Spatial autoregressive partially linear varying coefficient models

Jingru Mu, Guannan Wang & Li Wang

To cite this article: Jingru Mu, Guannan Wang & Li Wang (2020) Spatial autoregressive partially linear varying coefficient models, Journal of Nonparametric Statistics, 32:2, 428-451, DOI: 10.1080/10485252.2020.1759596

To link to this article: https://doi.org/10.1080/10485252.2020.1759596

View supplementary material

Published online: 28 Apr 2020.

Submit your article to this journal

Article views: 42

View related articles

View Crossmark data
Spatial autoregressive partially linear varying coefficient models

Jingru Mu\textsuperscript{a}, Guannan Wang\textsuperscript{b} and Li Wang\textsuperscript{c}

\textsuperscript{a}Department of Statistics, Kansas State University, Manhattan, KS, USA; \textsuperscript{b}Department of Mathematics, William & Mary College, Williamsburg, VA, USA; \textsuperscript{c}Department of Statistics, Iowa State University, Ames, IA, USA

ABSTRACT

In this article, we consider a class of partially linear spatially varying coefficient autoregressive models for data distributed over complex domains. We propose approximating the varying coefficient functions via bivariate splines over triangulation to deal with the complex boundary of the spatial domain. Under some regularity conditions, the estimated constant coefficients are asymptotically normally distributed, and the estimated varying coefficients are consistent and possess the optimal convergence rate. A penalized bivariate spline estimation method with a more flexible choice of triangulation is proposed. We further develop a fast algorithm to calculate the geodesic distance. The proposed method is much more computationally efficient than the local smoothing methods, and thus capable of handling large scales of spatial data. In addition, we propose a model selection approach to identify predictors with constant and varying effects. The performance of the proposed method is evaluated by simulation examples and the Sydney real estate dataset.

1. Introduction

Recently, the wide availability of data observed over space, due to the widespread collection of networks and inexpensive geographical information systems, has stimulated the spatial data analysis. The scale and the complexity of the spatial data at the moment are far beyond our imagination. Complex data call for statistical models that are sufficiently flexible to adapt to underlying signals. Varying coefficient models have attracted significant attention in semiparametric or nonparametric regression studies, see for examples Hastie and Tibshirani (1993), Huang, Wu, and Zhou (2002), Fan and Huang (2005), and Yang, Park, Xue, and Härdle (2006). In spatial regression, spatially varying coefficient models (SVCMs) have also gained widespread popularity in recent years, enhancing the capability of spatial analysis by exploring spatial non-stationarity of a regression relationship in geo-referenced data analysis.

To estimate the varying coefficient functions in SVCMs, there are three popular methods, including Bayesian approach (Gelfand, Kim, Sirmans, and Banerjee 2003), local
smoothing method (Brusdon, Fotheringham, and Charlton 1996; Qingguo 2013) and basis expansion approach (Mu, Wang, and Wang 2018). Most existing articles on SVCMs focused on the modeling and methodology developments (Fotheringham, Brusdon, and Charlton 2002), and those with theoretical justifications usually considered the situation where the data are observed on grid points over a rectangular domain. Mu et al. (2018) studied the asymptotic properties of the estimator of SVCM for spatial data collected over a complex domain, however, the errors are assumed to be independent, which is too strong to be realistic for spatial data.

To incorporate the spatial dependence and balance the interpretability and flexibility of the SVCM, in this article, we consider a spatial autoregressive partially linear varying coefficient model (SAR-PLVCM). For $i = 1, \ldots, n$, let $Y_i$ represent the response variable, $Z_i = (Z_{i1}, \ldots, Z_{ip_1})^\top$ be a $p_1$-dimensional vector of explanatory variables which are linearly associated with the response variable, and $X_i = (X_{i1}, \ldots, X_{ip_2})^\top$ be a $p_2$-dimensional vector of explanatory variables, which have varying relationship with the response across different locations. Next, let $U_i = (U_{i1}, U_{i2})^\top$ be the location of the $i$-th observation randomly sampled from an arbitrary shaped spatial domain, $\Omega$. Then the SAR-PLVCM can be expressed as:

$$Y_i = \alpha_0 \sum_{j \neq i} w_{ij} Y_j + \sum_{l=1}^{p_1} Z_{il} \gamma_{0l} + \sum_{k=1}^{p_2} X_{ik} \beta_{0k}(U_i) + \epsilon_i, \quad i = 1, \ldots, n$$

where $\alpha_0 \in [0, 1]$ is a global parameter, $w_{ij}$ is the weight of the neighbor effects, satisfying $w_{ii} = 0$, and $\sum_{j \neq i} w_{ij} = 1$ for any $i, j = 1, \ldots, n$. $\gamma_{0l}$’s are unknown constant coefficient parameters, and $\beta_{0k}(\cdot)$’s are unknown varying-coefficient functions, the $\epsilon_i$’s are i.i.d random noises with $E(\epsilon_i) = 0$ and $\text{Var}(\epsilon_i) = \sigma_0^2$, and each $\epsilon_i$ is independent of $Z_i$ and $X_i$.

Model (1) encompasses several existing models as special cases, such as the SVCM in Mu et al. (2018) when all the covariates are assumed to have a varying effect on the response and neighbor effects are not considered; and the SAR model when all the covariates are assumed to have a constant effect on the response.

To estimate the unknown parameters in (1), Sun, Yan, Zhang, and Lu (2014) proposed a profile likelihood-based estimation procedure using the local smoothing technique. However, with the increasing volume of the spatial data, the computation burden of the local smoothing method is extremely high and usually out of the reach of a typical personal computer. In addition, although the traditional local smoothing method works well for a rectangular domain, it will encounter the well known ‘leakage’ problem when smoothing across the complex boundary which is common in practice. To overcome those difficulties, in this article, we propose using the bivariate spline over triangulation method for estimation. Furthermore, to handle the spatial dependence, SAR-PLVCM adds a weighted average of the neighbors to the explanatory variables. As a result, to estimate the SAR-PLVC model in (1), one very important step is to determine the weight matrix $W = \{w_{ij}\}_{i,j=1}^n$, which is typically based on the distance between observations. One conventional way to define the distance is by using the Euclidean distance. However, it completely ignores the shape of the domain when finding the shortest path from the start point to the endpoint. For a complicated and irregular shaped domain, the line segment that connects two points is often not included in the domain, and thus the weights based on Euclidean distance may not be appropriate. A better strategy is using the geodesic distance, which constrains the path.
between two points to be within the shape of the spatial domain. For example, Wang and Ranalli (2007) proposed the great-circle distance, which is the shortest distance measured along the surface of a sphere. They considered a restricted graph with every node only connected to its \( k \) nearest neighbors. However, \( k \) must be carefully selected to make sure that \( G \) is connected and avoid the case that endpoint is also among the \( k \)-nearest neighbors of the other. In addition, the computational burden dramatically increases for a large-scale dataset. To approximate the geodesic distance accurately and efficiently, in this article we propose two algorithms based on the triangulation technique.

Another major concern when fitting model (1) is the lack of prior knowledge on true model structure, specifically, which coefficients are constant and which are really varying over space. In this article, we propose a backward selection via Bayesian information criterion (BIC) for determining the linear and varying components in the SAR-PLVCM. Our numerical studies confirm the superb performances of our method in terms of estimation accuracy, model structure selection, and computational time.

The rest of the article is organized as follows. In Section 2 we present our estimation procedure. Section 3 is devoted to the asymptotic analysis of the proposed estimators. In Section 4, we discuss the details of the implementation, such as how to estimate the geodesic distance, and how to choose the penalty, and how to separate the constant and varying coefficient in the SAR-PLVCM. Section 5 presents simulation results comparing our method with its competitors. Section 6 illustrates our method using the Sydney housing price dataset. Technical details are provided in Appendix A and the online Supplementary Material.

2. Methodology

2.1. Model setting and likelihood

Denote \( n \)-dimensional vectors \( \mathbf{Y} = (Y_1, \ldots, Y_n)\top \), \( \epsilon = (\epsilon_1, \ldots, \epsilon_n)\top \) and \( \mu_0 = (\mu_{01}, \ldots, \mu_{0n})\top \), where \( \mu_{0i} = \mathbf{Z}_i\top \mathbf{y}_0 + \mathbf{X}_i\top \beta_0(\mathbf{U}_i), \ i = 1, \ldots, n \). Let \( \mathbf{y}_0 = (\gamma_{01}, \ldots, \gamma_{0p})\top \) and \( \beta_0(\mathbf{u}) = (\beta_{01}(\mathbf{u}), \ldots, \beta_{0p_2}(\mathbf{u}))\top \). Further let \( \mathbf{W} = (w_{ij}) \) be the \( n \times n \) weight matrix. Then model (1) can be written as the following matrix form:

\[
\mathbf{Y} = \alpha_0 \mathbf{WY} + \mu_0 + \epsilon. \quad (2)
\]

For any \( \alpha \), denote \( \mathbf{T}(\alpha) = \mathbf{I}_n - \alpha \mathbf{W} \) and \( \mathbf{T} = \mathbf{T}(\alpha_0) \), then the equilibrium vector \( \mathbf{Y} \) is

\[
\mathbf{Y} = \mathbf{T}^{-1}(\mu_0 + \epsilon). \quad (3)
\]

If the noise term in (2) is assumed to have the normal distribution \( N(0, \sigma^2) \), then the log-likelihood function can be written as

\[
L_n(\alpha, \beta, \sigma^2) = -\frac{n}{2} \log(2\pi) - \frac{n}{2} \log(\sigma^2) + \log(|\mathbf{T}(\alpha)|) - \frac{1}{2\sigma^2} \{\mathbf{Y}(\alpha) - \mu\}^\top \{\mathbf{Y}(\alpha) - \mu\}, \quad (4)
\]

where, for any value of \( \alpha \),

\[
\mathbf{Y}(\alpha) = (Y_1(\alpha), \ldots, Y_n(\alpha))\top = \mathbf{T}(\alpha)\mathbf{Y}. \quad (5)
\]

Such gaussian likelihood in (4) is commonly used in the SAR model literature; see, for example, Lee (2004) and Sun et al. (2014). The gaussian assumption on the noise term is not
necessary in the technical proofs for theoretical results. We propose using the profile log-
likelihood approach to estimate the unknown parameters. For each fixed \( \alpha \), we construct
the estimator of \( \beta(\cdot) \) as \( \hat{\beta}(\cdot; \alpha) \). Next, we plug \( \hat{\beta}(\cdot; \alpha) \) into the log-likelihood function, then
we estimate \( (\hat{\beta}, \hat{\sigma}^2) \) by maximizing (4). Finally, we obtain the estimator \( \beta(\cdot; \hat{\alpha}) \) for \( \beta(\cdot) \).

### 2.2. Bivariate spline over triangulation

Note that in the above maximum likelihood estimation, we need to implement some
appropriate smoothing method to approximate the coefficient functions. We consider the
bivariate spline approximation over triangulations (BST) proposed in Mu et al. (2018).
Below we briefly describe the techniques of triangulations and BST when \( \alpha \) is fixed in (4).

As shown in Lai and Wang (2013) and Mu et al. (2018), triangulation is an effective
strategy to handle data distributed on the domain with complex boundaries. We use \( \tau \) to
denote a triangle over a plane, and then a triangulation, \( \triangle \), of \( \Omega \) is defined as a collection of
\( K \) triangles, \( \triangle = \{\tau_1, \ldots, \tau_K\} \), within which if any two triangles intersect, their intersection
is either a common vertex or a shared edge. Let \( |\tau| \) denote the longest edge length for
triangle \( \tau \in \triangle \), and \( \rho_\tau \) denote the inscribed circle radius with respect to \( \tau \). Define the
shape parameter of \( \tau \) as \( R_\tau = |\tau| / \rho_\tau \), and the size of \( \triangle \) by the length of the longest edge of
\( \triangle \), i.e., \( |\triangle| = \max(|\tau|, \tau \in \triangle) \).

For nonnegative integers \( d, r \), and a triangulation \( \triangle \), define \( \mathcal{S}'_{d}(\triangle) = \{ s \in \mathbb{C}^r(\Omega) : s|_{\tau} \in \mathbb{P}_d(\tau), \tau \in \triangle \} \), where \( \mathbb{C}^r(\Omega) \) is the collection of all \( r \)th continuously differentiable func-
tions over \( \Omega \), \( s|_{\tau} \) is the polynomial piece of spline \( s \) restricted on triangle \( \tau \), and \( \mathbb{P}_d \) is the space of all polynomials of degree less than or equal to \( d \). It has been proved in Lai and Schumaker (2007) that when the datasets are noise-free, a spline space with \( d = 3r + 2 \) has the
optimal approximation rate. In this article, we consider the spline space \( \mathcal{S} = \mathcal{S}'_{3r+2}(\triangle) \).

Given \( \{(U_i, Z_i, X_i, Y_i(\alpha))\}_{i=1}^n \), we consider the following minimization problem:

\[
\min_{\gamma \in \mathcal{R}, \gamma \in \mathcal{S}, k = 1, \ldots, p_2} \sum_{i=1}^n \left\{ Y_i(\alpha) - \sum_{l=1}^{p_1} Z_{il} \gamma_l - \sum_{k=1}^{p_2} X_{ik} g_k(U_i) \right\}^2,
\]

where \( g_k(\cdot), k = 1, \ldots, p_2 \), can be approximated by bivariate splines. For any \( k = 1, \ldots, p_2 \), denote \( \{B_{kj}\}_{j \in \mathcal{J}_k} \) the set of bivariate Bernstein basis polynomials for \( \mathcal{S} \). Then \( g_k(u) \) can be represented as the linear combination of Bernstein basis:

\[
g_k(u) = \sum_{j \in \mathcal{J}_k} B_{kj}(u) c_k = B_k(u)^\top c_k,
\]

where \( c_k = (c_{kj}, j \in \mathcal{J}_k)^\top \) is the spline coefficient vector. To guarantee the
smoothness requirement of the splines, it is necessary to add some constraints on the spline
coefficients. Denote \( H_k \) the constraint matrix on the coefficients \( c_k \), which depends on \( d, r \)
and the structure of the triangulation. We refer to Section B.2.1 of the Supplementary
Materials of Yu, Wang, Wang, Liu, and Yang (2019) for examples of \( H_k \).

Thus, we consider the following constrained minimization:

\[
\sum_{i=1}^n \left\{ Y_i(\alpha) - \sum_{l=1}^{p_1} Z_{il} \gamma_l - \sum_{k=1}^{p_2} X_{ik} B_k(U_i)^\top c_k \right\}^2, \quad \text{subject to } H_k c_k = 0, \quad k = 1, \ldots, p_2.
\]

Next, we eliminate the constraint by applying a QR decomposition on \( H_k^\top = Q_k R_k = (Q_{1,k} Q_{2,k})_{R_{1,k}} \), where \( Q_k \) is an orthogonal matrix and \( R_{1,k} \) is an upper triangle matrix,
the submatrix \( Q_{1,k} \) is the first \( r_k \) columns of \( Q_k \), where \( r_k \) is the rank of matrix \( H_k \), and \( R_{2,k} \) is a matrix of zeros. We then consider the minimization of \( c_k \) using \( c_k = Q_{2,k} \theta_k \), then it is guaranteed that \( H_k c_k = 0 \). Thus, the above minimization problem can be converted to a conventional regression problem without any constraint:

\[
\sum_{i=1}^{n} \left\{ Y_i(\alpha) - \sum_{l=1}^{p_1} Z_{il} \gamma_l - \sum_{k=1}^{p_2} X_{ik} B_k^T(U_i) Q_{2,k} \theta_k \right\}^2.
\]

(6)

For simplicity, we assume \( B(u) = B_1(u) = \ldots = B_{p_2}(u) = \{B_j(u)\}_{j \in J} \), then \( H_1 = \ldots = H_{p_2} \) and \( Q_{2,1} = \ldots = Q_{2,p_2} \). Denote \( \theta = (\theta_1^T, \ldots, \theta_{p_2}^T)^T \). Let \( B^*(U_i) = Q_{2}^T B(U_i) \). Then the minimization problem in (6) can be written as

\[
\sum_{i=1}^{n} \left\{ Y_i(\alpha) - \sum_{l=1}^{p_1} Z_{il} \gamma_l - \sum_{k=1}^{p_2} X_{ik} B^*(U_i)^T \theta_k \right\}^2.
\]

(7)

Let ‘\( \otimes \)’ denote the Kronecker product. Let \( Z = (Z_1, \ldots, Z_n)^T, X_B = (X_1 \otimes B^*(U_1), \ldots, X_n \otimes B^*(U_n))^T \), and \( \Phi = (Z, X_B) \). Solving the least squares problem in (7), we obtain

\[
(\hat{\gamma}(\alpha)^T, \hat{\theta}(\alpha)^T)^T = (\Phi^T \Phi)^{-1} \Phi^T Y(\alpha),
\]

where \( Y(\alpha) \) is in (5).

Therefore, for any fixed \( \alpha \), one obtains the bivariate spline estimator of \( c_k \) and \( \beta_k(\cdot) \) as follows:

\[
\hat{c}_k(\alpha) = Q_{2} \hat{\theta}_k(\alpha) \quad \text{and} \quad \hat{\beta}_k(u; \alpha) = B(u)^T \hat{c}_k(\alpha).
\]

### 2.3. Estimation of \( \alpha_0 \) and \( \sigma_0^2 \)

We now consider the maximization problem for estimating \( \alpha_0 \) and \( \sigma_0^2 \). Plugging \( \hat{\beta}(u; \alpha) \) into (4) and leaving out the constant term, we define

\[
\ell_n(\gamma, \theta, \sigma^2; \alpha) = -\frac{n}{2} \log(\sigma^2) + \log(|T(\alpha)|) - \frac{1}{2\sigma^2} \{Y(\alpha) - \hat{\mu}(\alpha)\}^T \{Y(\alpha) - \hat{\mu}(\alpha)\},
\]

(8)

where \( \hat{\mu}(\alpha) = P_\Phi Y(\alpha) \) with \( P_\Phi = \Phi^T \Phi)^{-1} \Phi^T \) being the projection matrix of \( \Phi \). Setting the partial derivative of the objective function on \( \sigma^2 \) to zero and solving the equation, we have

\[
\hat{\sigma}^2(\alpha) = \frac{1}{n} \{Y(\alpha) - \hat{\mu}(\alpha)\}^T \{Y(\alpha) - \hat{\mu}(\alpha)\} = \frac{1}{n} Y(\alpha)^T (I_n - P_\Phi) Y(\alpha).
\]

Then, the concentrated log-likelihood function of \( \alpha \) is

\[
\ell_n(\alpha) = -\frac{n}{2} (\log(2\pi) + 1) - n \log(\hat{\sigma}(\alpha)) + \log(|T(\alpha)|).
\]

(9)

Maximizing the concentrated log-likelihood in (9), we obtain the quasi-maximum likelihood estimator of \( \alpha_0 \), i.e.

\[
\hat{\alpha} = \arg \max_{\alpha \in [0,1]} \left\{ -n \log(\hat{\sigma}(\alpha)) + \log(|T(\alpha)|) \right\}.
\]

Then, we replace \( \alpha \) by \( \hat{\alpha} \) to obtain the estimator of \( \beta_0 \) and \( \sigma_0^2 \).
3. Asymptotic results

This section provides the asymptotic results for the proposed estimators. To facilitate the presentation of the main results, we introduce the following notations.

Denote $G = WT^{-1}$. By (2) and (3), and the fact that $\alpha_0 G + I_n = T^{-1}$, we have

$$Y = \mu_0 + \alpha_0 G\mu_0 + (\alpha_0 G + I_n)e = \mu_0 + \alpha_0 G\mu_0 + T^{-1}e.$$ 

For any $l = 1, \ldots, p_1$, let $g_{i,l}^*(U_i) = (g_{i,1}^*(U_i), \ldots, g_{i,p_2}^*(U_i))^\top$ be the vector of functions that minimizes $E[(Z_i - X_i g(U_i))^2, i = 1, \ldots, n$. Denote $g^* = (g_1^*, \ldots, g_{p_1}^*)^\top$, and

$$\Xi = E\left[(Z_i^\top - X_i^\top g^*(U_i))^\top (Z_i^\top - X_i^\top g^*(U_i))\right],$$

The proofs of the following theorems are provided in the Appendix.

**Theorem 3.1:** Suppose Assumptions (A1)–(A12) in the Appendix hold, or under Assumptions (A1)–(A11), $\sigma_0^2$, $\alpha_0$ are globally identifiable and $\tilde{\sigma}^2$, $\tilde{\alpha}$ are consistent of $\sigma_0^2$ and $\alpha_0$, respectively.

Theorem 3.1 shows the consistency of the estimators $\tilde{\alpha}$, $\tilde{\sigma}^2$, and based on this result, we are able to derive the following asymptotic normality. Let $\psi = (\alpha, \sigma^2, \gamma^\top)^\top$, and $\psi_0 = (\alpha_0, \sigma_0^2, \gamma_0^\top)^\top$. Denote

$$\Sigma_n = -E\left\{\frac{1}{n}\frac{\partial^2 \ell_n(\psi)}{\partial \psi \partial \psi'}|_{\psi = \psi_0}\right\},$$

$$\Omega_n = E\left\{\frac{1}{n}\frac{\partial \ell_n(\psi)}{\partial \psi} \frac{\partial \ell_n(\psi)}{\partial \psi'}|_{\psi = \psi_0}\right\} + E\left\{\frac{1}{n}\frac{\partial^2 \ell_n(\psi)}{\partial \psi \partial \psi'}|_{\psi = \psi_0}\right\}. \tag{13}$$

Lemma A.3 in the Supplementary Material gives the explicit formula of $\Sigma_n$ and $\Omega_n$.

**Theorem 3.2:** Under Assumptions (A1)–(A12) or Assumptions (A1)–(A11), $\tilde{\sigma}$, $\tilde{\alpha}$ and (A13) in the Appendix, if $n^{-1/2}K_n \log n \to 0$ as $n \to \infty$, then we have

$$\sqrt{n}(\tilde{\psi} - \psi_0) \to N(0, \Sigma^{-1} + \Sigma^{-1} \Omega \Sigma^{-1}),$$

where $\Sigma = \lim_{n \to \infty} \Sigma_n$ and $\Omega = \lim_{n \to \infty} \Omega_n$.

Theorem 3.2 implies that the constant coefficient estimators in the proposed method have the convergence rate of order $n^{-1/2}$ when $\Sigma$ is nonsingular. Using these two theoretical results, we can establish the convergence rate of the BST estimator $\tilde{\beta}_k$, $k = 1, \ldots, p_2$, as stated in Theorem 3.3 below.

**Theorem 3.3:** Under the same Assumptions of Theorem 3.1, for any $k = 1, \ldots, p_2$, the spline estimator $\tilde{\beta}_k(\cdot)$ of the varying coefficient $\beta_k(\cdot)$ is consistent and $\|\tilde{\beta}_k - \beta_k\|_2 = O_p(|\Delta|^{\ell+1} + n^{-1/2}|\Delta|^{-1})$.

Theorem 3.3 implies that if the number of triangles $K_n \asymp n^{1/(\ell+1)}$, then the spline estimators would satisfy $\|\tilde{\beta}_k - \beta_k\|_2^2 = O_p\left(n^{(\ell+1)/(\ell+2)}\right)$, which is also the optimal convergence rate in Stone (1982).
4. Implementation

In this section, we discuss some details in the implementation of the proposed method.

4.1. Distance and shortest path over irregular domain

There are two popular row-normalized weight functions in spatially autoregressive regression (SAR) models: (i) exponential weights:

\[ w_{ij} = \exp\left(-c_1 d_{ij}\right) / \sum_{k \neq i} \exp\left(-c_1 d_{ik}\right), \quad i \neq j, \quad c_1 > 0; \]

(ii) reciprocal (inverse) weights:

\[ w_{ij} = d_{ij}^{-c_2} / \sum_{k \neq i} (d_{ik}^{-c_2}), \quad i \neq j, \quad c_2 > 0, \]

where \( d_{ij} \) measures the distance between locations \( i \) and \( j \).

In the following, we develop two efficient algorithms to approximate the geodesic distance over a complex domain based on the triangulation technique. When the sample size is moderately large, for example, at hundreds or thousands level, we propose Algorithm 1 for calculating the distance between any sampled locations. In our extensive simulation studies, we find the number of auxiliary points \( m \) doesn’t need to be very large, usually, \( m = 1000 \) is sufficient to construct a fine triangulation. In Algorithm 1, \( \delta \) is used for determining the threshold of the convergence, which can be set as \( \frac{\|D(s)\|_F}{n} - \frac{\|D(s-1)\|_F}{n} \) or \( \frac{\|W(s)\|_F}{n} - \frac{\|W(s-1)\|_F}{n} \), depending on the interest is the distances or the weights.

Section S.5 in the Supplementary Material provides the comparison of the weights calculated based on the estimated distances by Algorithm 1, and those calculated based on the ‘oracle’ geodesic distances using the approach proposed by Miller and Wood (2014). It shows the convergence of this algorithm as \( S \) increases. Based on these numerical studies, we recommend setting \( S = 30–50 \) for small sample sizes and \( S = 10–30 \) for sample size larger than 2000. Algorithm 1 is much faster than the traditional methods in the literature, however, according to the simulation studies, it is still not fast enough when a dataset contains more than tens of thousands of observations. For such a large scale, we propose a fast version of Algorithm 1; see Algorithm 2 below. Simulation studies in Section S.5 in the Supplementary Material demonstrate the performance of Algorithm 2. From Table S.2, one sees that, when the sample size is large, Algorithm 2 is much faster than Algorithm 1 without sacrificing much accuracy. The Matlab code for Algorithms 1 and 2 can be found from https://github.com/funstatpackages/SARVCM.

4.2. Bivariate penalized spline

For fitting sparse spatial data, bivariate penalized splines over triangulation (Lai and Wang 2013, BPST) is more flexible than the unpenalized bivariate splines method referred to as the BST method in Section 2.2. To define the penalized spline method, for directions \( u_1 \) and \( u_2 \), denote \( D_{uj}^q f(x) \) as the \( q \)-th order derivative in the direction \( u_j \) at the point \( u \). Given \( \{(U_i, Z_i, X_i, Y_i(\alpha))\}_{i=1}^n \), we now consider the following penalized minimization:

\[
\min_{\gamma_j \in \mathbb{R}, l=0, \ldots, p_1, g_k \in S, k=0, \ldots, p_2} \frac{1}{n} \sum_{i=1}^n \left\{ Y_i(\alpha) - \sum_{l=1}^{p_1} Z_{il} \gamma_l - \sum_{k=1}^{p_2} X_{ik} g_k(U_i) \right\}^2 + \sum_{k=1}^{p_2} \lambda_k \mathcal{E}(g_k),
\]

where \( \mathcal{E}(g_k) = \int_{\Omega} \left\{ (D_{u_1}^2 g_k)^2 + 2(D_{u_1} D_{u_2} g_k)^2 + (D_{u_2}^2 g_k)^2 \right\} du_1 \ du_2, \) and \( \lambda_k > 0, k = 1, \ldots, p_2 \), are the tuning parameters which control the smoothness of the functions \( g_k \)'s. Adopting the bivariate splines approximation approach, we solve the following minimization
Algorithm 1: Algorithm to estimate the geodesic distance matrix.

Data: Location of sample observations \( \{U_i\}_{i=1}^n \) and boundary of the domain \( \Omega \)

Output: Distance matrix \( D = \{d_{ij}\}_{i,j=1}^n \)

1. Initialize: \( d_{ij} = \infty, \ s = 0 \)
2. while \( \delta > \) threshold and \( s < S \) do
3. (i) \( s = s + 1 \)
4. (ii) add \( m \) random auxiliary points over \( \Omega \)
5. (iii) construct triangulation mesh \( \triangle \) using the auxiliary points and the original sample points as vertices
6. (iv) for any two points \( U_i \) and \( U_j \) do
7. set \( d_{ij}^{(s)} = \min\{d_{ij}^{(s-1)}, \) the shortest path through the edges of \( \triangle \) based on the edge lengths\}
8. (v) calculate the distance matrix \( D^{(s)} = \{d_{ij}^{(s)}\}_{i,j=1}^n \)

problem:

\[
\sum_{i=1}^{n} \left\{ Y_i(\alpha) - \sum_{l=1}^{p_1} Z_{il} \gamma_l - \sum_{k=1}^{p_2} X_{ik} B_k(U_i) \right\}^2 + \sum_{k=1}^{p_2} \lambda_k c_k^\top P_k c_k,
\]

where \( P_k \) is the diagonal block penalty matrix satisfying that \( c_k^\top P_k c_k = E_{\nu}(B_k^\top c_k) \).

Similar to Section 2.2, we wipe out the smoothness constraint by the QR decomposition of \( H_k^\top \). Then the minimization problem is now converted to the following:

\[
\sum_{i=1}^{n} \left\{ Y_i(\alpha) - \sum_{l=1}^{p_1} Z_{il} \gamma_l - \sum_{k=1}^{p_2} X_{ik} B_k(U_i)Q_{2,k}^\top c_k \right\}^2 + \sum_{k=1}^{p_2} \lambda_k \theta_k^\top Q_{2,k}^\top P_k Q_{2,k}^\theta_k. \tag{14}
\]

Let \( \Lambda = \text{diag}(\lambda_1, \ldots, \lambda_{p_2}) \), and \( D_\Lambda = \text{diag}(0, \Lambda \otimes (Q_2^\top P Q_2)) \), where \( 0 \) is a \( p_1 \times p_1 \) matrix of zeros. Assuming the same set of spline basis functions is used for all the coefficient functions, we obtain that

\[
\begin{pmatrix} \hat{\gamma}^*(\alpha) \\ \hat{\theta}^*(\alpha) \end{pmatrix} = (\Phi^\top \Phi + D_\Lambda)^{-1} \Phi^\top Y(\alpha), \quad \hat{\mu}(\alpha) = \Phi(\Phi^\top \Phi + D_\Lambda)^{-1} \Phi^\top Y(\alpha) = S_\Lambda Y(\alpha).
\]

by solving the least squares problem in (14). We choose the best combination of \( (\lambda_1, \ldots, \lambda_{p_2}) \) by minimizing the generalized cross-validation (GCV):

\[
GCV = n(n - m)^{-2} \| Y(\alpha) - S_\Lambda Y(\alpha) \|_2^2, \tag{15}
\]

where \( m = \text{tr}(S_\Lambda) \) can be regarded as the effective degrees of freedom (edf) for the model.

4.3. Model selection

A natural question in SAR-PLVCM is how to determine which explanatory variables have a linear effect and which ones have a varying effect. As shown in Section 3, if the choice
Algorithm 2: Algorithm to estimate the geodesic distance for large scale datasets

**Data:** Location of sample observations \( \{U_i\}_{i=1}^n \) and boundary of the domain \( \Omega \)

**Output:** Geo-desic distance matrix \( D = \{d_{ij}\}_{i,j=1}^n \)

1. For each pair of locations, \( U_i \) and \( U_j \), check whether the line segment between them falls within \( \Omega \)
   2. **if** the line segment falls within the domain **then**
   3. Set \( d^0_{ij} \) = the Euclidean distance between the two locations
   4. **else**
   5. Set \( d^0_{ij} = \infty \)
   6. Create triangulation mesh, \( \triangle = \{\tau_1, \ldots, \tau_K\} \), and find the centroid of each triangle \( \tau_k \in \triangle, k = 1, \ldots, K \)
   7. Using Algorithm 1, find the geo-desic distance matrix, \( C = \{c_{kl}\}_{k,l=1}^n \), where \( c_{kl} \) is the geodesic distance between the centroids of \( \tau_k \) and \( \tau_l \)
   8. **for** any two points \( U_i \in \tau_k \) and \( U_j \in \tau_l \) **do**
   9. **if** \( \tau_k = \tau_l \) **then**
   10. Set \( d_{ij}^{\triangle} \) = the Euclidean distance between \( U_i \) and \( U_j \)
   11. **else** if \( \tau_k \neq \tau_l \) **then**
   12. (i) Calculate the Euclidean distance between the point and the corresponding centroid, \( v_{ik} \) and \( v_{jl} \)
   13. (ii) Set \( d_{ij}^{\triangle} = c_{kl} + v_{ik} + v_{jl} \)
   14. The approximate geo-desic distance between \( U_i \) and \( U_j \) is \( d_{ij} = \min \{d^0_{ij}, d_{ij}^{\triangle}\} \)

of constant effect is correctly specified, the bias in the estimation of these components is eliminated and root-\( n \) convergence rates can be obtained for the constant coefficients.

To separate the constant and varying coefficients, we propose a backward selection type of algorithm based on \( \text{BIC} = -2\log(\ell_n) + m \log(n) \), where \( m \) is the edf as in (15). In Algorithm 3, we present our model selection procedure using BIC.

Algorithm 3: BPST-BIC

**Data:** Observations \( \{U_i, X_i, Y_i\}_{i=1}^n \) and \( Y \)

**Output:** Index sets \( M^v \) and \( M^c \) which indicates the index of the explanatory variables with varying and constant effect, respectively

1. Initialize: \( M^v = \{1, \ldots, p\}; M^c = \emptyset; \text{BIC}_{\text{old}} = \text{BIC}_{\text{new}} = \text{BIC}(M^v, M^c) \)
2. **while** \( \text{BIC}_{\text{new}} \leq \text{BIC}_{\text{old}} \land |M^v| > 0 \) **do**
   3. (i) set \( \text{BIC}_{\text{old}} = \text{BIC}_{\text{new}} \)
   4. (ii) let \( k^* = \arg \min_{k \in M^v} \text{BIC}(M^v \backslash \{k\}, M^c \cup \{k\}) \)
   5. (iii) set \( M^v = M^v \backslash \{k^*\}; M^c = M^c \cup \{k^*\} \)
   6. (iv) calculate \( \text{BIC}_{\text{new}} = \text{BIC}(M^v, M^c) \)
5. Simulation studies

In this section, we investigate the numerical performance of the proposed model selection and estimation method. R code of the simulation studies presented in this section can be accessed from Github: https://github.com/funstatpackages/SARVCM. We conduct two simulation studies based on two types of spatial domains: (1) a regular rectangular domain, and (2) an irregular horseshoe domain.

In both simulation studies, for any two different locations $i$ and $j$, we consider the exponential weights $w_{ij} = \exp(-10d_{ij})/\sum_{k \neq i} \exp(-10d_{ik})$, where $d_{ij}$ is the Euclidean distance for the rectangular domain in Simulation Study 1, and $d_{ij}$ is the geodesic distance for the horseshoe domain with a complex boundary in Simulation Study 2. In both examples, when implementing the proposed BPST method, we use Algorithm 1 to approximate the true distance. For the kernel method proposed in Sun et al. (2014), the Euclidean distance is used.

5.1. Simulation study 1

Note that the Kernel method proposed in Sun et al. (2014) only works for data distributed on rectangular domains, in this example, we consider a regular squared domain, $\Omega = [0, 1]^2$. Similar to Sun et al. (2014), we generate data from the following SAR-PLVCM:

$$Y_i = \alpha_0 \sum_{j \neq i} w_{ij} Y_j + \sum_{l=1}^2 Z_i \gamma_{0l} + \sum_{k=1}^3 X_{ik} \beta_{0k} (U_i) + \epsilon_i, \quad i = 1, \ldots, n. \quad (16)$$

Following Sun et al. (2014), we set $\alpha_0 = 0.5$, $\beta_{01}(u) = \sin(\|u\|^2 \pi)$, $\beta_{02}(u) = \cos(\|u\|^2 \pi)$ and $\beta_{03}(u) = \exp(\|u\|^2)$. Moreover, we add two constant components into model (16) with actual coefficients $\gamma_{01} = -1$ and $\gamma_{02} = 1$. The first row in Figure S.2 in the Supplementary Material shows the contour plot for the true coefficient functions: $\beta_{01}$, $\beta_{02}$ and $\beta_{03}$. Both the explanatory variables, $X_i$’s and $Z_i$’s, and the random noise $\epsilon_i$’s are generated independently from the standard normal distribution. The sample size $n$ takes values 500 and 1000 in each Monte Carlo experiment with 200 replications. In this example, we consider two different triangulations as shown in the first column of Figure S.2 in the Supplementary Material. The first triangulation, $\triangle_1$, contains 8 triangles and 9 vertices, and $\triangle_2$ contains 18 triangles and 16 vertices.

We first investigate the performance of the BST and BPST methods when the true model structure is known, referred to as BST-ORACLE and BPST-ORACLE. To examine the accuracy of the proposed estimation procedure, we calculate the mean squared error (MSE) for the estimators of the constant coefficients and the estimator of the neighborhood effect parameter. We also calculate the mean integrated squared error (MISE) for the functional coefficient estimators. Table 1 summarizes the average MSEs and MISEs based on 200 replications. Figures S.2 and S.3 in the supplemental materials show the estimated functions via different methods for a typical simulation run based on sample size $n = 500$ and 1000, respectively. From Table 1 and Figures S.2 and S.3, it is easy to see that the estimation accuracy improves for both the BST and BPST methods as $n$ increases. From Table 1, we also observe that, for the BPST method, two different triangulations yield similar results, while for the BST method, when the sample size is moderate, a fine triangulation is usually not
Table 1. Mean integrated squared error (MISE) of varying coefficient estimators, mean squared error (MSE) of constant coefficient estimators and average computing time (seconds per replication) in Simulation 1.

| n   | Method                  | $\hat{\beta}_1$ | $\hat{\beta}_2$ | $\hat{\beta}_3$ | $\hat{\gamma}_1$ | $\hat{\gamma}_2$ | $\hat{\alpha}$ | Time (s) |
|-----|-------------------------|------------------|------------------|------------------|------------------|------------------|----------------|----------|
| 500 | BST-ORACLE($\triangle_1$) | 0.0323           | 0.0358           | 0.0293           | 0.0024           | 0.0022           | 0.0072         | 0.3      |
|     | BPST-ORACLE($\triangle_1$) | 0.0313           | 0.0303           | 0.0243           | 0.0025           | 0.0022           | 0.0075         | 0.3      |
|     | BST-ORACLE($\triangle_2$)  | 0.0578           | 0.0526           | 0.0487           | 0.0025           | 0.0022           | 0.0073         | 0.3      |
|     | BPST-ORACLE($\triangle_2$) | 0.0340           | 0.0281           | 0.0254           | 0.0024           | 0.0022           | 0.0087         | 0.3      |
|     | BST-BIC($\triangle_1$)   | 0.0323           | 0.0358           | 0.0293           | 0.0024           | 0.0022           | 0.0072         | 2.6      |
|     | BPST-BIC($\triangle_1$)  | 0.0313           | 0.0303           | 0.0243           | 0.0025           | 0.0022           | 0.0075         | 2.8      |
|     | BST-BIC($\triangle_2$)   | 0.0578           | 0.0526           | 0.0487           | 0.0025           | 0.0022           | 0.0073         | 3.2      |
|     | BPST-BIC($\triangle_2$)  | 0.0337           | 0.0280           | 0.0249           | 0.0027           | 0.0025           | 0.0083         | 3.2      |
|     | Kernel-BIC($0.5h_{cov}, h_{cov}$) | 0.0510       | 0.0493           | 0.0722           | 0.0139           | 0.0132           | 0.0115         | 922.8    |
|     | Kernel-BIC($h_{cov}, h_{cov}$) | 0.0510           | 0.0493           | 0.0722           | 0.0117           | 0.0112           | 0.0115         | 1019.2   |
|     | Kernel-BIC($2h_{cov}, h_{cov}$) | 0.0510           | 0.0493           | 0.0722           | 0.0041           | 0.0037           | 0.0115         | 1319.5   |
|     | Kernel-BIC($4h_{cov}, h_{cov}$) | 0.0775           | 0.0493           | 0.0722           | 0.0031           | 0.0032           | 0.0115         | 1332.8   |
| 1000| BST-ORACLE($\triangle_1$) | 0.0165           | 0.0190           | 0.0142           | 0.0009           | 0.0010           | 0.0057         | 0.9      |
|     | BPST-ORACLE($\triangle_1$) | 0.0168           | 0.0177           | 0.0128           | 0.0009           | 0.0010           | 0.0058         | 0.9      |
|     | BST-ORACLE($\triangle_2$)  | 0.0228           | 0.0205           | 0.0207           | 0.0009           | 0.0010           | 0.0059         | 0.9      |
|     | BPST-ORACLE($\triangle_2$) | 0.0197           | 0.0150           | 0.0135           | 0.0009           | 0.0010           | 0.0060         | 1.0      |
|     | BST-BIC($\triangle_1$)   | 0.0165           | 0.0190           | 0.0142           | 0.0009           | 0.0010           | 0.0057         | 10.0     |
|     | BPST-BIC($\triangle_1$)  | 0.0168           | 0.0177           | 0.0128           | 0.0009           | 0.0010           | 0.0060         | 10.0     |
|     | BST-BIC($\triangle_2$)   | 0.0228           | 0.0205           | 0.0207           | 0.0009           | 0.0010           | 0.0059         | 11.1     |
|     | BPST-BIC($\triangle_2$)  | 0.0197           | 0.0150           | 0.0135           | 0.0009           | 0.0010           | 0.0062         | 11.2     |
|     | Kernel-BIC($0.5h_{cov}, h_{cov}$) | 0.0332           | 0.0277           | 0.0455           | 0.0090           | 0.0091           | 0.0085         | 3932.3   |
|     | Kernel-BIC($h_{cov}, h_{cov}$) | 0.0322           | 0.0277           | 0.0455           | 0.0074           | 0.0084           | 0.0085         | 5508.4   |
|     | Kernel-BIC($2h_{cov}, h_{cov}$) | 0.0322           | 0.0277           | 0.0455           | 0.0016           | 0.0017           | 0.0085         | 6167.2   |
|     | Kernel-BIC($4h_{cov}, h_{cov}$) | 0.0354           | 0.0277           | 0.0452           | 0.0016           | 0.0017           | 0.0085         | 6761.6   |

Note: BST-ORACLE($\triangle_i$): bivariate spline estimator based on triangulation $\triangle_i$ when the true model structure is known; BPST-ORACLE($\triangle_i$): bivariate penalized spline estimator based on triangulation $\triangle_i$ when the true model structure is known; BST-BIC($\triangle_i$): bivariate spline estimator based on triangulation $\triangle_i$ when the proposed BIC method is used for model selection; BPST-BIC($\triangle_i$): bivariate penalized spline estimator based on triangulation $\triangle_i$ when the proposed BIC method is used for model selection; Kernel-BIC($c^*h_{cov}, h_{cov}$): the BIC local smoothing method (Sun et al. 2014) with bandwidth $c^*h_{cov}$ for model selection and bandwidth $h_{cov}$ for model refitting after selection.

suggested. Overall, the BPST is more convenient for model fitting than the BST since it is less sensitive to the triangulations used in the estimation.

Next, we investigate the performance of the proposed model identification method, abbreviated as BST-BIC and BPST-BIC, in terms of model selection and estimation accuracy. We start with the spatial autoregressive model with all varying coefficients and then implement our proposed Algorithm 3 to choose the model structure. Note that the true constant set is $\mathcal{M}^c = \{1, 2\}$ and true varying set is $\mathcal{M}^v = \{3, 4, 5\}$. For comparison, we carry out the local smoothing method with model selection based on the BIC proposed in Sun et al. (2014) (Kernel-BIC).

For the BST-BIC and BPST-BIC methods, we still consider two triangulations $\triangle_1$ and $\triangle_2$ as described above. For the Kernel-BIC method, if the component is identified to be a constant, Sun et al. (2014) used the average of the estimated coefficient function of all sampled locations to estimate the constant coefficient. It is well known that the bandwidth of the kernel exhibits a strong influence on the model selection and estimation. Since Sun et al. (2014) did not give explicit suggestion on the bandwidth selection in the model selection, to illustrate the effect of the bandwidth, we consider different bandwidths in the model.
Table 2. Proportion of $\hat{\mathcal{M}}^c$ (estimated constant set) selected among 200 replications using different methods in Simulation 1.

| Spline | Method       | $\emptyset$ | [1] | [2] | [1, 2] |
|--------|--------------|-------------|-----|-----|--------|
| n = 500| BST-BIC($\triangle_1$) | 0.000 | 0.000 | 0.000 | 1.000 |
|        | BPST-BIC($\triangle_1$)  | 0.000 | 0.000 | 0.000 | 1.000 |
|        | BST-BIC($\triangle_2$)  | 0.000 | 0.000 | 0.000 | 1.000 |
|        | BPST-BIC($\triangle_2$)  | 0.010 | 0.005 | 0.000 | 0.985 |
| n = 1000| BST-BIC($\triangle_1$) | 0.000 | 0.000 | 0.000 | 1.000 |
|        | BPST-BIC($\triangle_1$)  | 0.000 | 0.000 | 0.000 | 1.000 |
|        | BST-BIC($\triangle_2$)  | 0.000 | 0.000 | 0.000 | 1.000 |
|        | BPST-BIC($\triangle_2$)  | 0.000 | 0.000 | 0.000 | 1.000 |

| Kernel | Method       | $\emptyset$ | [1] | [2] | [1, 2] | [1, 2, 3] | [1, 3] |
|--------|--------------|-------------|-----|-----|--------|--------|-----|
|        | Kernel-BIC(0.5$h_{cov}$) | 0.000 | 0.500 | 0.450 | 0.050 | 0.000 | 0.000 |
|        | Kernel-BIC($h_{cov}$) | 0.000 | 0.320 | 0.320 | 0.360 | 0.000 | 0.000 |
|        | Kernel-BIC(2$h_{cov}$) | 0.000 | 0.025 | 0.055 | 0.920 | 0.000 | 0.000 |
|        | Kernel-BIC(4$h_{cov}$) | 0.010 | 0.015 | 0.025 | 0.860 | 0.085 | 0.005 |

Note: BST-BIC($\triangle_j$): bivariate spline estimator based on triangulation $\triangle_j$ when the proposed BIC method is used for model selection; BPST-BIC($\triangle_j$): bivariate penalized spline estimator based on triangulation $\triangle_j$ when the proposed BIC method is used for model selection; Kernel-BIC($c_{cov}$): local smoothing method (Sun et al. 2014) with bandwidth $c_{cov}$ when BIC is used for model selection.

The advantage enjoyed by the BST and the BPST is even more pronounced in the model selection problem because the estimation procedure has to be done in many iterations to achieve convergence.

5.2. Simulation study 2

In this example, we consider a more complex domain, and perform the data analysis based on a modified horseshoe domain proposed by Wood, Bravington, and Hedley (2008).
In this simulation study, the lattice points are obtained by dividing the entire domain into $N = 80 \times 180 = 14,400$ evenly. In this example, we set $\alpha_0 = 0.5$. The first coefficient function $\beta_1(\cdot)$ is the same as in Wood (2003), and the second coefficient function $\beta_2(u_1, u_2) = 4 \sin((\pi/2)u_1u_2)$. The contour plots for true coefficient functions are shown in Figure 1. Similar as in Simulation Study 1, the linear coefficients are $\gamma_{01} = 1$ and $\gamma_{02} = -1$. The explanatory variables, $X_i$'s and $Z_i$'s, and random noises, $\epsilon_i$'s, are generated independently from the standard normal distribution. We perform the study with sample size $n = 500$, and 1000.

To approximate the complex horseshoe domain, we consider the BPST method based on three triangulations in this study. The triangulations used in this example are shown in the first column of Figure 1. In the first triangulation, there are 89 triangles and 73 vertices; $\triangle_2$ contains 165 triangles with 120 vertices, and $\triangle_3$ has 238 triangles and 161 vertices. For the Kernel-BIC, as described in Simulation Study 1, the bandwidth used in model selection is adjusted by multiplying a constant $c$ ($c = 0.5, 1.0, 2.0, 4.0$), then the model is refitted using bandwidth $h_{\text{cov}}$.

For each simulation setting, we first calculate the average MSEs and MISEs for the BPST-ORACLE method across 200 replications when the true model structure is known; see Table 3. It also summarizes the MSEs and MISEs for the BPST-BIC and Kernel-BIC methods, which shows that the proposed estimates are far more accurate than the local smoothing estimates.

Next, we compare the performance of the proposed BPST-BIC method with the Kernel-BIC method (Sun et al. 2014). Note that the true constant set is $M^c = \{1, 2\}$ and the true varying set is $M^v = \{3, 4\}$. Table 4 lists the proportion of $\hat{M}^c$ selected. From Table 4, one sees that the BPST-BIC far outperform the Kernel-BIC in identifying the correct model structure. Similar to what we find in Simulation Example 1, the Kernel-BIC is very sensitive to the choice of the bandwidth, and it fails to identify the correct model for some bandwidths. In contrast, the BPST-BIC can consistently identify the right model regardless of the choice of the triangulation.

The estimated functions under two methods for $n = 1000$ can be visualized in Figure 1. The performance of the Kernel-BIC method is impaired over the complex domain while the results of BPST and BPST-BIC method are stable. For the sake of space saving, Figures S.4 for case $n = 500$ are illustrated in the Supplementary Material.

### 6. Application

We apply the proposed method to Sydney real estate data analysis. In this study, we are interested in examining how some economics and social factors affect housing prices in Sydney. The dataset can be obtained from the R package HRW (Harezlak, Ruppert, and Wand 2018), and it contains 37,676 properties sold in the Sydney Statistical Division (an official geographical region including Sydney) in the calendar year of 2001. We focus on the winter quarter only to avoid the temporal issue, and there are 7291 properties. Based on the values of housing price, we classify the observations in the dataset into five different groups: (1) less than 250K, (2) 250–500K, (3) 500–750K, (4) 750–1000K, and (6) greater than 1000K. These groups are displayed in Figure 2(a).

The factors we consider including lot size (LS), average weekly income (Income), distance from house to main road in kilometers (DR), levels of particulate matter with a
Figure 1. Contour plots for the estimated coefficient functions in Simulation Study 2 based on \( n = 1000 \). BPST(\( \Delta_j \)): bivariate penalized spline estimator based on triangulation \( \Delta_j \), \( j = 1, 2, 3 \); Kernel(\( h_{cov} \)): local smoothing method (Sun et al. 2014) with bandwidth \( h_{cov} \).
Table 3. Mean integrated squared error (MISE) of varying coefficient estimators, mean squared error (MSE) of constant coefficient estimators and average computing time (seconds per replication) in Simulation 2.

| n   | Method                  | $\hat{\beta}_1$ | $\hat{\beta}_2$ | $\hat{\gamma}_1$ | $\hat{\gamma}_2$ | $\hat{\alpha}$ | Time (s) |
|-----|-------------------------|-------------------|-------------------|-------------------|-------------------|-----------------|--------|
| 500 | BPST-ORACLE($\Lambda_1$) | 0.0444            | 0.0347            | 0.0021            | 0.0019            | 0.0007          | 1.7    |
|     | BPST-ORACLE($\Lambda_2$) | 0.0499            | 0.0350            | 0.0022            | 0.0019            | 0.0008          | 4.1    |
|     | BPST-ORACLE($\Lambda_3$) | 0.0441            | 0.0352            | 0.0021            | 0.0019            | 0.0009          | 7.9    |
|     | BPST-BIC($\Lambda_1$)   | 0.0450            | 0.0496            | 0.0025            | 0.0030            | 0.0007          | 15.3   |
|     | BPST-BIC($\Lambda_2$)   | 0.0530            | 0.0377            | 0.0028            | 0.0029            | 0.0007          | 41.8   |
|     | BPST-BIC($\Lambda_3$)   | 0.0529            | 0.0556            | 0.0030            | 0.0040            | 0.0008          | 82.3   |
|     | Kernel-BIC(0.5$h_{cov\text{, }h_{cov}}$) | 0.1770            | 0.1089            | 0.0517            | 0.0579            | 0.0141          | 856.2  |
|     | Kernel-BIC($h_{cov\text{, }h_{cov}}$) | 0.1770            | 0.1089            | 0.0479            | 0.0538            | 0.0141          | 890.2  |
|     | Kernel-BIC(2$h_{cov\text{, }h_{cov}}$) | 0.1770            | 0.1089            | 0.0125            | 0.0140            | 0.0141          | 1014.0 |
|     | Kernel-BIC(4$h_{cov\text{, }h_{cov}}$) | 0.1770            | 0.1089            | 0.0072            | 0.0078            | 0.0141          | 999.8  |
| 1000| BPST-ORACLE($\Lambda_1$) | 0.0277            | 0.0194            | 0.0010            | 0.0009            | 0.0005          | 3.8    |
|     | BPST-ORACLE($\Lambda_2$) | 0.0326            | 0.0199            | 0.0010            | 0.0009            | 0.0006          | 9.2    |
|     | BPST-ORACLE($\Lambda_3$) | 0.0275            | 0.0192            | 0.0010            | 0.0009            | 0.0006          | 14.8   |
|     | BPST-BIC($\Lambda_1$)   | 0.0276            | 0.0193            | 0.0011            | 0.0012            | 0.0005          | 41.5   |
|     | BPST-BIC($\Lambda_2$)   | 0.0326            | 0.0197            | 0.0012            | 0.0012            | 0.0006          | 99.3   |
|     | BPST-BIC($\Lambda_3$)   | 0.0305            | 0.0216            | 0.0010            | 0.0010            | 0.0005          | 163.2  |
|     | Kernel-BIC(0.5$h_{cov\text{, }h_{cov}}$) | 0.0885            | 0.0633            | 0.0242            | 0.0412            | 0.0335          | 3475.4 |
|     | Kernel-BIC($h_{cov\text{, }h_{cov}}$) | 0.0885            | 0.0633            | 0.0249            | 0.0357            | 0.0335          | 3623.7 |
|     | Kernel-BIC(2$h_{cov\text{, }h_{cov}}$) | 0.0885            | 0.0633            | 0.0039            | 0.0058            | 0.0335          | 4161.6 |
|     | Kernel-BIC(4$h_{cov\text{, }h_{cov}}$) | 0.0885            | 0.0633            | 0.0020            | 0.0021            | 0.0335          | 4353.8 |

Note: BPST-ORACLE($\Lambda_j$): bivariate spline estimator based on triangulation $\Lambda_j$ when the true model structure is known; BPST-BIC($\Lambda_j$): bivariate penalized spline estimator based on triangulation $\Lambda_j$ when the true model structure is known; BST-BIC($\Lambda_j$): bivariate spline estimator based on triangulation $\Lambda_j$ when the proposed BIC method is used for model selection; Kernel-BIC($c*h_{cov\text{, }h_{cov}}$): the BIC local smoothing method (Sun et al. 2014) with bandwidth $c*h_{cov}$ for model selection and bandwidth $h_{cov}$ for model refitting after selection.

As shown in Figure 2(b), we use a triangulation with 197 triangle and 172 vertices for the bivariate spline smoothing. For comparison, we also conduct the Kernel-BIC, the linear regression model (LM), and the SAR-LM. Unfortunately, due to the size of the dataset and huge computational cost, the Kernel-BIC cannot be implemented using regular PCs. The estimated housing price using BPST-BIC, LM and SAR-LM are plotted in Figure 2(c–e). All of these plots indicate that the inland housing prices are obviously lower than the coastal regions. We observe that the LM significantly overestimate the inland house values while it underestimates the coastal housing prices. The SAR-LM method has a similar estimation issue when using the Euclidean distance. In contrast, SAR-PLVCM is able to produce much
Table 4. Proportion of $\hat{M}^c$ (estimated constant set) selected among 200 replications using different methods in Simulation 2.

| Method          | $\emptyset$ | [1] | [2] | [4] | [1,2] | [1,4] |
|-----------------|-------------|-----|-----|-----|-------|-------|
| $n = 500$       |             |     |     |     |       |       |
| BPST-BIC($\Delta_1$) | 0.015       | 0.010 | 0.005 | 0.005 | 0.965  | 0.000 |
| BPST-BIC($\Delta_2$) | 0.020       | 0.010 | 0.015 | 0.000 | 0.955  | 0.000 |
| BPST-BIC($\Delta_3$) | 0.030       | 0.025 | 0.015 | 0.000 | 0.925  | 0.005 |
| $n = 1000$      |             |     |     |     |       |       |
| BPST-BIC($\Delta_1$) | 0.010       | 0.005 | 0.000 | 0.000 | 0.985  | 0.000 |
| BPST-BIC($\Delta_2$) | 0.010       | 0.005 | 0.005 | 0.000 | 0.980  | 0.000 |
| BPST-BIC($\Delta_3$) | 0.005       | 0.005 | 0.000 | 0.000 | 0.990  | 0.000 |

| Method          | $\emptyset$ | [1] | [2] | [1,2] |
|-----------------|-------------|-----|-----|-------|
| Kernel-BIC(0.5$h_{cov}$) | 0.000       | 0.545 | 0.455 | 0.000 |
| Kernel-BIC($h_{cov}$) | 0.000       | 0.510 | 0.390 | 0.100 |
| Kernel-BIC(2$h_{cov}$) | 0.000       | 0.065 | 0.085 | 0.850 |
| Kernel-BIC(4$h_{cov}$) | 0.000       | 0.045 | 0.025 | 0.930 |

Note: BST-BIC($\Delta_j$): bivariate spline estimator based on triangulation $\Delta_j$ when the proposed BIC method is used for model selection; BPST-BIC($\Delta_j$): bivariate penalized spline estimator based on triangulation $\Delta_j$ when the proposed BIC method is used for model selection; Kernel-BIC($c h_{cov}$): local smoothing method (Sun et al. 2014) with bandwidth $c = h_{cov}$ when BIC is used for model selection.

Figure 2. Scatter plot of observed and estimated housing prices for Sydney housing dataset. (a) observed price. (b) Triangulation. (c) BPST estimated price. (d) OLS estimated price and (e) SAR-LM estimated price.

Figure 3 summarizes the coefficient estimation results via the proposed method. As shown in Figure 3(a), compared with the northwest region, lot size has a higher positive effect on the housing price in the southeast region. Income is generally positively associated with the housing price as shown in Figure 3(b). The impact of distance to coastline on the housing price is negative especially for the houses located on the east coast.

To evaluate different methods, we adopt the 10-fold cross-validation to calculate the out-of-sample prediction errors. The estimation and prediction results are listed in Table 5.
Figure 3. Coefficient maps of the fitted SAR-PLVCM for Sydney housing dataset. (a) lotsize. (b) income. (c) dist to main road. (d) PM 10 and (e) dist to coastline.

Table 5. Mean squared error (MSE) and 10-fold cross-validation (CV) mean squared prediction error (MSPE) of the housing prices using different methods.

| Method     | MSE  | MSPE | \(\widehat{\alpha}\) |
|------------|------|------|-----------------------|
| BPST-BIC   | 0.0723 | 0.0773 | 0.23                  |
| Kernel-BIC | –     | –    | –                     |
| LM         | 0.1171 | 0.1173 | –                     |
| SAR-LM     | 0.1467 | 0.1461 | 0.87                  |

Note: BPST-BIC: bivariate penalized spline estimator over triangulation when the proposed BIC method is used for model selection; Kernel-BIC: local smoothing method (Sun et al. 2014) when BIC is used for model selection; LM: linear regression model with all constant coefficients; SAR-LM: spatial autoregressive model with all constant coefficients.

From this table, it is obvious that BPST-BIC provides the most accurate result among the three methods. A histogram and a scatter plot of the residuals of log-transformed housing prices via BPST-BIC are showed in Figure 4(a,b), respectively.

7. Conclusion and discussion

We built a semiparametric spatial model for for data distributed over complex domains accounting for spatial nonstationarity and spatial dependence through a spatially varying coefficient autoregressive model. We propose approximating the varying coefficient functions via bivariate splines over a triangulation. This type of approximations avoids
the problem of ‘leakage’ across the complex domains. To facilitate the calculation of the weight matrix in the model, large spatial datasets, we developed two efficient algorithms to approximate the geodesic distance over a complex domain based on the triangulation technique. Our empirical results confirmed that the distance estimated using our method is really close to the geodesic distance. For the model selection, we proposed a BIC method which directly identifies predictors with constant and varying effects. Monte Carlo simulations showed that our model selection approach performs better than some of the existing methods in finite samples.

A few more issues still merit further research. For instance, for the model selection procedure, one could also consider some regularization methods, such as group lasso. It is interesting to study how the regularization methods work for spatial regression models. It is also interesting to consider a unified approach to perform estimation, variable selection and model structure identification simultaneously for the SAR-PLVCMs when the number of predictors is large. In this paper we addressed only the spatial data. The wide availability of data observed over time and space due to widespread collection of network and inexpensive geographical information systems, has stimulated many studies in a variety of disciplines. It would be interesting to extend our approach to spatiotemporal problems.

Disclosure statement
No potential conflict of interest was reported by the author(s).

Funding
Li Wang’s research was partially supported by the National Science Foundation grants DMS-1542332 and DMS-1916204; Division of Mathematical Sciences.

ORCID
Li Wang  http://orcid.org/0000-0001-8432-9986
References

Brunsdon, C., Fotheringham, A.S., and Charlton, M.E. (1996), ‘Geographically Weighted Regression: A Method for Exploring Spatial Nonstationarity’, *Geographical Analysis*, 28, 281–298.

Fan, J., and Huang, T. (2005), ‘Profile Likelihood Inferences on Semiparametric Varying-coefficient Partially Linear Models’, *Bernoulli*, 11, 1031–1057.

Fotheringham, A., Brunsdon, C., and Charlton, M. (2002), *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*, Hoboken: John Wiley & Sons.

Gelfand, E.A., Kim, H.J., Sirmans, C.F., and Banerjee, S. (2003), ‘Spatial Modeling With Spatially Varying Coefficient Processes’, *Journal of the American Statistical Association*, 98, 387–396.

Harezlak, J., Ruppert, D., and Wand, M. (2018), *Semiparametric Regression with R*, New York: Springer.

Hastie, T., and Tibshirani, R. (1993), ‘Varying-Coefficient Models’, *Journal of the Royal Statistical Society. Series B*, 55, 757–796.

Huang, J.Z., Wu, C.O., and Zhou, L. (2002), ‘Varying-coefficient Models and Basis Function Approximations for the Analysis of Repeated Measurements’, *Biometrika*, 89, 111–128.

Kelejian, H.H., and Prucha, I.R. (2001), ‘On the Asymptotic Distribution of the Moran I Test Statistic with Applications’, *Journal of Econometrics*, 104, 219–257.

Lai, M.J., and Schumaker, L.L. (2007), *Bivariate Penalized Splines for Regression*, *Statistica Sinica*, 23, 1399–1417.

Lee, L.F. (2004), ‘Asymptotic Distributions of Quasi-maximum Likelihood Estimators for Spatial Autoregressive Models’, *Econometrica*, 72, 1899–1925.

Miller, D.L., and Wood, S.N. (2014), ‘Finite Area Smoothing with Generalized Distance Splines’, *Econometrica*, 21, 715–731.

Mu, J., Wang, G., and Wang, L. (2018), ‘Estimation and Inference in Spatially Varying Coefficient Models’, *Environmetrics*, 29, e2485.

Qingguo, T. (2013), ‘B-spline Estimation for Semiparametric Varying-coefficient Partially Linear Regression with Spatial Data’, *Journal of Nonparametric Statistics*, 25, 361–378.

Stone, C.J. (1982), ‘Optimal Global Rates of Convergence for Nonparametric Regression’, *The Annals of Statistics*, 10, 1040–1053.

Sun, Y., Yan, H., Zhang, W., and Lu, Z. (2014), ‘A Semiparametric Spatial Dynamic Model’, *The Annals of Statistics*, 42, 700–727.

Wang, H., and Ranalli, M.G. (2007), ‘Low-rank Smoothing Splines on Complicated Domains’, *Biometrics*, 63, 209–217.

Yang, L., Park, B.U., Xue, L., and Härdle, W. (2006), ‘Estimation and Testing for Varying Coefficients in Additive Models With Marginal Integration’, *Journal of the American Statistical Association*, 101, 1212–1227.

Yu, S., Wang, G., Wang, L., Liu, C., and Yang, L. (2019), ‘Estimation and Inference for Generalized Geoadditive Models’, *Journal of the American Statistical Association*, 1–27.
Appendix

A.1 Regularity assumptions

To derive the asymptotic properties of the proposed estimators, we introduce some notation of inner products and norms. For any function $f$ over the closure of domain $\Omega$, let $E_n(f) = n^{-1} \sum_{i=1}^{n} f(U_i)$ and $E(f) = E(f(U))$. Define the empirical inner product and norm as $(f_1, f_2)_n^\Omega = E_n(f_1 f_2)$ and $(f_1, f_2)_n^\Omega$ for measurable functions $f_1$ and $f_2$ on $\Omega$. The theoretical $L_2$ inner product and the induced norm are given by $(f_1, f_2)_L^2 = E(f_1 f_2)$ and $\|f\|_2^2 = (f, f)_L^2$.

Denote $\|f\|_{\infty, \Omega} = \sup_{u \in \Omega} |f(u)|$ as the supremum norm of function $f$, and denote $\|f\|_{u, \infty, \Omega} = \max_{i=1}^{n} |D_i^n D_j^n f(x)|_{\infty, \Omega}$ as the maximum norms of all the $v$th order derivatives of $f$ over $\Omega$.

Let $W^{l,\infty}(\Omega) = \{ f : |f|_{k,\infty, \Omega} < \infty, 0 \leq k \leq \ell \}$ be the standard Sobolev space. Given random variables $T_n$ for $n \geq 1$, we write $T_n = O_P(b_n)$ if $\lim_{n \to \infty} \sup P(|T_n| \geq c b_n) = 0$, and $T_n = o_P(b_n)$ if $\lim_{n \to \infty} P(|T_n| \geq c b_n) = 0$, for any constant $c > 0$. Also, we write $a_n \asymp b_n$ if there exist two positive constants $c_1, c_2$ such that $c_1 |a_n| \leq |b_n| \leq c_2 |a_n|$, for all $n \geq 1$.

We make the following assumptions to establish the results.

(A1) The bivariate functions $g_{ij}^x(\cdot)$ used in (10), $l = 1, \ldots, p_1$, $j = 1, \ldots, p_2$; and the true functions $\beta_{ik}(\cdot), k = 1, \ldots, p_2$, $\in W^{l+1,\infty}(\Omega)$ for an integer $l \geq 1$.

(A2) The random variables $Z_{il}$ and $X_{ik}$ are bounded, uniformly on $i = 1, \ldots, n$, $l = 1, \ldots, p_1$, and $k = 1, \ldots, p_2$, $(Z_{i1} X_{i1})^T, \ldots, (Z_{in} X_{in})^T$ is an i.i.d. random sample and is independent of $(e_1, \ldots, e_n)$. The eigenvalues $\lambda_j(u) \leq \cdots \leq \lambda_{p_1+p_2}(u)$ of $\Sigma(u)$ in (11) are bounded away from 0 and infinity; that is, there are positive constants $C_1$ and $C_2$ such that $C_1 \leq \lambda_1(u) \leq \cdots \leq \lambda_{p_1+p_2}(u) \leq C_2$ for all $u \in \Omega$.

(A3) There exists a constant $C$ such that $E(\epsilon^{2a}) \leq C < \infty$ for some $a > 2$.

(A4) The density function $f(u)$ of $U_i$ is bounded away from zero and infinity on $\Omega$.

(A5) The triangulation $|\triangle|$ is $\pi$-quasi-uniform, that is, there exists a positive constant $\pi$ such that $(\min_{e_1 \in \mathcal{R}_1} |\triangle|) \leq \pi$.

(A6) The number $K_n$ of the triangles and the sample size $n$ satisfy that $N = C n^\xi$ for some constant $C > 0$ and $1/(\ell + 2) < \xi < 1$.

(A7) $T^{-1}$ exists.

(A8) $W$ and $T^{-1}$ are uniformly bounded in both row and column sums.

(A9) $T(\alpha)^{-1}$ are uniformly bounded in either row or column sums, uniformly in $\alpha \in \mathcal{D}$, where $\mathcal{D}$ is a compact parameter space and $\alpha_0$ is an interior point of $\mathcal{D}$.

(A10) Elements $w_{ij}$ of $W$ are at most of $O(v_n^{-1})$, uniformly in all $i, j$, where $\lim_{n \to \infty} v_n/n = 0$.

(A11) $n^{-1} (Z, X, G_{\mu_0})^T (Z, X, G_{\mu_0})$ exists and is nonsingular, where $Z^T = (Z_1, \ldots, Z_n)$, $X^T = (X_1, \ldots, X_n)$.

(A12) $\lim_{n \to \infty} n^{-1} \mathbb{E}((G_{\mu_0})^T (I_n - P_G)G_{\mu_0}) \neq 0$.

(A12') $\lim_{n \to \infty} n^{-1} \mathbb{E}((G_{\mu_0})^T (I_n - P_G)G_{\mu_0}) = 0$.

(A13) The $\{v_n\}$ is a bounded sequence and for any $\alpha \neq \alpha_0$,

$$\lim_{n \to \infty} \left\{ n^{-1} \log \|\sigma_{\alpha}^2 (T^{-1})^T\| - \frac{1}{n} \log \|\sigma^{2}(\alpha) T^{-1}(\alpha) \{T^{-1}(\alpha)^T\}^T\right\} \neq 0.$$
pseudo-likelihood function of $\alpha$. If Assumption (A12') holds, then Assumption (A13) will ensure the uniqueness of maximizer.

### A.2 Proof of Theorems 3.1–3.3

For any fixed $\alpha$, denote

$$s_n(\alpha) = \max_{\gamma, \theta, \sigma^2} \mathbb{E}[\ell_n(\gamma, \theta, \sigma^2; \alpha)], \tag{A1}$$

where $\ell_n(\theta, \sigma^2; \alpha)$ is in (8). Solving the maximization problem in (A1) yields the solutions $(\hat{\gamma}(\alpha), \hat{\sigma}(\alpha)^\top) = (\Phi^\top \Phi)^{-1} \Phi^\top T(\alpha) T^{-1} \mu_0$, and $\hat{\sigma}^2(\alpha) = (1/n) \mathbb{E}[|T(\alpha) Y - P\Phi T(\alpha) T^{-1} \mu_0|^2]$, where

$$\hat{\sigma}^2(\alpha) = \frac{1}{n} \mathbb{E}[|T(\alpha) T^{-1} (\mu_0 + \epsilon) - P\Phi T(\alpha) T^{-1} \mu_0|^2]$$

$$= \frac{(\alpha_0 - \alpha)^2}{n} \left[ \mathbb{E} \left\{ \mathbb{G} \mu_0^\top (I_n - P\Phi) (\mathbb{G} \mu_0) \right\} \right] + \frac{\sigma_0^2}{n} \mathrm{tr} \left\{ (T^{-1} )^T T(\alpha)^T T(\alpha) T^{-1} \right\}$$

$$+ \frac{1}{n} \mathbb{E} \mu_0^\top (I_n - P\Phi) \mu_0 + \frac{2(\alpha_0 - \alpha)}{n} \mathbb{E} \mu_0^\top (I_n - P\Phi) \mathbb{G} \mu_0. \tag{A2}$$

Then, for $s_n(\alpha)$ in (A1), we have $s_n(\alpha) = -(n/2)[\log(2\pi) + 1] - (n/2) \log \hat{\sigma}^2(\alpha) + \log |T(\alpha)|$.

**Lemma A.1:** Under Assumptions (A1)–(A11), $n^{-1} s_n(\alpha)$ is uniformly equicontinuous on $[0, 1]$.

**Lemma A.2:** Under Assumptions (A1)–(A11), $\sup_{\alpha \in [0, 1]} n^{-1} |\ell_n(\alpha) - s_n(\alpha)| = o_P(1)$.

#### A.2.1 Proof of Theorem 3.1

**Proof:** Note that

$$\ell_n^*(\sigma^2, \alpha) = -\frac{n}{2} \log(\sigma^2) + \log |T(\alpha)| - \frac{1}{\sigma^2} \|T(\alpha) Y\|^2 \tag{A3}$$

is the log-likelihood function of a standard SAR model: $Y = \alpha W Y + \epsilon$, where $\epsilon \sim N(0, \sigma^2 I_n)$. Denote

$$\sigma^2(\alpha) = \frac{\sigma_0^2}{n} \mathrm{tr} \{ [T(\alpha) T^{-1}]^T T(\alpha) T^{-1} \}.$$

Define $s_n^*(\alpha) = \max_{\sigma^2} \mathbb{E}[\ell_n^*(\sigma^2, \alpha)]$, which can be represented as

$$s_n^*(\alpha) = -\frac{n}{2} [\log(2\pi) + 1] - \frac{n}{2} \log \sigma^2(\alpha) + \log |T(\alpha)|.$$

It is easy to see that, for any $\alpha \in D$,

$$\frac{1}{n} \{ s_n(\alpha) - s_n(\alpha_0) \} = \frac{1}{n} \left\{ s_n^*(\alpha) - s_n^*(\alpha_0) \right\} + \frac{1}{2} \left\{ \log \sigma^2(\alpha) - \log \hat{\sigma}^2(\alpha) - \log \sigma_0^2 \right\}$$

It is clear that

$$n^{-1} \left\{ s_n^*(\alpha) - s_n^*(\alpha_0) \right\} = n^{-1} \left[ \mathbb{E}_{(\sigma_0^2, \alpha_0)} [\ell_n^*(\sigma_0^2, \alpha)] - \max_{\sigma^2} \mathbb{E}_{(\sigma_0^2, \alpha_0)} [\ell_n^*(\sigma^2, \alpha)] \right] \leq 0,$$

and by Lemma S.10, $\log \hat{\sigma}^2(\alpha_0) - \sigma_0^2 = o_P(1)$. According to (S.24) in Section S.1 in the Supplementary Material, $\log \sigma^2(\alpha) - \log \hat{\sigma}^2(\alpha) \leq 0$, thus, $n^{-1} \{ s_n(\alpha) - s_n(\alpha_0) \} \leq 0$. 


We can prove the uniqueness of $\alpha_0$ by contradiction. If the uniqueness of $\alpha_0$ doesn’t hold, then exist $\delta > 0$, and a sequence $\{\alpha_n\} \in \overline{N}_\delta(\alpha_0)$, such that,

$$
\lim_{n \to \infty} n^{-1} [s_n(\alpha_n) - s_n(\alpha_0)] = 0, \quad \lim_{n \to \infty} \alpha_n = \tilde{\alpha} \neq \alpha_0, \quad (A4)
$$

where $\overline{N}_\delta(\alpha_0)$ is the open neighborhood of $\alpha_0$ with radius $\delta$, $\overline{N}_\delta(\alpha_0)$ is the closure of $\overline{N}_\delta(\alpha_0)$.

From Lemma A.2, we have shown that $\{n^{-1}s_n(\alpha)\}$ is uniform equicontinuous of $\alpha$. Thus,

$$
\lim_{n \to \infty} \frac{1}{n} |s_n(\alpha) - s_n(\alpha_0)| \leq \lim_{n \to \infty} \frac{1}{n} |s_n(\alpha_n) - s_n(\tilde{\alpha})| + \lim_{n \to \infty} \frac{1}{n} |s_n(\alpha) - s_n(\alpha_0)| = 0.
$$

Note that

$$
s_n(\alpha_n) - s_n(\alpha) = \frac{n}{2} [\log \hat{\sigma}^2(\alpha) - \log \sigma_0^2] + \log |T| - \log |T(\alpha)|
$$

$$
= \frac{n}{2} [\log \hat{\sigma}^2(\alpha) - \log \sigma_0^2] + \frac{n}{2} [\log \sigma^2(\alpha) - \log \sigma_0^2] + \log |T| - \log |T(\alpha)|.
$$

Note that we have shown in Lemma A.1 that $(n/2)[\log \sigma^2(\alpha) - \log \sigma_0^2] + \log |T| - \log |T(\alpha)| \geq 0$, for all $\alpha \in D$, and $\hat{\sigma}^2(\alpha) - \sigma^2(\alpha) \geq 0$, for all $\alpha \in D$. Then, we obtain that

$$
\lim_{n \to \infty} \frac{1}{n} [\log \hat{\sigma}^2(\alpha) - \log \sigma_0^2] = 0, \quad (A5)
$$

$$
\lim_{n \to \infty} \frac{1}{n} [\log \sigma^2(\alpha) - \log \sigma_0^2] + \log |T| - \log |T(\alpha)| = 0. \quad (A6)
$$

Thus, by (A5), we have $\lim_{n \to \infty} [\hat{\sigma}^2(\alpha) - \sigma^2(\alpha)] = 0$, which implies $\lim_{n \to \infty} \frac{1}{n} \| (I_n - P_\Phi)(I_n - (\alpha_0 - \hat{\alpha})) \|_2^2 = 0$. Therefore, $\lim_{n \to \infty} \frac{1}{n} (G \mu_0)^T (I_n - P_\Phi) G \mu_0 = 0$. Then, (A6) conflicts with Assumption (A13). So $\alpha_0$ is unique.

Recall $\hat{\sigma}(\alpha)$ from (S.25) in Section S.1 in the Supplementary Material. By Lemma S.10, $\lim_{n \to \infty} n^{-1} \mu_0^T (I_n - P_\Phi) \mu_0 = 0$. According to (S.27) and (S.29),

$$
\lim_{n \to \infty} E[\epsilon^T (T^{-1})^T T(\alpha)^T (I_n - P_\Phi) T(\alpha) T^{-1} \epsilon]
$$

$$
= \sigma_0^2 \lim_{\alpha \to \alpha_0} \text{tr}[(T^{-1})^T T(\alpha)^T (I_n - P_\Phi) T(\alpha) T^{-1}] = n \sigma_0^2.
$$

From the discussion in Lemmas A.1, A.2 and by Theorem 3.4 in White (1996), we obtain that $\hat{\alpha} = \arg \max_{\alpha \in D} \ell_n(\alpha) \to \alpha_0$ in probability. Therefore, $\hat{\sigma}(\alpha) \to \sigma_0^2$ in probability. ■

A.2.2 Proof of Theorem 3.2

To show the proof of the asymptotic results, we first introduce some notations. For $\mathcal{X}_B = (X_1 \otimes B^*(U_1), \ldots, X_n \otimes B^*(U_n))^\top$, let $P_{\mathcal{X}_B} = \mathcal{X}_B^\top (\mathcal{X}_B^\top \mathcal{X}_B)^{-1} \mathcal{X}_B$, and denote

$$
\Sigma_{11,n} = \frac{1}{n} E[\text{tr}(G^2)] + \frac{1}{n} E[\text{tr}(G^\top G)] + \frac{1}{n \sigma_0^2} E[\mu_0^T G^\top (I_n - P_{\mathcal{X}_B}) G \mu_0],
$$

$$
\Sigma_{22,n} = \frac{1}{2 \sigma_0^2}, \quad \Sigma_{33,n} = \frac{1}{n \sigma_0^2} E[Z^\top (I_n - P_{\mathcal{X}_B}) Z], \quad \Sigma_{23,n} = 0,
$$

$$
\Sigma_{12,n} = \Sigma_{21,n} = \frac{1}{n \sigma_0^2} E[\text{tr}(G)], \quad \Sigma_{13,n} = \Sigma_{31,n} = \frac{1}{n \sigma_0^2} E[Z^\top (I_n - P_{\mathcal{X}_B}) G \mu_0].
$$
Next, let $m_3$ and $m_4$ be the third and fourth moments of $\epsilon_i$, and denote

$$
\Omega_{11,n} = \frac{m_4 - \sigma_0^4}{n \sigma_0^3} E \left\{ \sum_{i=1}^{n} (G)^2 \right\} + \frac{2m_3}{n \sigma_0^5} \sum_{i=1}^{n} E \left\{ (G)_{ii} (I_n - P \chi_b) \right\},
$$

$$
\Omega_{22,n} = \frac{m_4 - 3 \sigma_0^4}{4n \sigma_0^8}, \quad \Omega_{33,n} = 0, \quad \Omega_{23,n} = \Omega_{32,n} = \frac{m_3}{2n \sigma_0^6} E \left\{ (I_n - P \chi_b) Z \right\},
$$

$$
\Omega_{12,n} = \Omega_{21,n} = \frac{1}{2n \sigma_0^6} E \left\{ (m_4 - 3 \sigma_0^4) \operatorname{tr}(G) + m_3 (I_n - P \chi_b) G \mu_0 \right\},
$$

$$
\Omega_{13,n} = \Omega_{31,n} = \frac{m_3}{2n \sigma_0^6} E \left\{ (I_n - P \chi_b) Z \right\}.
$$

where $(G)_{ii}$ represents the $(i, i)$th entry of matrix $G$, $A_i \cdot$ represents the $i$th row of matrix $A$.

Lemma A.3: For matrices $\Sigma_n$ and $\Omega_n$ defined in (12) and (13), under Assumptions (A1)–(A8), we have

$$
\Sigma_n = \begin{pmatrix} \Sigma_{11,n} & \Sigma_{12,n} & \Sigma_{13,n} \\ \Sigma_{21,n} & \Sigma_{22,n} & \Sigma_{23,n} \\ \Sigma_{31,n} & \Sigma_{32,n} & \Sigma_{33,n} \end{pmatrix}, \quad \Omega_n = \begin{pmatrix} \Omega_{11,n} & \Omega_{12,n} & \Omega_{13,n} \\ \Omega_{21,n} & \Omega_{22,n} & \Omega_{23,n} \\ \Omega_{31,n} & \Omega_{32,n} & \Omega_{33,n} \end{pmatrix}
$$

with probability approaching to one.

Proof: First of all, we calculate the first order derivatives of the profile log-likelihood function:

$$
\frac{\partial \ell_n(\alpha, \sigma^2, \gamma)}{\partial \alpha} = -\operatorname{tr} \left\{ T(\alpha)^{-1} W \right\} + \frac{1}{\sigma_0^2} \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \} (I_n - P \chi_b) W Y,
$$

$$
\frac{\partial \ell_n(\alpha, \sigma^2, \gamma)}{\partial \sigma^2} = -\frac{n}{2 \sigma^2} + \frac{1}{2 \sigma_0^2} \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \}^\top \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \},
$$

$$
\frac{\partial \ell_n(\alpha, \sigma^2, \gamma)}{\partial \gamma} = -\frac{1}{\sigma_0^2} \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \}^\top (I_n - P \chi_b) Z.
$$

Then, we obtain the second order derivatives of the profile log-likelihood function:

$$
\frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \alpha^2} = -\operatorname{tr}(G^2) - \frac{1}{\sigma_0^2} (W Y)^\top (I_n - P \chi_b) W Y,
$$

$$
\frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \sigma^2} = \frac{n}{2 \sigma^4} - \frac{1}{\sigma_0^4} \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \}^\top \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \},
$$

$$
\frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \gamma^2} = -\frac{1}{\sigma_0^2} Z^\top (I_n - P \chi_b) Z.
$$

In addition, we have

$$
\frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \alpha \partial \sigma^2} = -\frac{1}{\sigma_0^4} \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \}^\top (I_n - P \chi_b) W Y,
$$

$$
\frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \sigma^2 \partial \gamma} = -\frac{1}{\sigma_0^2} Z^\top (I_n - P \chi_b) W Y,
$$

$$
\frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \sigma^2 \partial \gamma} = -\frac{1}{\sigma_0^4} Z^\top (I_n - P \chi_b) \{ Y(\alpha) - Z \gamma - \chi_B \theta(\alpha, \gamma) \}.
$$

For $\psi = (\alpha, \sigma^2, \gamma)^\top$, $\psi_0 = (\alpha_0, \sigma_0^2, \gamma_0)^\top$, note that

$$
\Sigma_n = -E \left\{ \frac{1}{n} \frac{\partial^2 \ell_n(\psi)}{\partial \psi^\top \partial \psi} \bigg|_{\psi = \psi_0} \right\}.
$$
by (12). Under Assumptions (A1)–(A8), we have \( n^{-1}(\mu_0 - Z\gamma_0)^\top (I_n - P_{X\beta})(\mu_0 - Z\gamma_0) = O(|\Delta|^{2d+2}) \). Also, for any matrix \( A \), \( E(\epsilon^\top A\epsilon) = \sigma_0^2 \text{tr}(A) \) holds. Then, it is easy to show that

\[
E \left\{ -\frac{1}{n} \frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \alpha^2} \bigg| \psi = \psi_0 \right\} = \Sigma_{11,n}, \quad E \left\{ -\frac{1}{n} \frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \gamma^2} \bigg| \psi = \psi_0 \right\} = \Sigma_{22,n} + o(1),
\]

\[
E \left\{ -\frac{1}{n} \frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \gamma \partial \gamma'} \bigg| \psi = \psi_0 \right\} = \Sigma_{33,n}, \quad E \left\{ -\frac{1}{n} \frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \alpha \partial \gamma} \bigg| \psi = \psi_0 \right\} = \Sigma_{23,n} + o(1),
\]

\[
E \left\{ -\frac{1}{n} \frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \sigma^2 \partial \gamma} \bigg| \psi = \psi_0 \right\} = \Sigma_{12,n} + o(1), \quad E \left\{ -\frac{1}{n} \frac{\partial^2 \ell_n(\alpha, \sigma^2, \gamma)}{\partial \alpha \partial \sigma^2} \bigg| \psi = \psi_0 \right\} = \Sigma_{13,n}.
\]

By the definition of matrix \( \Omega_n \) in (13), we have

\[
\Omega_n = E \left\{ n^{-1} \left( \frac{\partial \ell_n(\psi)}{\partial \psi} \right) \left( \frac{\partial \ell_n(\psi)}{\partial \psi'} \right)^\top \bigg| \psi = \psi_0 \right\} - \Sigma_n.
\]

Therefore, (A7) is derived based on the calculation of

\[
E \left\{ n^{-1} \left( \frac{\partial \ell_n(\psi)}{\partial \psi} \right) \left( \frac{\partial \ell_n(\psi)}{\partial \psi'} \right)^\top \bigg| \psi = \psi_0 \right\}.
\]

\[\square\]

**Proof of Theorem 3.2**: For \( \beta_k(s), k = 1, \ldots, p_2 \), there exists \( \theta_k^* \) such that \( |\beta_k(s) - \psi(s)\top \theta_k^*|_\infty = O(|\Delta|^{d+1}) \). Denote that \( \theta^* = (\theta_{p_2}^\top, \ldots, \theta_1^\top)\top \). Notice that

\[
Y(\alpha_0) - Z\gamma_0 - \mathcal{X}'\theta(\alpha_0, \gamma_0) = (I_n - P_{X\beta})(\epsilon + \zeta),
\]

where \( \zeta = (\sum_{k=1}^{p_2} \mathcal{X}_{ik}\beta_k(S_i) - \mathcal{X}_{ik}'\psi(S_i)^\top \theta_k^*)^n_{k=1} = O(|\Delta|^{d+1}) \). Then the first-order derivatives of the log-likelihood function with respect to \( \alpha, \sigma^2 \) and \( \gamma \) evaluated at \( \psi_0 \) are dominated by \( -\text{tr}(G) - \sigma_0^{-2} \epsilon^\top (I_n - P_{X\beta})GZ'\gamma_0 + \sigma_0^{-2} \epsilon^\top (I_n - P_{X\beta})Ge, -0.5n\sigma_0^{-2} + 0.5\sigma_0^{-4} \epsilon^\top (I_n - P_{X\beta})\epsilon \) and \( -\sigma_0^{-2} \epsilon^\top (I_n - P_{X\beta})Z \), respectively. Then by Theorem 3.1 from Kelejian and Prucha (2001), we can set up the asymptotic normality of \( n^{-1/2} \frac{\partial \ell_n(\psi_0)}{\partial \psi} \) with mean zero and covariance matrix we calculated above.

\[\square\]

**A.2.3 Proof of Theorem 3.3**

**Proof**: Let \( \bar{\theta} = U_{22}\mathcal{X}'_{12}(I_n - P_Z)TY, \bar{\beta}_k(u) = B^*(u)^\top \bar{\theta}_k \), for \( k = 1, \ldots, p_2 \), by Lemma S.16, \( \|\bar{\beta} - \bar{\theta}\| = O_P(|\Delta|^{d+1} + 1/(\sqrt{n} |\Delta|)) \), \( \|\gamma_0 - \gamma^*\| = O(n^{-1/2}) \). Denote

\[
\hat{\theta} = U_{22}\mathcal{X}'_{12}(I_n - P_Z)TY, \quad \hat{\beta}_k(u) = B^*(u)^\top \bar{\theta}_k, \quad k = 1, \ldots, p_2.
\]

Then, by Lemmas S.10, S.11 and S.13,

\[
\|\hat{\theta} - \bar{\theta}\|^2 \asymp n^{-1} \|U_{22}\mathcal{X}'_{12}(I_n - P_Z)W^{-1}TY\|^2 \asymp n^{-3}K_2\|\mathcal{X}'_{12}(I_n - P_Z)G(\mu_0 + \epsilon)\|^2
\]

\[
\asymp n^{-3}K_2\|\mathcal{X}'_{12}(I_n - P_Z)G\mu_0\|^2 + n^{-3}K_2\|\mathcal{X}'_{12}(I_n - P_Z)Ge\|^2
\]

\[
\asymp n^{-1}K_n + n^{-2}K_n,
\]

and \( \|\hat{\beta} - \beta\| \leq |\Delta|\|\bar{\theta} - \bar{\theta}\| + \|\beta_0 - \bar{\theta}\| = O_P(|\Delta|^{d+1} + n^{-1/2}|\Delta|). \)

\[\square\]