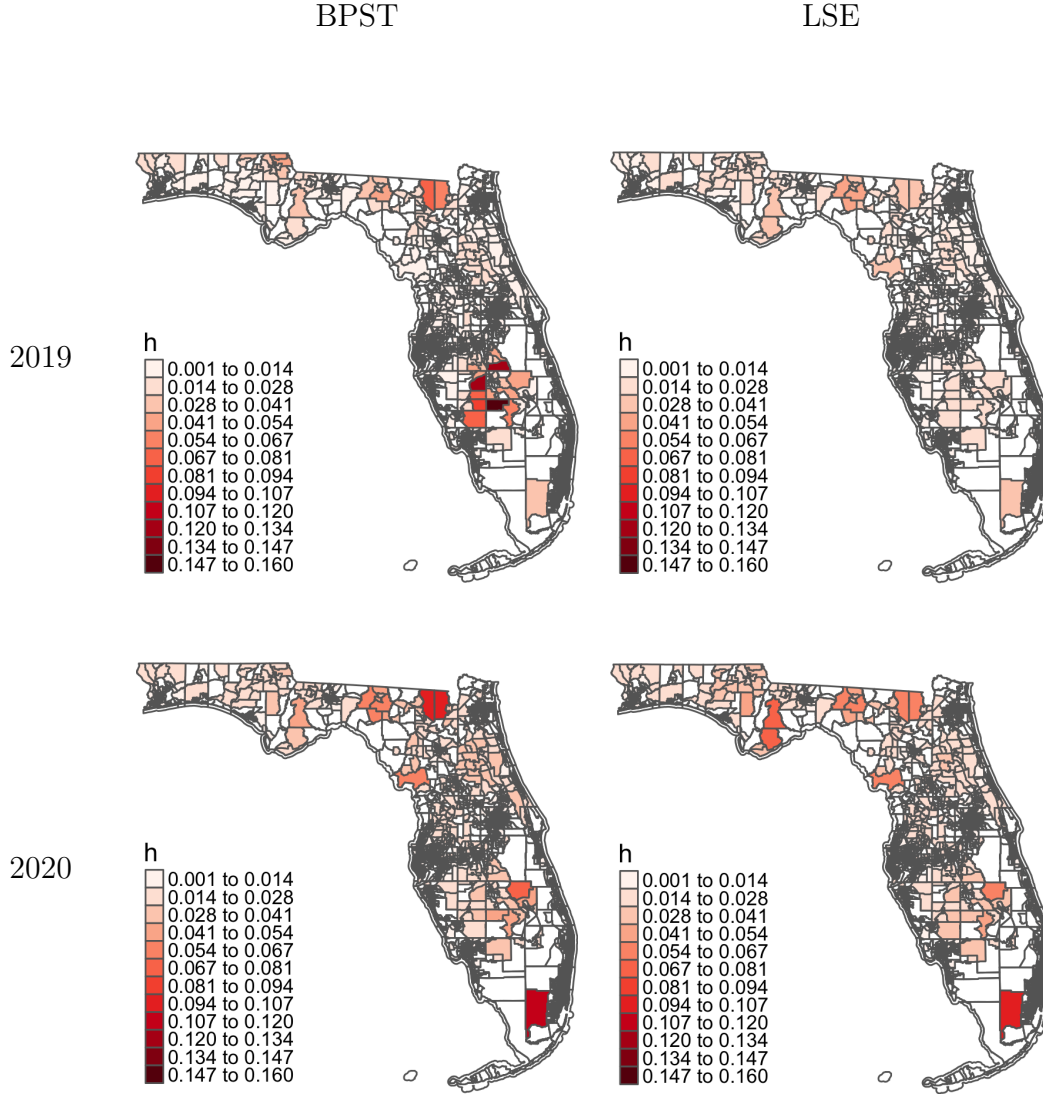


Figure 4.7: *The estimated volatility of the log return of the median house price in 2019 and 202 using two-stage estimation based on BPST, referred to as “BPST” and the least square estimation based method, referred to as “LSE”.*

In Figure 4.8 (a)-(b), we provide the contour plots of the estimated coefficient functions for land area. In 2019, the spatial volatility of the house price is positively associated with the land area in Florida except for those regions near the western and eastern border. While in 2020, there is a distinct pattern that the land area has an increasingly positive impact on this spatial volatility from west to east in Florida.

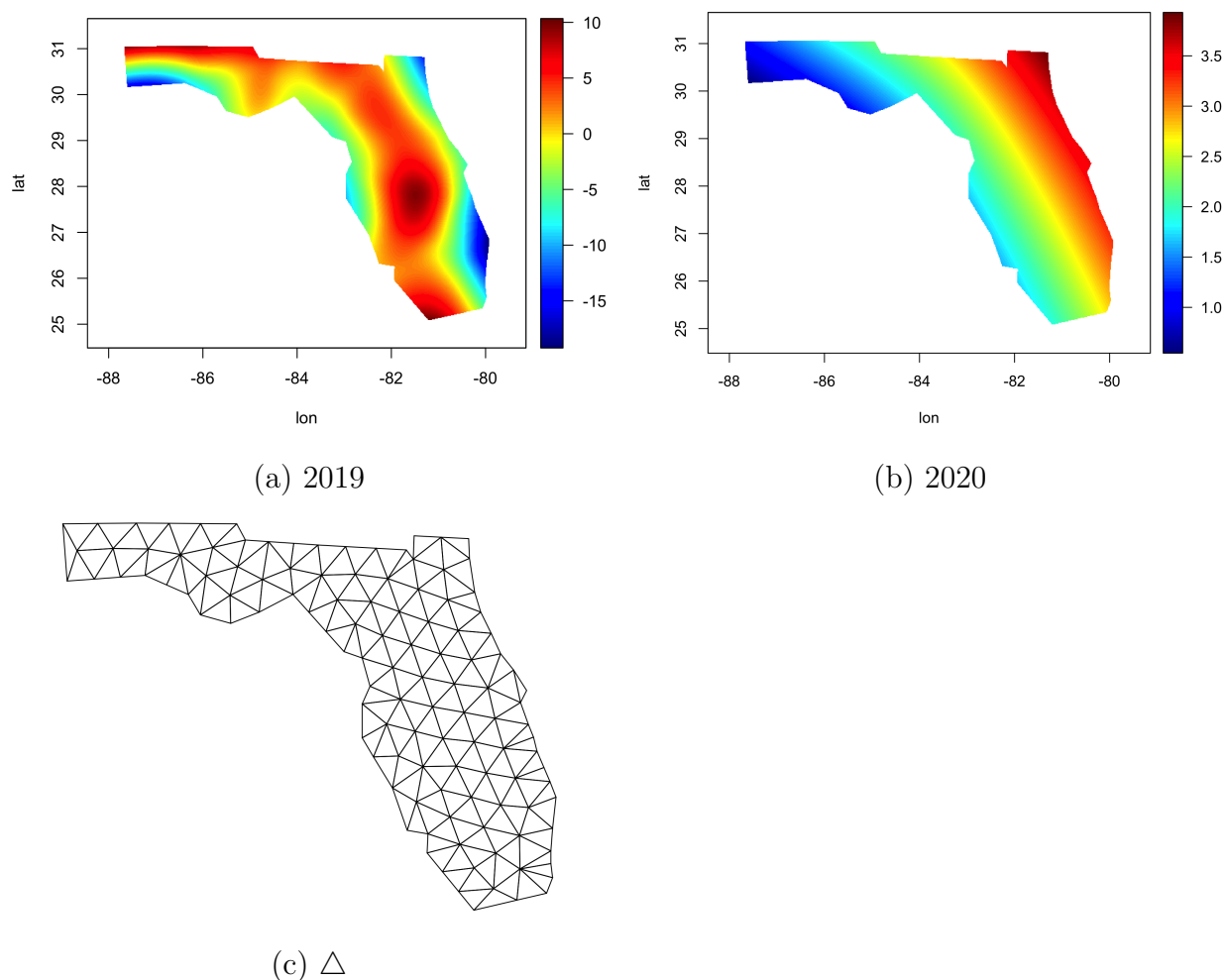


Figure 4.8: *The estimated coefficient functions of land area in 2019 and 2020 using two-stage estimation based on BPST; Δ is the triangulation over the domain of Florida.*

4.5 Concluding Remarks

In this article, we develop a new class of spatial volatility models with a nonparametric component as an extension of the SGARCH model. This model incorporates spatially varying coefficients to account for the spatially nonstationary relationship between some exogenous explanatory variables and the volatility while examining the spatial volatility behavior, such as volatility clustering and spillover effect. We have proposed a two-stage procedure based on bivariate penalized splines over triangulation method to separately estimate the intercept from the other parameters to avoid a biased estimation of the intercept. Another advantage of our estimation method is that it avoids the “leakage” problem over irregularly shaped domains. Our simulation results justified that our two-stage estimators are more accurate than existing methods in finite samples.

In the future, we will consider extending this work in a few aspects. The first extension would be to assume that some explanatory variables have a constant effect on the volatility while others have spatially varying effects. This assumption is very realistic and would add practicability to the existing model. Developing a model selection technique to separate those variables would be necessary for this extension. Another helpful extension is to incorporate the temporal effect in the analysis. It would be interesting to explore the spatio-temporal features exhibited in volatility through such a temporal extension of our model.

Chapter 5

Conclusion

In this dissertation, we have studied several semi-parametric varying coefficient models to develop more efficient statistical methodology for analyzing spatial data collected from complicated domains. The first topic is estimation and model structure selection in generalized partially linear spatially varying coefficient models (GPLSVCMs). The second topic is R package development to enhance GPLSVCMs' practicability and accessibility. The last topic is to model the volatility and its association with given covariates for spatial data.

In Chapter 2, we focus on a generalized version of SVCMS while including some constant coefficients. Our maximum likelihood estimation procedure through BPST are very computationally efficient and especially powerful in handling irregularly shaped domains. The consistency of the linear coefficient estimators and bivariate spline estimators are established. The model selection problem of identifying which coefficient is varying and which not is well addressed. Several numerical studies have confirmed the superior performance of our approach compared with existing methods.

In Chapter 3, we describe the main functionality of the R package **GPLSVCM** using a series of illustrative simulations and real-data applications. More complicated but common model selection problem is solved to effectively identify the model structure and recognize the redundant covariates at the same time. Predictive inference tools contained in this package can provide reliable information of a prediction. On the whole, **GPLSVCM** offers a

lot convenience and flexibility for real-world data analysis using GPLSVCMs.

In Chapter 4, we aim at an investigation of spatial nonstationarity of the coefficients in spatial GARCH models for analyzing the interdependent volatility exhibited in data. We use extensive simulation studies to demonstrate the high accuracy of our proposed estimators. A log return house price data is analyzed using our model to elaborate on its application in practice.

Regarding our current work in this dissertation, some extensions and future work might worth a consideration. For example, we haven't proved the asymptotic properties of either the model selection estimators in Chapter 2 and 3 or the model estimators in Chapter 4. Further studies on those theoretical properties would add more credibility to the approaches. It would also be useful to consider a spatio-temporal extension of all the models in this dissertation.

Bibliography

- Torben G Andersen, Tim Bollerslev, Francis X Diebold, and Paul Labys. The distribution of stock return volatility. *Preprint*, pages 1–48, 1999.
- Renato M Assunção. Space varying coefficient models for small area data. *Environmetrics: The official journal of the International Environmetrics Society*, 14(5):453–473, 2003.
- Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486–507, 2021.
- Paul Bergman and Sara J Berman. *The criminal law handbook: Know your rights, survive the system*. Nolo, 2022.
- Roger Bivand and Danlin Yu. *spgwr: Geographically Weighted Regression*, 2020. URL <https://CRAN.R-project.org/package=spgwr>. R package version 0.6-34.
- Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327, 1986.
- Patrick Breheny and Jian Huang. Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and Computing*, 25: 173–187, 2015.
- Chris Brunsdon, A Stewart Fotheringham, and Martin E Charlton. Geographically weighted regression: a method for exploring spatial nonstationarity. *Geographical analysis*, 28(4): 281–298, 1996.
- Jin-Chuan Duan. The garch option pricing model. *Mathematical finance*, 5(1):13–32, 1995.

- Robert F Engle. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the econometric society*, pages 987–1007, 1982.
- Jianqing Fan and Wenyang Zhang. Statistical methods with varying coefficient models. *Statistics and its Interface*, 1(1):179, 2008.
- Jianqing Fan, Tao Huang, et al. Profile likelihood inferences on semiparametric varying-coefficient partially linear models. *Bernoulli*, 11(6):1031–1057, 2005.
- Andrew O Finley and Sudipto Banerjee. Bayesian spatially varying coefficient models in the spbayes r package. *Environmental Modelling & Software*, 125:104608, 2020.
- Andrew O. Finley, Sudipto Banerjee, and Bradley P. Carlin. spBayes: An R package for univariate and multivariate hierarchical point-referenced spatial models. *Journal of Statistical Software*, 19(4):1–24, 2007. URL <https://www.jstatsoft.org/article/view/v019i04>.
- A Stewart Fotheringham, Chris Brunsdon, and Martin Charlton. *Geographically weighted regression: the analysis of spatially varying relationships*. John Wiley & Sons, 2003.
- Alan E Gelfand, Hyon-Jung Kim, CF Sirmans, and Sudipto Banerjee. Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association*, 98(462):387–396, 2003.
- Isabella Gollini, Binbin Lu, Martin Charlton, Christopher Brunsdon, and Paul Harris. GW-model: An R package for exploring spatial heterogeneity using geographically weighted models. *Journal of Statistical Software*, 63(17):1–50, 2015. doi: 10.18637/jss.v063.i17.
- Trevor Hastie and Robert Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(4):757–779, 1993.
- Xuming He and Peide Shi. Bivariate tensor-product b-splines in a partly linear model. *Journal of Multivariate Analysis*, 58(2):162–181, 1996.

- Jianhua Z Huang, Colin O Wu, and Lan Zhou. Polynomial spline estimation and inference for varying coefficient models with longitudinal data. *Statistica Sinica*, pages 763–788, 2004.
- Eric Jacquier, Nicholas G Polson, and Peter E Rossi. Bayesian analysis of stochastic volatility models. *Journal of Business & Economic Statistics*, 20(1):69–87, 2002.
- Bo Kai, Runze Li, and Hui Zou. New efficient estimation and variable selection methods for semiparametric varying-coefficient partially linear models. *Annals of statistics*, 39(1):305, 2011.
- Myungjin Kim and Li Wang. Generalized spatially varying coefficient models. *Journal of Computational and Graphical Statistics*, pages 1–10, 2020.
- M. J. Lai and L. L. Schumaker. *Spline functions on triangulations*. Cambridge University Press, 2007.
- M. J. Lai and L. Wang. Bivariate penalized splines for regression. *Statistica Sinica*, 23: 1399–1417, 2013.
- Ming-Jun Lai and Li Wang. *Triangulation: Triangulation in 2D domain*, 2020. R package version 0.1.0.
- Eun Ryung Lee, Hohsuk Noh, and Byeong U Park. Model selection via bayesian information criterion for quantile regression models. *Journal of the American Statistical Association*, 109(505):216–229, 2014.
- Xinyi Li, Li Wang, Huixia Judy Wang, and Alzheimer’s Disease Neuroimaging Initiative. Sparse learning and structure identification for ultrahigh-dimensional image-on-scalar regression. *Journal of the American Statistical Association*, pages 1–15, 2020.
- Heng Lian. Semiparametric estimation of additive quantile regression models by two-fold penalty. *Journal of Business & Economic Statistics*, 30(3):337–350, 2012.

- Heng Lian, Hua Liang, and Lan Wang. Generalized additive partial linear models for clustered data with diverging number of covariates using gee. *Statistica Sinica*, 24(1):173–196, 2014.
- Hua Liang and Runze Li. Variable selection for partially linear models with measurement errors. *Journal of the American Statistical Association*, 104(485):234–248, 2009.
- Xiang Liu, Li Wang, and Hua Liang. Estimation and variable selection for semiparametric additive partial linear models (ss-09-140). *Statistica Sinica*, 21(3):1225, 2011.
- Santosh Mishra, Liangjun Su, and Aman Ullah. Semiparametric estimator of time series conditional variance. *Journal of Business & Economic Statistics*, 28(2):256–274, 2010.
- Jingru Mu, Guannan Wang, and Li Wang. Estimation and inference in spatially varying coefficient models. *Environmetrics*, 29(1):e2485, 2018.
- Jingru Mu, Guannan Wang, and Li Wang. Spatial autoregressive partially linear varying coefficient models. *Journal of Nonparametric Statistics*, 32(2):428–451, 2020.
- John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384, 1972.
- Byeong U Park, Enno Mammen, Young K Lee, and Eun Ryung Lee. Varying coefficient regression models: a review and new developments. *International Statistical Review*, 83(1):36–64, 2015.
- T. Ramsay. Spline smoothing over difficult regions. *Journal of the Royal Statistical Society, Series B*, 64:307–319, 2002.
- Peter M Robinson. Large-sample inference on spatial dependence. *The Econometrics Journal*, 12:S68–S82, 2009.
- L. Sangalli, J. Ramsay, and T. Ramsay. Spatial spline regression models. *Journal of the Royal Statistical Society, Series B*, 75:681–703, 2013.

- Takaki Sato and Yasumasa Matsuda. Spatial autoregressive conditional heteroskedasticity models. *Journal of the Japan Statistical Society*, 47(2):221–236, 2017.
- Takaki Sato and Yasumasa Matsuda. Spatial extension of generalized autoregressive conditional heteroskedasticity models. *Spatial Economic Analysis*, 16(2):148–160, 2021.
- G William Schwert. Why does stock market volatility change over time? *The journal of finance*, 44(5):1115–1153, 1989.
- Charles J Stone. Additive regression and other nonparametric models. *The annals of Statistics*, pages 689–705, 1985.
- Yan Sun, Hongjia Yan, Wenyang Zhang, Zudi Lu, et al. A semiparametric spatial dynamic model. *Annals of Statistics*, 42(2):700–727, 2014.
- Yanlin Tang, Huixia Judy Wang, Zhongyi Zhu, and Xinyuan Song. A unified variable selection approach for varying coefficient models. *Statistica Sinica*, pages 601–628, 2012.
- Dewei Wang and KB Kulasekera. Parametric component detection and variable selection in varying-coefficient partially linear models. *Journal of Multivariate Analysis*, 112:117–129, 2012.
- Hansheng Wang, Runze Li, and Chih-Ling Tsai. Tuning parameter selectors for the smoothly clipped absolute deviation method. *Biometrika*, 94(3):553–568, 2007.
- Kangning Wang and Lu Lin. Robust structure identification and variable selection in partial linear varying coefficient models. *Journal of Statistical Planning and Inference*, 174:153–168, 2016.
- Li Wang, Guannan Wang, Min-Jun Lai, and Lei Gao. Efficient estimation of partially linear models for spatial data over complex domains. *arXiv preprint arXiv:1605.08737*, 2016.
- Lifeng Wang, Hongzhe Li, and Jianhua Z Huang. Variable selection in nonparametric varying-coefficient models for analysis of repeated measurements. *Journal of the American Statistical Association*, 103(484):1556–1569, 2008.

- Lily Wang, Ming-Jun Lai, GuanNan Wang, Myungjin Kim, , Xinyi Li, , Jingru Mu, , Jue Wang, , Yueying Wang, , and Shan Yu and. *BPST: Bivariate Spline over Triangulation*, 2020. R package version 0.1.0.
- Fengrong Wei, Jian Huang, and Hongzhe Li. Variable selection and estimation in high-dimensional varying-coefficient models. *Statistica Sinica*, 21(4):1515, 2011.
- David Wheeler. *gwrr: Fits geographically weighted regression models with diagnostic tools*, 2013. URL <https://CRAN.R-project.org/package=gwrr>. R package version 0.2-1.
- S. N. Wood. Thin plate regression splines. *Journal of the Royal Statistical Society, Series B*, 65:95–114, 2003.
- S. N. Wood, M. V. Bravington, and S. L. Hedley. Soap film smoothing. *Journal of the Royal Statistical Society, Series B*, 70:931–955, 2008a.
- Simon N Wood, Mark V Bravington, and Sharon L Hedley. Soap film smoothing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):931–955, 2008b.
- Huiliang Xie and Jian Huang. Scad-penalized regression in high-dimensional partially linear models. *The Annals of Statistics*, 37(2):673–696, 2009.
- Jun Yan. Spatial stochastic volatility for lattice data. *Journal of agricultural, biological, and environmental statistics*, 12(1):25–40, 2007.
- Lijian Yang, Byeong U Park, Lan Xue, and Wolfgang Härdle. Estimation and testing for varying coefficients in additive models with marginal integration. *Journal of the American Statistical Association*, 101(475):1212–1227, 2006.
- Shan Yu, Guannan Wang, Li Wang, Chenhui Liu, and Lijian Yang. Estimation and inference for generalized geoadditive models. *Journal of the American Statistical Association*, 115(530):761–774, 2020.

- Yiyun Zhang, Runze Li, and Chih-Ling Tsai. Regularization parameter selections via generalized information criterion. *Journal of the American Statistical Association*, 105(489): 312–323, 2010.
- L. Zhou and H. Pan. Smoothing noisy data for irregular regions using penalized bivariate splines on triangulations. *Computational Statistics*, 29:263–281, 2014.

Appendix A

Theoretical Results with Details for Chapter 2

This section provides the detailed proof for the asymptotic properties of the proposed estimators established in section 2.2.4.

A.1 Properties of Penalized Quasi-likelihood Estimators

In the following, we use c, C, c_1, c_2, C_1, C_2 , etc. as generic constants, which may be different even in the same line. For any two sequences $\{a_n\}, \{b_n\}$, $n = 1, 2, \dots$, we use $a_n \asymp b_n$ if there are constants $0 < c_1 < c_2 < \infty$ such that $c_1 < a_n/b_n < c_2$ for all n sufficiently large.

Denote the index of Bernstein basis polynomials by $\mathcal{J} = \left\{1, 2, \dots, \frac{(d+1)(d+2)N}{2}\right\}$ where N is the number of triangles in a triangulation Δ and d is the degree of Bernstein basis polynomials. Though we could use different triangulations for $\hat{\alpha}_k(\mathbf{u}), k = 1, \dots, p_2$, for convenience, we use a common triangulation for $\hat{\alpha}_k(\mathbf{u}), k = 1, \dots, p_2$, that is, we assume $\Delta_k = \Delta$, $\mathbf{B}_k = \mathbf{B} = \{\mathbf{B}(\mathbf{U}_1), \dots, \mathbf{B}(\mathbf{U}_n)\}^\top$ and $\mathbf{H}_k = \mathbf{H}$ where $\mathbf{B}_k(\mathbf{U}_i) = \mathbf{B}(\mathbf{U}_i) = \{B_j(\mathbf{U}_i), j \in \mathcal{J}\}^\top$ for $i = 1, \dots, n$ and $k = 1, \dots, p_2$. For a bivariate Bernstein basis polynomials space $\mathbb{S}_d^r(\Delta)$,

define the corresponding approximate model space as

$$\mathcal{M}_s = \left\{ \eta : \eta(\mathbf{t}) = \sum_{l=1}^{p_1} x_l \beta_l + \sum_{k=1}^{p_2} z_k \psi_k(\mathbf{u}), \beta_l \in \mathbb{R}, \mathbf{u} \in \Omega, \psi_k \in \mathbb{S}_d^r(\Delta) \right\}.$$

LEMMA 1. [Lemma S.1. in Kim and Wang (2020)] Let $\boldsymbol{\psi}(\mathbf{u}) = \{\psi_1(\mathbf{u}), \dots, \psi_{p_2}(\mathbf{u})\}^\top$, where $\psi_k(\mathbf{u}) = \sum_{j \in \mathcal{J}} \gamma_{kj} B_j(\mathbf{u})$. Then, under Assumptions (A2) and (A6), $\|\boldsymbol{\psi}\|^2 \asymp \sum_{k=1}^{p_2} \|\psi_k\|_{L_2}^2$.

LEMMA 2. [Lai and Schumaker (2007)] Under Assumptions (A7), there exist positive constants c_j, C_j depending on the smoothness r and the shape parameter δ such that

$$c_j |\Delta|^2 \sum_{j \in \mathcal{J}} \gamma_j^2 \leq \left\| \sum_{j \in \mathcal{J}} \gamma_j B_j \right\|_{L_2}^2 \leq C_j |\Delta|^2 \sum_{j \in \mathcal{J}} \gamma_j^2.$$

LEMMA 3. Under Assumptions (A2), (A3) and (A7), there exist positive constants c, C such that

$$c \left(\sum_{l=1}^{p_1} \beta_l^2 + |\Delta|^2 \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} \gamma_{kj}^2 \right) \leq \left\| \sum_{l=1}^{p_1} X_l \beta_l + \sum_{k=1}^{p_2} Z_k \psi_k \right\|^2 \leq C \left(\sum_{l=1}^{p_1} \beta_l^2 + |\Delta|^2 \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} \gamma_{kj}^2 \right).$$

Proof. By Assumption (A2), (A3) and Lemma 2, we have

$$\begin{aligned} \left\| \sum_{l=1}^{p_1} X_l \beta_l + \sum_{k=1}^{p_2} Z_k \psi_k \right\|^2 &\leq 2 \left\| \sum_{l=1}^{p_1} X_l \beta_l \right\|^2 + 2 \left\| \sum_{k=1}^{p_2} Z_k \psi_k \right\|^2 \\ &\leq 2 \left\| \sum_{l=1}^{p_1} X_l \beta_l \right\|^2 + 2C_f \left\| \sum_{k=1}^{p_2} Z_k \psi_k \right\|_{L_2}^2 \\ &= 2 \left\| \sum_{l=1}^{p_1} X_l \beta_l \right\|^2 + 2C_f \|\boldsymbol{\psi}\|^2 \\ &\leq C \left(\sum_{l=1}^{p_1} \beta_l^2 + |\Delta|^2 \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} \gamma_{kj}^2 \right). \end{aligned} \tag{A.1}$$

Next, denote $\mu_{X_l} = \mathbb{E}X_l$ and $\mu_{\psi_k} = \mathbb{E}\psi_k(\mathbf{U})$, then we have

$$\left\| \sum_{l=1}^{p_1} X_l \beta_l + \sum_{k=1}^{p_2} Z_k \psi_k \right\|^2 = \left\| \sum_{l=1}^{p_1} \beta_l (X_l - \mu_{X_l}) + \sum_{k=1}^{p_2} Z_k (\psi_k - \mu_{\psi_k}) \right\|^2 + \left(\sum_{l=1}^{p_1} \beta_l \mu_{X_l} + \sum_{k=1}^{p_2} Z_k \mu_{\psi_k} \right)^2.$$

According to Lemma 1 by [Stone \(1985\)](#), we have

$$\left\| \sum_{l=1}^{p_1} \beta_l (X_l - \mu_{X_l}) + \sum_{k=1}^{p_2} Z_k (\psi_k - \mu_{\psi_k}) \right\|^2 \geq c_0 \left\{ \sum_{l=1}^{p_1} \|\beta_l (X_l - \mu_{X_l})\|^2 + \left\| \sum_{k=1}^{p_2} Z_k (\psi_k - \mu_{\psi_k}) \right\|^2 \right\}.$$

Therefore, by Assumption (A3) and Lemma 2,

$$\begin{aligned} \left\| \sum_{l=1}^{p_1} X_l \beta_l + \sum_{k=1}^{p_2} Z_k \psi_k \right\|^2 &\geq c_1 \left\{ \left\| \sum_{l=1}^{p_1} X_l \beta_l \right\|^2 + \left\| \sum_{k=1}^{p_2} Z_k \psi_k \right\|^2 \right\} \\ &\geq c \left(\sum_{l=1}^{p_1} \beta_l^2 + |\Delta|^2 \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} \gamma_{kj}^2 \right). \end{aligned} \quad (\text{A.2})$$

The desired result follows [\(A.1\)](#) and [\(A.2\)](#). $\square$

LEMMA 4. *[Theorem 10.2 in Lai and Schumaker (2007)] Assume $\psi(\cdot)$ is in Sobolev space $W^{d+1,\infty}(\Omega)$ and $|\Delta|$ is a π quasi-uniform triangulation of a polygonal domain Ω . For bi-integer (a_1, a_2) with $0 \leq a_1 + a_2 \leq d$, there exists a unique spline fit $\psi^*(\cdot) \in \mathbb{S}_d^r(\Delta)$ ($d \geq 3r + 2$) such that*

$$\|D_{u_1}^{a_1} D_{u_2}^{a_2} (\psi - \psi^*)\|_\infty \leq C |\Delta|^{d+1-a_1-a_2} |\psi|_{d+1,\infty},$$

where C is an absolute constant depending on d, r and shape parameter δ .

For $k, k' = 1, \dots, p_1$ and $j, j' \in \mathcal{J}$, let $\langle B_{kj}, B_{k'j'} \rangle_n = \frac{1}{n} \sum_{i=1}^n Z_{ik} Z_{ik'} B_{kj}(\mathbf{U}_i) B_{k'j'}(\mathbf{U}_i)$ and $\langle B_{kj}, B_{k'j'} \rangle = \mathbb{E}[Z_{ik} Z_{ik'} B_{kj}(\mathbf{U}_i) B_{k'j'}(\mathbf{U}_i)]$.

LEMMA 5. Under Assumptions (A2)-(A3) and (A6)-(A8),

$$\max_{\substack{1 \leq k, k' \leq p_1 \\ j, j' \in \mathcal{J}}} |\langle B_{kj}, B_{k'j'} \rangle_n - \langle B_{kj}, B_{k'j'} \rangle| = O_{\text{a.s.}} \left\{ |\Delta| n^{-1/2} \log^{1/2} n \right\}, \quad (\text{A.3})$$

$$\max_{1 \leq l, l' \leq p_1} \left| \frac{1}{n} \sum_{i=1}^n X_{il} X_{il'} - E \{X_{il} X_{il'}\} \right| = O_{\text{a.s.}} \left(n^{-1/2} \log^{1/2} n \right), \quad (\text{A.4})$$

$$\max_{\substack{1 \leq l \leq p_1 \\ 1 \leq k \leq p_2 \\ j \in \mathcal{J}}} \left| \frac{1}{n} \sum_{i=1}^n Z_{ik} B_{kj}(\mathbf{U}_i) X_{il} - E \{Z_{ik} B_{kj}(\mathbf{U}_i) X_{il}\} \right| = O_{\text{a.s.}} \left\{ |\Delta| n^{-1/2} \log^{1/2} n \right\}. \quad (\text{A.5})$$

Proof. By Assumption (A2) and (A6), (A.3) holds according to Lemma S.2 in Kim and Wang (2020), next we prove (A.4) and (A.5). Let $Q_{i,l,l'} = X_{il} X_{il'} - E X_{il} X_{il'}$. Note that, by Assumption (A3), we have

$$|E X_{il} X_{il'}|^q \asymp C_1, E |X_{il} X_{il'}|^q \asymp C_2. \quad (\text{A.6})$$

Then, by (A.6), the second moment and q th moment satisfy the following:

$$E Q_{i,l,l'}^2 = E \{X_{il}^2 X_{il'}^2\} - \{E X_{il} X_{il'}\}^2 \asymp C,$$

and

$$\begin{aligned} E |Q_{i,l,l'}|^q &\leq 2^{q-1} \{E |X_{il} X_{il'}|^q + |E X_{il} X_{il'}|^q\} \\ &\leq C q! 2^{q-2} E Q_{i,l,l'}^2 \end{aligned}$$

for some constant C which satisfies the Cramer's condition with the Cramer's constant $c^* = 2C^{(q-2)^{-1}}$. By Bernstein's inequality in Bosq (1998), for $\delta > 0$ large enough,

$$\begin{aligned} P \left(\frac{1}{n} \left| \sum_{i=1}^n Q_{i,l,l'} \right| \geq \delta n^{-1/2} \log^{1/2} n \right) &\leq 2 \exp \left(\frac{-\delta^2 \log n}{4 + 2c^* \delta n^{-1/2} \log^{1/2} n} \right) \\ &\leq 2n^{-4}. \end{aligned} \quad (\text{A.7})$$

Thus for such δ in (A.7),

$$\begin{aligned}
& \sum_{n=1}^{\infty} P \left(\max_{1 \leq l, l' \leq p_1} \frac{1}{n} \left| \sum_{i=1}^n Q_{i,l,l'} \right| \geq \delta n^{-1/2} \log^{1/2} n \right) \\
& \leq \sum_{n=1}^{\infty} \sum_{1 \leq l, l' \leq p_1} P \left(\frac{1}{n} \left| \sum_{i=1}^n Q_{i,l,l'} \right| \geq \delta n^{-1/2} \log^{1/2} n \right) \\
& \leq \sum_{n=1}^{\infty} p_1^2 \cdot 2n^{-4} \leq C \sum_{n=1}^{\infty} \frac{1}{n^4} < \infty,
\end{aligned}$$

where p_1 is the number of linear covariates. It follows from Borel-Cantelli Lemma that

$$\max_{1 \leq l, l' \leq p_1} \frac{1}{n} \left| \sum_{i=1}^n Q_{i,l,l'} \right| = O_{\text{a.s.}} \{ (\log n)^{1/2} n^{-1/2} \}.$$

As a result, (A.4) is obtained. Similarly, (A.5) could be obtained. $\square$

LEMMA 6. *Let $W_1 = \sum_{l=1}^{p_1} X_l \beta_{l,1} + \sum_{k=1}^{p_2} Z_k \psi_{k,1}$, $W_2 = \sum_{l=1}^{p_1} X_l \beta_{l,2} + \sum_{k=1}^{p_2} Z_k \psi_{k,2}$. Suppose Assumptions (A2)-(A3) and (A6)-(A8) hold, then we have*

$$R_n = \sup_{W_1, W_2 \in \mathcal{M}_s} \left| \frac{\langle W_1, W_2 \rangle_n - \langle W_1, W_2 \rangle}{\|W_1\| \|W_2\|} \right| = O_{\text{a.s.}} \left(n^{-1/2} \log^{1/2} n \right).$$

Proof. By Lemma 3, we have

$$\|W_1\| \|W_2\| \asymp \left(\sum_{l=1}^{p_1} \beta_{l,1}^2 + |\Delta|^2 \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} \gamma_{kj,1}^2 \right)^{1/2} \left(\sum_{l=1}^{p_1} \beta_{l,2}^2 + |\Delta|^2 \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} \gamma_{kj,2}^2 \right)^{1/2}.$$

Also, note that

$$\begin{aligned}
& \langle W_1, W_2 \rangle_n - \langle W_1, W_2 \rangle = \\
& \sum_{l,l'=1}^{p_1} \beta_{l,1} \beta_{l',2} \left[\frac{1}{n} \sum_{i=1}^n X_{il} X_{il'} - \mathbb{E} \{X_{il} X_{il'}\} \right] + \\
& \sum_{k,k'=1}^{p_2} \sum_{j,j' \in \mathcal{J}} \gamma_{kj,1} \gamma_{k'j',2} [\langle B_{kj}, B_{k'j'} \rangle_n - \langle B_{kj}, B_{k'j'} \rangle] + \\
& \sum_{l=1}^{p_1} \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} (\beta_{l,1} \gamma_{kj,2} + \beta_{l,2} \gamma_{kj,1}) \left[\frac{1}{n} \sum_{i=1}^n Z_{ik} B_{kj}(\mathbf{U}_i) X_{il} - \mathbb{E} \{Z_{ik} B_{kj}(\mathbf{U}_i) X_{il}\} \right].
\end{aligned}$$

Thus, we have

$$\begin{aligned}
& \langle W_1, W_2 \rangle_n - \langle W_1, W_2 \rangle \leq \\
& \sum_{l,l'=1}^{p_1} |\beta_{l,1} \beta_{l',2}| \max_{1 \leq l, l' \leq p_1} \left| \frac{1}{n} \sum_{i=1}^n X_{il} X_{il'} - \mathbb{E} \{X_{il} X_{il'}\} \right| + \\
& \sum_{k,k'=1}^{p_2} \sum_{j,j' \in \mathcal{J}} |\gamma_{kj,1} \gamma_{k'j',2}| \max_{\substack{1 \leq k, k' \leq p_2 \\ j, j' \in \mathcal{J}}} |\langle B_{kj}, B_{k'j'} \rangle_n - \langle B_{kj}, B_{k'j'} \rangle| + \\
& \sum_{l=1}^{p_1} \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} (|\beta_{l,1} \gamma_{kj,2}| + |\beta_{l,2} \gamma_{kj,1}|) \max_{\substack{1 \leq l \leq p_1 \\ 1 \leq k \leq p_2 \\ j \in \mathcal{J}}} \left| \frac{1}{n} \sum_{i=1}^n Z_{ik} B_{kj}(\mathbf{U}_i) X_{il} - \mathbb{E} \{Z_{ik} B_{kj}(\mathbf{U}_i) X_{il}\} \right|.
\end{aligned}$$

Denote

$$\begin{aligned}
I_1 &= \sum_{l,l'=1}^{p_1} |\beta_{l,1} \beta_{l',2}| \max_{1 \leq l, l' \leq p_1} \left| \frac{1}{n} \sum_{i=1}^n X_{il} X_{il'} - \mathbb{E} \{X_{il} X_{il'}\} \right|, \\
I_2 &= \sum_{k,k'=1}^{p_2} \sum_{j,j' \in \mathcal{J}} |\gamma_{kj,1} \gamma_{k'j',2}| \max_{\substack{1 \leq k, k' \leq p_2 \\ j, j' \in \mathcal{J}}} |\langle B_{kj}, B_{k'j'} \rangle_n - \langle B_{kj}, B_{k'j'} \rangle|, \\
I_3 &= \sum_{l=1}^{p_1} \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} (|\beta_{l,1} \gamma_{kj,2}| + |\beta_{l,2} \gamma_{kj,1}|) \max_{\substack{1 \leq l \leq p_1 \\ 1 \leq k \leq p_2 \\ j \in \mathcal{J}}} \left| \frac{1}{n} \sum_{i=1}^n Z_{ik} B_{kj}(\mathbf{U}_i) X_{il} - \mathbb{E} \{Z_{ik} B_{kj}(\mathbf{U}_i) X_{il}\} \right|.
\end{aligned}$$

By Lemma 5, we have

$$I_1 = \left(\sum_{l=1}^{p_1} \beta_{l,1}^2 \right)^{1/2} \left(\sum_{l=1}^{p_1} \beta_{l,2}^2 \right)^{1/2} \times O_{\text{a.s.}} \left(n^{-1/2} \log^{1/2} n \right),$$

$$I_2 = \sum_{k,k'=1}^{p_2} \left(\sum_{j \in \mathcal{J}} \gamma_{kj,1}^2 \right)^{1/2} \left(\sum_{j \in \mathcal{J}} \gamma_{kj,2}^2 \right)^{1/2} \times O_{\text{a.s.}} \left(|\Delta| n^{-1/2} \log^{1/2} n \right).$$

It follows from Cauchy Schwarz inequality that

$$\begin{aligned} & \sum_{l=1}^{p_1} \sum_{k=1}^{p_2} \sum_{j \in \mathcal{J}} (|\beta_{l,1} \gamma_{kj,2}| + |\beta_{l,2} \gamma_{kj,1}|) \\ &= \sum_{k=1}^{p_2} \left[\left(\sum_{l=1}^{p_1} |\beta_{l,1}| \right) \left(\sum_{j \in \mathcal{J}} |\gamma_{kj,2}| \right) + \left(\sum_{l=1}^{p_1} |\beta_{l,2}| \right) \left(\sum_{j \in \mathcal{J}} |\gamma_{kj,1}| \right) \right] \\ &\leq C |\Delta|^{-1} \left\{ \left(\sum_{l=1}^{p_1} \beta_{l,1}^2 \right)^{1/2} \left(\sum_{j \in \mathcal{J}} \gamma_{j,2}^2 \right)^{1/2} + \left(\sum_{l=1}^{p_1} \beta_{l,2}^2 \right)^{1/2} \left(\sum_{j \in \mathcal{J}} \gamma_{j,1}^2 \right)^{1/2} \right\}, \end{aligned}$$

which implies that

$$I_3 = \left(\sum_{l=1}^{p_1} \beta_{l,1}^2 \right)^{1/2} \left(\sum_{j \in \mathcal{J}} \gamma_{j,2}^2 \right)^{1/2} \times O_{\text{a.s.}} \left(n^{-1/2} \log^{1/2} n \right) +$$

$$\left(\sum_{l=1}^{p_1} \beta_{l,2}^2 \right)^{1/2} \left(\sum_{j \in \mathcal{J}} \gamma_{j,1}^2 \right)^{1/2} \times O_{\text{a.s.}} \left(n^{-1/2} \log^{1/2} n \right).$$

Combining I_1 , I_2 and I_3 , we obtain the desired result. $\square$

As a direct result of Lemma 6, we obtain that

$$\sup_{W \in \mathcal{M}_s} \left| \frac{\|W\|_n^2}{\|W\|^2} - 1 \right| = O_{\text{a.s.}} \left(n^{-1/2} \log^{1/2} n \right).$$

Let $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_1, \dots, \lambda_{p_2})$, $\mathbf{D}_{\mathbf{\Lambda}} = \mathbf{\Lambda} \otimes \mathbf{Q}_2^{\top} \mathbf{P} \mathbf{Q}_2$ and $\mathbf{B}^*(\mathbf{U}_i) = \mathbf{Q}_2^{\top} \mathbf{B}(\mathbf{U}_i)$ where $\mathbf{B}(\mathbf{U}_i) = \{B_j(\mathbf{U}_i), j \in \mathcal{J}\}^{\top}$. Denote

$$\Phi_{n,\lambda} = \frac{1}{n} \sum_{i=1}^n \begin{pmatrix} \mathbf{X}_i \mathbf{X}_i^{\top} & \mathbf{X}_i \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^{\top} \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \mathbf{X}_i^{\top} & \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^{\top} + \frac{1}{n} \mathbf{D}_{\mathbf{\Lambda}} \end{pmatrix}.$$

LEMMA 7. Under Assumptions (A2), (A6), (A7) and (A8), there exist constants $0 < c_{\Phi} < C_{\Phi} < \infty$ such that $c_{\Phi} \leq v_{\min}(\Phi_{n,\lambda}) \leq v_{\max}(\Phi_{n,\lambda}) \leq C_{\Phi}$ almost surely, for n large enough, where $v_{\min}(\Phi_{n,\lambda})$ and $v_{\max}(\Phi_{n,\lambda})$ are the minimum and maximum eigenvalues of $\Phi_{n,\lambda}$ respectively.

Proof. It's easy to see that for any vector $(\beta^{\top}, \gamma^{*\top})^{\top}$,

$$\begin{aligned}
& (\beta^{\top}, \gamma^{*\top})^{\top} \Phi_{n,0} (\beta^{\top}, \gamma^{*\top})^{\top} \\
&= \frac{1}{n} \sum_{i=1}^n (\beta^{\top}, \gamma^{*\top})^{\top} \begin{pmatrix} \mathbf{X}_i \mathbf{X}_i^{\top} & \mathbf{X}_i \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^{\top} \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \mathbf{X}_i^{\top} & \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^{\top} \end{pmatrix} \\
& \quad (\beta^{\top}, \gamma^{*\top})^{\top} \\
&= \|\mathbf{X}^{\top} \beta + \mathbf{Z}^{\top} \psi_{\gamma}(\mathbf{U})\|_n^2 \\
&= \|\eta_{\beta, \gamma}\|_n^2,
\end{aligned}$$

where $\eta_{\beta, \gamma}(\mathbf{t}) = \mathbf{x}^{\top} \beta + \mathbf{z}^{\top} \psi_{\gamma}(\mathbf{u})$ and $\psi_{\gamma}(\mathbf{u}) = (\psi_{\gamma_1}(\mathbf{u}), \dots, \psi_{\gamma_{p_2}}(\mathbf{u}))^{\top}$ with $\psi_{\gamma_k}(\mathbf{u}) = \gamma_k^{\top} \mathbf{B}(\mathbf{u})$ and $\gamma_k = \mathbf{Q}_2 \gamma_k^*$ for $k = 1, \dots, p_2$. By Lemma 3 and Lemma 6, we have

$$\begin{aligned}
c(1 - R_n) (\|\beta\|^2 + |\Delta|^2 \|\gamma\|^2) &\leq (1 - R_n) \|\eta_{\beta, \gamma}\|^2 \leq \|\eta_{\beta, \gamma}\|_n^2 \\
&\leq (1 + R_n) \|\eta_{\beta, \gamma}\|^2 \leq C(1 + R_n) (\|\beta\|^2 + |\Delta|^2 \|\gamma\|^2),
\end{aligned}$$

where $\gamma = (\gamma_1^{\top}, \dots, \gamma_{p_2}^{\top})^{\top}$.

Note that $R_n \rightarrow 0$ as $n \rightarrow \infty$ and by Assumption (A8), $|\Delta|^2 \rightarrow 0$ as $n \rightarrow \infty$, therefore,

$$c \|\beta\|^2 \leq (\beta^{\top}, \gamma^{*\top})^{\top} \Phi_{n,0} (\beta^{\top}, \gamma^{*\top})^{\top} \leq C \|\beta\|^2, \quad (\text{A.8})$$

almost surely for n large enough.

By the proof of Proposition 1 in Lai and Wang (2013), we have the inequality

$$\sum_{k=1}^{p_2} \left\| \sum_{j \in \mathcal{J}} \gamma_{kj} B_j \right\|_{\mathcal{E}_v}^2 \leq C |\Delta|^{-4} \sum_{k=1}^{p_2} \left\| \sum_{j \in \mathcal{J}} \gamma_{kj} B_j \right\|_{L_2}^2,$$

thus we have $\sum_{k=1}^{p_2} \left\| \sum_{j \in \mathcal{J}} \gamma_{kj} B_j \right\|_{\mathcal{E}_v}^2 = O(|\Delta|^{-2} \|\gamma\|^2)$, then by Assumption (A8), we obtain that

$$\begin{aligned}
& (\boldsymbol{\beta}^\top, \boldsymbol{\gamma}^{*\top}) \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{n} \mathbf{D}_\Lambda \end{pmatrix} (\boldsymbol{\beta}^\top, \boldsymbol{\gamma}^{*\top})^\top \\
&= \frac{1}{n} \boldsymbol{\gamma}^{*\top} \mathbf{D}_\Lambda \boldsymbol{\gamma}^* \\
&= \frac{1}{n} \boldsymbol{\gamma}^{*\top} (\Lambda \otimes \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2) \boldsymbol{\gamma}^* = \frac{1}{n} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^{*\top} \mathbf{Q}_2^\top \mathbf{P} \mathbf{Q}_2 \boldsymbol{\gamma}_k^* \\
&= \frac{1}{n} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^{*\top} \mathbf{Q}_2^\top \{ \langle B_j, B_{j'} \rangle_{\mathcal{E}_v} \}_{j, j' \in \mathcal{J}} \mathbf{Q}_2 \boldsymbol{\gamma}_k^* \\
&= \frac{1}{n} \sum_{k=1}^{p_2} \lambda_k \boldsymbol{\gamma}_k^\top \{ \langle B_j, B_{j'} \rangle_{\mathcal{E}_v} \}_{j, j' \in \mathcal{J}} \boldsymbol{\gamma}_k \\
&= O \{ \lambda_{\max} |\Delta|^{-2} n^{-1} \|\gamma\|^2 \} \\
&= o \{ \|\gamma\|^2 \}, \tag{A.9}
\end{aligned}$$

where $\lambda_{\max} = \max_{1 \leq k \leq p_2} \lambda_k$ and $\boldsymbol{\gamma}_k = \mathbf{Q}_2 \boldsymbol{\gamma}_k^*$ for $k = 1, \dots, p_2$. The desired result follows (A.8) and (A.9). $\square$

A.2 Consistency of Penalized Quasi-likelihood Estimators in the GPLSVCM

By Assumption (A1) and Lemma 4, there exists $\tilde{\alpha}_k(\mathbf{u}) = \mathbf{B}^*(\mathbf{u})^\top \tilde{\boldsymbol{\gamma}}^*$ which is the best approximation to α_k with the approximation rate at $\|\alpha_k - \tilde{\alpha}_k\|_\infty = O(|\Delta|^{d+1})$. Denote that $\eta_i(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) = \mathbf{X}_i^\top \boldsymbol{\beta} + \sum_{k=1}^{p_2} Z_{ik} \mathbf{B}^*(\mathbf{U}_i)^\top \boldsymbol{\gamma}_k^*$ for $i = 1, \dots, n$, then we have

$$DL^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) = - \sum_{i=1}^n q_1 \{ \eta_i(\boldsymbol{\beta}, \boldsymbol{\gamma}^*), Y_i \} \begin{pmatrix} \mathbf{X}_i \\ \{ \mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i) \} \end{pmatrix} + \begin{pmatrix} 0 \\ \mathbf{D}_\Lambda \boldsymbol{\gamma}^* \end{pmatrix}$$

$$\begin{aligned}
D^2 L^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) = & \\
& - \sum_{i=1}^n q_2(\eta_i(\boldsymbol{\beta}, \boldsymbol{\gamma}^*), Y_i) \begin{pmatrix} \mathbf{X}_i \mathbf{X}_i^\top & \mathbf{X}_i \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \mathbf{X}_i^\top & \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \end{pmatrix} \\
& + \begin{pmatrix} 0 & 0 \\ 0 & \mathbf{D}_\Lambda \end{pmatrix}.
\end{aligned}$$

Note that $\eta_i^0 = \sum_{l=1}^{p_1} X_{il} \beta_l + \sum_{k=1}^{p_2} Z_{ik} \alpha_k(\mathbf{U}_i)$, for the notation simplicity, denote $\hat{\eta}_i = \eta_i(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}^*)$ and $\tilde{\eta}_i = \eta_i(\boldsymbol{\beta}, \tilde{\boldsymbol{\gamma}}^*)$ where $\tilde{\boldsymbol{\gamma}}^* = (\tilde{\boldsymbol{\gamma}}_1^{*\top}, \dots, \tilde{\boldsymbol{\gamma}}_{p_2}^{*\top})^\top$ and $\dot{g}^{-1}(\eta) = \{g^{-1}(\eta)\}'$.

LEMMA 8. *Under Assumptions (A1)-(A7), we have*

$$\begin{aligned}
\left| \frac{1}{n} DL^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) \right| &= O_{a.s} \left(|\Delta| n^{-1/2} \log^{1/2} n + |\Delta|^{d+3} + \frac{\lambda_{\max}}{n|\Delta|^2} \right), \\
\left\| \frac{1}{n} D^2 L^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) \right\| &= O_{a.s} \left(n^{-1/2} \log^{1/2} n + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right).
\end{aligned}$$

Proof. Let $\tilde{\eta}(\mathbf{t}) = \mathbf{x}^\top \boldsymbol{\beta} + \sum_{k=1}^{p_2} z_k \mathbf{B}^*(\mathbf{u})^\top \tilde{\boldsymbol{\gamma}}_k^*$, then $\|\tilde{\eta} - \eta\|_\infty = O(|\Delta|^{d+1})$. By Assumption (A2), we have

$$\begin{aligned}
n^{-1} DL^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) &= -\frac{1}{n} \sum_{i=1}^n q_1\{\tilde{\eta}_i, Y_i\} \begin{pmatrix} \mathbf{X}_i \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ \frac{1}{n} \mathbf{D}_\Lambda \tilde{\boldsymbol{\gamma}}^* \end{pmatrix} \\
&= -\frac{1}{n} \sum_{i=1}^n \frac{Y_i - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V\{g^{-1}(\tilde{\eta}_i)\}} \dot{g}^{-1}(\tilde{\eta}_i) \begin{pmatrix} \mathbf{X}_i \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ \frac{1}{n} \mathbf{D}_\Lambda \tilde{\boldsymbol{\gamma}}^* \end{pmatrix} \\
&= -\frac{1}{n} \sum_{i=1}^n \frac{Y_i - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) \{1 + o(1)\} \begin{pmatrix} \mathbf{X}_i \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \end{pmatrix} \\
&\quad + \begin{pmatrix} \mathbf{0} \\ \frac{1}{n} \mathbf{D}_\Lambda \tilde{\boldsymbol{\gamma}}^* \end{pmatrix} \\
&= \mathbf{V}_1\{1 + o(1)\} + \mathbf{V}_2\{1 + o(1)\} + \mathbf{V}_3,
\end{aligned}$$

where

$$\begin{aligned}\mathbf{V}_1 &= -\frac{1}{n} \sum_{i=1}^n \frac{Y_i - g^{-1}(\eta_i^0)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) \begin{pmatrix} \mathbf{X}_i \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \end{pmatrix}, \\ \mathbf{V}_2 &= -\frac{1}{n} \sum_{i=1}^n \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) \begin{pmatrix} \mathbf{X}_i \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \end{pmatrix}, \\ \mathbf{V}_3 &= \begin{pmatrix} \mathbf{0} \\ \frac{1}{n} \mathbf{D}_\Lambda \tilde{\gamma}^* \end{pmatrix}.\end{aligned}$$

For vector $\mathbf{V}_1$, we have

$$\mathbb{E} \left[\frac{Y_i - g^{-1}(\eta_i^0)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} \right] = 0, \quad \mathbb{E} \left[\frac{Y_i - g^{-1}(\eta_i^0)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} \right]^2 = O(1).$$

According to Assumption (A3) and applying the Bernstein inequality, for any $l = 1, \dots, p_1$, we obtain that

$$\frac{1}{n} \sum_{i=1}^n \frac{Y_i - g^{-1}(\eta_i^0)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} = O_{\text{a.s.}} \left(n^{-1/2} \log^{1/2} n \right). \quad (\text{A.10})$$

Similarly, for any $j \in \mathcal{J}$, $k = 1, \dots, p_2$,

$$\frac{1}{n} \sum_{i=1}^n \frac{Y_i - g^{-1}(\eta_i^0)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) Z_{ik} B_j^*(\mathbf{U}_i) = O_{\text{a.s.}} \left(|\Delta| n^{-1/2} \log^{1/2} n \right). \quad (\text{A.11})$$

For the vector $\mathbf{V}_2$, we first look into the term $n^{-1} \sum_{i=1}^n \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il}$. We write

$$\frac{1}{n} \sum_{i=1}^n \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} = \sum_{i=1}^n \xi_i + \mathbb{E} \left[\frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V \{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} \right],$$

where

$$\xi_i = \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{n\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} - \mathbb{E} \left[\frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{n\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} \right].$$

Note that

$$\mathbb{E} \left[\frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} \right] = O(|\Delta|^{d+1}). \quad (\text{A.12})$$

Then we have, for any $r \geq 3$,

$$\mathbb{E} |\xi_i|^r \leq \{Cn^{-1} |g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)| \|X_l\|_\infty\}^{r-2} \mathbb{E} |\xi_i|^2 \leq \{Cn^{-1} |\Delta|^{d+1}\}^{r-2} \mathbb{E} |\xi_i|^2,$$

thus $\{\xi_i\}_{i=1}^n$ satisfy the Cramer's condition with constant $Cn^{-1}|\Delta|^{d+1}$. Also, note that

$$\begin{aligned} \mathbb{E} |\xi_i|^2 &= \mathbb{E} \left| \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{n\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} \right|^2 - \left[\mathbb{E} \left\{ \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{n\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} \right\} \right]^2 \\ &= O(n^{-2} |\Delta|^{2d+2}). \end{aligned}$$

By the the Bernstein inequality, we have for $\delta > 0$ large enough,

$$P \left\{ \left| \sum_{i=1}^n \xi_i \right| \geq \delta |\Delta|^{d+1} n^{-1/2} \log^{1/2} n \right\} \leq 2 \exp \left\{ \frac{-\delta^2 \log n}{4 + Cn^{-1/2} \log^{1/2} n} \right\} \leq 2n^{-4}.$$

Therefore,

$$\sum_{n=1}^{\infty} P \left\{ \left| \sum_{i=1}^n \xi_i \right| \geq \delta |\Delta|^{d+1} n^{-1/2} \log^{1/2} n \right\} < \infty.$$

Then, Borel-Cantelli Lemma entails that

$$\sum_{i=1}^n \xi_i = O_{\text{a.s.}} \left\{ |\Delta|^{d+1} n^{-1/2} \log^{1/2} n \right\}. \quad (\text{A.13})$$

Combining (A.11) and (A.12), we obtain that

$$\frac{1}{n} \sum_{i=1}^n \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} = O_{a.s.} \left(|\Delta|^{d+1} n^{-1/2} \log^{1/2} n + |\Delta|^{d+1} \right). \quad (\text{A.14})$$

Similarly,

$$\frac{1}{n} \sum_{i=1}^n \frac{g^{-1}(\eta_i^0) - g^{-1}(\tilde{\eta}_i)}{\sigma^2 V\{g^{-1}(\eta_i^0)\}} \dot{g}^{-1}(\eta_i^0) X_{il} B_j^*(\mathbf{U}_i) = O_{a.s.} \left(|\Delta|^{d+2} n^{-1/2} \log^{1/2} n + |\Delta|^{d+3} \right). \quad (\text{A.15})$$

For the vector $\mathbf{V}_3$, first note that by Theorem 2.19 in [Lai and Schumaker \(2007\)](#), the (j, j') th element of $\mathbf{P}$ is $O(|\Delta|^{-2})$ for any $j, j' \in \mathcal{J}$. Besides, $|\tilde{\gamma}^*| = O(1)$. Hence, we have

$$|\mathbf{V}_3| = O(\lambda_{\max} n^{-1} |\Delta|^{-2}). \quad (\text{A.16})$$

Combining (A.10), (A.11), (A.14), (A.15) and (A.16), we obtain that

$$\begin{aligned} \left| \frac{1}{n} DL^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) \right| &= O_{a.s.} \left(|\Delta| n^{-1/2} \log^{1/2} n + |\Delta|^{d+3} + \frac{\lambda_{\max}}{n|\Delta|^2} \right), \\ \left\| \frac{1}{n} D^2 L^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*) \right\| &= O_{a.s.} \left(n^{-1/2} \log^{1/2} n + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right). \end{aligned}$$

□

LEMMA 9. *If $(\bar{\boldsymbol{\beta}}^\top, \bar{\boldsymbol{\gamma}}^{*\top})^\top$ is the vector that satisfies that $\left\| (\bar{\boldsymbol{\beta}}^\top, \bar{\boldsymbol{\gamma}}^{*\top})^\top - (\boldsymbol{\beta}^\top, \tilde{\boldsymbol{\gamma}}^{*\top})^\top \right\| = O_{a.s.}(1)$, then, under Assumptions (A2)-(A7), there exist constants c and C , such that*

$$c\mathbf{I} \leq - \left\{ \frac{1}{n} D^2 L^P(\bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\gamma}}^*) \right\}^{-1} \leq C\mathbf{I}$$

almost surely for n large enough.

Proof. Let $\bar{\eta}_i = \eta_i(\bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\gamma}}^*)$, we have

$$\begin{aligned}
& \frac{1}{n} D^2 L^P(\bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\gamma}}^*) \\
&= -\frac{1}{n} \sum_{i=1}^n q_2(\bar{\eta}_i, Y_i) \begin{pmatrix} \mathbf{X}_i \mathbf{X}_i^\top & \mathbf{X}_i \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \mathbf{X}_i^\top & \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \end{pmatrix} \\
&\quad + \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{n} \mathbf{D}_\Lambda \end{pmatrix} \\
&= -\frac{1}{n} \sum_{i=1}^n [\{Y_i - g^{-1}(\eta_i^0)\} \rho'_1 \{g^{-1}(\bar{\eta}_i)\} + \{g^{-1}(\bar{\eta}_i) - g^{-1}(\eta_i^0)\} \rho'_1 \{g^{-1}(\bar{\eta}_i)\} - \rho_2 \{g^{-1}(\bar{\eta}_i)\}] \\
&\quad \times \begin{pmatrix} \mathbf{X}_i \mathbf{X}_i^\top & \mathbf{X}_i \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \mathbf{X}_i^\top & \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \end{pmatrix} + \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{n} \mathbf{D}_\Lambda \end{pmatrix} \\
&= -\frac{1}{n} \sum_{i=1}^n \left[\rho_2 \{g^{-1}(\bar{\eta}_i)\} \begin{pmatrix} \mathbf{X}_i \mathbf{X}_i^\top & \mathbf{X}_i \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \\ \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \mathbf{X}_i^\top & \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\} \{\mathbf{Z}_i \otimes \mathbf{B}^*(\mathbf{U}_i)\}^\top \end{pmatrix} \right] \\
&\quad \times \{1 + o_{a.s.}(1)\} + \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{n} \mathbf{D}_\Lambda \end{pmatrix}.
\end{aligned}$$

Therefore, by Assumption (A4) and Lemma 7, we obtain the desired result that

$$c\mathbf{I} \leq n^{-1} D^2 L^P(\bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\gamma}}^*) \leq C\mathbf{I},$$

almost surely for n large enough. □

Proof of Theorem 1 and Theorem 2.

Proof. We now prove that

$$\left\| (\hat{\boldsymbol{\beta}}^\top, \hat{\boldsymbol{\gamma}}^{*\top})^\top - (\boldsymbol{\beta}^\top, \tilde{\boldsymbol{\gamma}}^{*\top})^\top \right\| = O_{a.s.} \left(n^{-1/2} \log^{1/2} n + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right). \quad (\text{A.17})$$

Since $(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}^*)$ is the minimizer of $L^P(\boldsymbol{\beta}, \boldsymbol{\gamma}^*)$, we have $DL^P(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}^*) = 0$. Then, by mean value

theorem, we have

$$(\widehat{\boldsymbol{\beta}}^\top, \widehat{\boldsymbol{\gamma}}^{*\top})^\top - (\boldsymbol{\beta}^\top, \widetilde{\boldsymbol{\gamma}}^{*\top})^\top = -\{D^2 L^P(\bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\gamma}}^*)\}^{-1} D L^P(\boldsymbol{\beta}, \widetilde{\boldsymbol{\gamma}}^*),$$

where $(\bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\gamma}}^*)$ is some value between $(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\gamma}}^*)$ and $(\boldsymbol{\beta}, \widetilde{\boldsymbol{\gamma}}^*)$. Therefore, by Lemma 8 and 9, we obtain (A.17). Then, by Assumption (A5), we have

$$\begin{aligned}\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\| &= O_{a.s.} \left(n^{-1/2} \log^{1/2} n + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right), \\ \|\widehat{\boldsymbol{\gamma}}^* - \widetilde{\boldsymbol{\gamma}}^*\| &= O_{a.s.} \left(n^{-1/2} \log^{1/2} n + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right).\end{aligned}$$

By Lemma 1 and Lemma 2, we then have

$$\begin{aligned}\|\widehat{\alpha}_k - \widetilde{\alpha}_k\|_{L_2} &= \|\mathbf{B}^\top \mathbf{Q}(\widehat{\boldsymbol{\gamma}}_k^* - \widetilde{\boldsymbol{\gamma}}_k^*)\|_{L_2} \\ &= O_{a.s.} \left[|\Delta|^{-1} \left\{ \left(\frac{\log n}{n} \right)^{1/2} + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right\} \right].\end{aligned}$$

Then by the triangle inequality, we obtain that

$$\begin{aligned}\|\widehat{\alpha}_k - \alpha_k\|_{L_2} &\leq \|\widehat{\alpha}_k - \widetilde{\alpha}_k\|_{L_2} + \|\widetilde{\alpha}_k - \alpha_k\|_{L_2} \leq \|\widehat{\alpha}_k - \widetilde{\alpha}_k\|_{L_2} + \|\widetilde{\alpha}_k - \alpha_k\|_\infty \\ &= O_{a.s.} \left[|\Delta|^{-1} \left\{ \left(\frac{\log n}{n} \right)^{1/2} + |\Delta|^{d+2} + \frac{\lambda_{\max}}{n|\Delta|^3} \right\} \right].\end{aligned}$$

□

Appendix B

R Code for Chapter 3

In this section, we provide all the codes used in chapter 3.

```
R> family=poisson()
R> ngrid = 0.02
R> pop = Datagenerator(family, ngrid)
R> N=nrow(pop)
R> ind_s=sample(N,n,replace=FALSE)
R> data=as.matrix(pop[ind_s,])
R> Y=as.matrix(data[,1])
R> X=cbind(rep(1,length(Y)),data[,c(6:9)])
R> U=data[,c(10:11)]
R> ind_l=c(1,4,5)
R> ind_nl=c(2,3)
R> library(Triangulation)
R> data('boundary.horseshoe')
R> Tr=TriMesh(boundary.horseshoe, n = 8)$Tr
R> V=TriMesh(boundary.horseshoe, n = 8)$V
R> lambda=exp(seq(log(0.0001),log(10),length.out=10))
R> mfit = gplsvcm_fit(Y=Y,X=X,ind_l=ind_l,ind_nl=ind_nl,U=U,V=V,Tr=Tr,d=2,
```

```

+   r=1,lambda=lambda,family=family)
R> set.seed(123)
R> compute_PIs(Y_train,X_train,ind_l,ind_n1,U_train,X_pred,U_pred,V,Tr,d = 2
+   ,r = 1,lambda=lambda,family=family,off = 0,r_theta = c(2, 8),eps = 0.01,method = "CV+"
+   ,cp = 0.95,nfold = 10)
R> uvpop=cbind(pop$u,pop$v)
R> lambda_start=0.1; lambda_end=0.2; nlambda=50;
R> lambda1=exp(seq(log(lambda_start),log(lambda_end),length.out=nlambda))
R> lambda2=exp(seq(log(lambda_start),log(lambda_end),length.out=nlambda))
R> mfit=gplsvcm_aglasso(Y=Y,X=X,uvpop=uvpop,U=U,index=ind_s,V=V,Tr=Tr,
+   d=2,r=1,penalty="agrLasso",criteria="BIC", lambda1=lambda1,
+   lambda2=lambda2,family=family,lambda=lambda)
R> library(GPLSVCM)
R> TriNYC<- TriMesh(as.matrix(NYCBoundary), n = 18)
R> Tr<-TriNYC$Tr; V<-TriNYC$V
R> lambda<- exp(seq(log(0.001),log(10),length.out=10))
R> Y<- NYCfelony$Freq
R> X<- as.matrix(NYCfelony[,c("Population","Poverty","Marriage","Youth",
+   "Unemployed","Household","Education","Male")])
R> U<- as.matrix(NYCfelony[,c("lon","lat")])
R> MIDFNYCfelony<- gplsvcm_aglasso(Y,X,U,V,Tr,d=2,r=1,family=poisson(),
+   lambda=lambda)
R> MIDFNYCfelony$constant_varname
R> MIDFNYCfelony$varying_varname
R> fitNYCfelony$beta_hat
R> fitNYCfelony$alpha_hat
R> fitNYCfelony$theta_hat
R> mean((fitNYCfelony$fitted.values-fitNYCfelony$Y)^2,na.rm = TRUE)
R> cv_gplsvcm(Y,X,fitNYCfelony$ind_l,fitNYCfelony$ind_n1,U,V,Tr,d=2,r=1,

```

```
R> + lambda=lambda,family=poisson())  
R> gplsvcm_plot(fitNYCfelony,gridnumber=500,xlab=rep("lon",7),  
R> + ylab=rep("lat",7),main=rep(expression(paste("The Estimated Surface for"  
R> + , " ",hat(alpha))),7))
```