Computational Statistics and Data Analysis 177 (2023) 107581

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

www.elsevier.com/locate/csda

Sparse spatially clustered coefficient model via adaptive regularization

Yan Zhong [a,*], Huiyan Sang [b], Scott J. Cook [c], Paul M. Kellstedt [c]

[a] KLATASDS-MOE, School of Statistics, East China Normal University, China
[b] Department of Statistics, Texas A&M University, United States of America
[c] Department of Political Science, Texas A&M University, United States of America

ARTICLE INFO

Article history:
Received 10 November 2021
Received in revised form 18 July 2022
Accepted 22 July 2022
Available online 29 July 2022

Keywords:
Spatial variable selection
Variable-dependent graph
Varying coefficient regression
COVID-19 vaccination acceptance

ABSTRACT

Large spatial datasets with many spatial covariates have become ubiquitous in many fields in recent years. A question of interest is to identify which covariates are likely to influence a spatial response, and whether and how the effects of these covariates vary across space, including potential abrupt changes from region to region. To solve this question, a new efficient regularized spatially clustered coefficient (RSCC) regression approach is proposed, which could achieve variable selection and identify latent spatially heterogeneous covariate effects with clustered patterns simultaneously. By carefully designing the regularization term of RSCC as a chain graph guided fusion penalty plus a group lasso penalty, the RSCC model is computationally efficient for large spatial datasets while still achieving the theoretical guarantees for estimation. RSCC also adopts the idea of adaptive learning to allow for adaptive weights and adaptive graphs in its regularization terms and further improves the estimation performance. RSCC is applied to study the acceptance of COVID-19 vaccines using county-level data in the United States and discover the determinants of vaccination acceptance with varying effects across counties, revealing important within-state and across-state spatially clustered patterns of covariates effects.

1. Introduction

Large spatial data, which include numerous observations and variables on a large spatial domain, have become increasingly common across a number of real-world applications. Using these spatial data, researchers are often curious to discover the relationship between a response variable and a set of covariates while also accounting for spatial dependence among the observations. In conventional spatial regression models, the relationships between response and a pre-determined set of covariates are often assumed to be constants over the spatial domain, and the spatial dependence among observations unexplained by the covariates is captured by a spatial random effect or, equivalently, by a spatially varying intercept term. However, two problems arise when dealing with large spatial data: First, as a study domain becomes larger, there may exhibit spatially heterogeneous effects of covariates on the response. Second, when many covariates are available, to properly study the spatial heterogeneity in covariates effects, variable selection is usually needed to avoid overfitting and improve model interpretability.

* Corresponding author.
E-mail address: yzhong@fem.ecnu.edu.cn (Y. Zhong).

https://doi.org/10.1016/j.csda.2022.107581
0167-9473/

To date, most existing methodologies in spatial statistics have separately focused on one of these two issues—that is, modeling spatially heterogeneous covariates effects or spatial variable selection in large spatial data. Firstly, spatially varying coefficient regression (SVC) is an essential tool to account for spatially heterogeneous covariate effects, as SVC regression allows the linear relationships between response and covariates to vary across the spatial domain. Suppose that a response variable, y, and p covariates, $\mathbf{x}$, are collected at n locations with $(y_1, \mathbf{x}_n), \ldots, (y_n, \mathbf{x}_n) \in \mathbb{R} \times \mathbb{R}^p$. For each location, the SVC model assumes that y_i and $\mathbf{x}_i$ follow

$$y_i = a + \mathbf{x}_i^T \mathbf{b}_i + \epsilon_i, \tag{1}$$

where $\mathbf{b}_i \in \mathbb{R}^p$ is the location-specific coefficient vector, a is the global intercept term fixed for all locations, and ϵ_i is a random error term. To capture the spatial dependence unexplained by covariates, the first entry of $\mathbf{x}_i$ can be set to 1, and then the first entry of $\mathbf{b}_i$ refers to the spatially varying intercept for the i^{th} location. Among existing SVC models, two different kinds of assumptions are usually made on the spatial dependence of $\mathbf{b}_i$: The first kind of assumption is that $\mathbf{b}_i$ changes smoothly across the spatial domain. For example, the geographically weighted regression model (GWR, Fotheringham et al., 2003) extends the local linear regression to the geographical content to estimate the varying coefficients with smooth spatial kernel weighting functions. Gelfand et al. (2003) build a Bayesian framework of SVC by formulating a multivariate Gaussian process for $\mathbf{b}_i$. Recently, Kim and Wang (2021) propose a generalized SVC with $\mathbf{b}_i$ approximated by smoothing splines. The second kind of assumption is that the effects of covariates vary across the study domain rather abruptly across areas within the study domain, thereby forming latent spatially clustered patterns. Under this assumption, Li and Sang (2019) propose a spatially clustered coefficient regression (SCC) model to estimate $\mathbf{b}_i$, which builds a graph fused lasso regularized optimization to encourage homogeneity between $\mathbf{b}_i$ at two adjacent locations. Zhang et al. (2019) extend the SCC model to a regression setting over networks, and Lee et al. (2021) provide a clustered SVC model for spatio-temporal data using scan statistics. Luo et al. (2021) construct the Bayesian spatially clustered coefficient (BSCC) model with a random spanning tree cut partition prior. Each of the studies mentioned above is designed for small p, that is, for studies with few covariates. When p is large, model overfitting will occur in most of these approaches, and many also face the high computational cost issue, especially the Bayesian methods.

Secondly, in research on high-dimensional statistics, several methods have been proposed for spatial variable selection. Zhu et al. (2010) develop a spatial adaptive lasso method for simultaneous model selection and parameter estimation in spatial regression for lattice data. Feng et al. (2016) and Shin et al. (2019) consider binary spatial regression models and proposed penalized quasi-likelihood methods with spatial dependence for variable selections. Thurman et al. (2015) and Choiruddin et al. (2018) propose variable selection methods via sparse regularization for point process models. Yet, each of these methods assumes that the regression coefficients of selected variables are constant across space. In doing so, these approaches may eliminate relevant variables with complex spatial relationships with the response that only emerge once spatially varying covariate effects are undertaken.

Currently, the study of variable selection in SVC models is at an early stage with limited work. Smith and Fahrmeir (2007) work on lattice data and propose a Bayesian variable selection procedure for SVC models. Reich et al. (2010) propose a Bayesian variable selection of Gaussian process-based SVC with a stochastic search algorithm to select important covariates, which require large computation when the numbers of locations and covariates are large. Wheeler (2009) and Li and Lam (2018) provide GWR-based variable selection method with lasso and elastic-net penalties respectively. Both of them return variable selection results at each location that may not be easy to interpret in real applications. Recently, Dambon et al. (2021) provide a variable selection model for Gaussian process-based SVC models using optimization and covariance tapering to reduce computation. Each of these methods is based on the assumption of smoothly varying coefficients. Therefore, a generally flexible and computationally efficient model is needed for researchers to implement simultaneous variable selection and identification of spatial patterns in regression coefficients, especially in the case that the coefficients vary with abrupt changes across the study domain with spatially clustered patterns.

Under the assumption that coefficients vary abruptly, we propose a new efficient regularized spatially clustered coefficient (RSCC) model which selects important variables and estimates spatially varying coefficients simultaneously. Our model can be seen as an extension of the spanning tree graph fused lasso-based SCC model proposed by Li and Sang (2019), with a newly designed regularization that differs from their method in several respects. Our regularization takes an additive form of two terms: The first term is a fusion penalty guided by a carefully designed variable-dependent chain graph that delivers covariate-specific spatially clustered patterns. The second term is a group lasso penalty to achieve spatial variable selection. This specific selection of the regularization term can speed up the algorithm and achieve low space and time complexity, thereby making the model well suited for large spatial datasets. This contrasts with naïvely combining existing general or tree graph-based fusion and group sparsity penalties, which would encounter severe computational challenges. In particular, we introduce the idea of adaptive learning to allow for adaptive weights and adaptive graphs in our regularization terms, extending upon current SCC regularization methods that are built upon fixed graphs (e.g. Li and Sang (2019); Sass et al. (2021)). Computationally, we provide an efficient proximal gradient algorithm tailored for our model which has low computational complexity. Theoretically, we provide the non-asymptotic theoretical results to justify the proposed new regularization in our approach. We also show that the adaptive method has a faster convergence rate than its non-adaptive counterpart.

In addition, the proposed RSCC model has several other advantages. It allows the investigation of different clustered patterns with great flexibility in their shapes for different regression coefficients. Moreover, the number of selected variables

and clustered patterns are both treated as unknown and determined from data-driven approaches. Finally, since the method is built upon graphs, it can be used beyond the spatial context to solve the variable selection problem in other clustered coefficient models where observations can be related by a graph or a network. For examples of such contexts, see our discussion in Section 7.

To demonstrate the utility of the RSCC model to real data, we undertake a study of county-level COVID-19 vaccination rates in the United States. Several recent studies have shown that COVID-19 vaccination uptake varies by county and is not consistently correlated to key characteristics (Tolbert et al., 2021; Mollalo and Tatar, 2021). As these studies demonstrate, the novelty of the recent pandemic means researchers have yet to arrive at a canonical set of covariates that explain cross-county vaccination rates. Therefore, rather than only focus on a small pre-defined set of variables, we instead collect 38 social, demographic, and economic variables to serve as the candidate set of potential covariates. Our analysis of 3076 counties identifies a number of influential factors (e.g., education, households, and cumulative COVID-19 cases) that govern vaccination acceptance. Moreover, our findings also indicate several interesting within-state and across-state spatially clustered patterns of covariate effects.

The paper is organized as follows. In Section 2, we propose our RSCC model. Section 3 describes the efficient algorithm for the implementation of our model. In Section 4, we present the theoretical results of this model. Sections 5 and 6 include the simulations to illustrate the model performance and the application to the COVID-19 vaccination acceptance. We offer discussions in Section 7. The proofs, additional simulation results, and information about the real data are provided in the supplementary material.

2. Methodology

2.1. Spatially clustered coefficients

We first review the original spatially clustered coefficients (SCC) model described by Li and Sang (2019) that ignores variable selections. To begin with, SCC encodes the spatial proximity information into a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the vertex set $\mathcal{V}$ represents n locations, and the edge set $\mathcal{E}$ consists of the edges between locations. $\mathcal{G}$ is required to be a connected graph that for each pair of vertices in $\mathcal{V}$, there is a path of edges in $\mathcal{E}$ connecting them. When we have point-referenced spatial data, one popular choice is to construct the nearest neighbor graph that connects each vertex with its K-nearest neighbors (K-NN graph) or neighbors within a certain radius r (r-NN graph). In practice, the number of neighbors in K-NN graph or the radius in r-NN graph needs to be chosen with care to guarantee that $\mathcal{G}$ is a connected graph. Another approach is to use the Delaunay triangulation graph (Lee, 1980). When spatial data are areal units (polygons), one may construct $\mathcal{G}$ treating areal units as point-referenced data using the centroid point to represent each unit. Alternatively, $\mathcal{G}$ can be constructed to reflect the bordering information between areal units. We call this kind of $\mathcal{G}$ built by spatial information as a spatial graph.

Based on $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, SCC estimates the coefficient matrix $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)^T \in \mathbb{R}^{n \times p}$ of the SVC model (1) by solving

$$\underset{a, \mathbf{B}}{\operatorname{argmin}} \frac{1}{2n} \sum_{i=1}^{n} (y_i - a - \mathbf{x}_i^T \mathbf{b}_i)^2 + \lambda P(\mathbf{B}; \mathcal{G}), \tag{2}$$

where

$$P(\mathbf{B}; \mathcal{G}) = \sum_{j=1}^{p} \sum_{(i_1, i_2) \in \mathcal{E}} |b_{i_1, j} - b_{i_2, j}| \tag{3}$$

is the graph fused lasso penalty (Hoefling, 2010), which incorporates graph information to pursue latent clustered patterns of regression coefficients. For the j^{th} covariate, the graph fused lasso pursues a sparse solution on $b_{i_1, j} - b_{i_2, j}$ for $(i_1, i_2) \in \mathcal{E}$. When $b_{i_1, j} - b_{i_2, j} = 0$, the i_1^{th} and i_2^{th} locations shares the same coefficient. On the contrary, the non-zero elements of $b_{i_1, j} - b_{i_2, j}$ correspond to a set of edges which, if removed from the graph, will partition the vertices of $\mathcal{G}$ into a number of disjoint connected components. These edges can be called as the cutting edge of $\mathcal{G}$ induced by $P(\mathbf{B}; \mathcal{G})$. By using a spatial graph in $P(\mathbf{B}; \mathcal{G})$, the SCC method naturally partitions locations into several disjoint groups that locations in the same group are spatially continuous and share the same coefficient. Thus, each regression coefficient is piecewise constant and can be interpreted as having a spatially clustered pattern in the spatial domain. For different covariates, $P(\mathbf{B}; \mathcal{G})$ also allows them to have different latent clustered patterns in their coefficients.

The main issue of the SCC model in (2) is that solving the regression problem with the general graph fused lasso is usually computationally expensive. Though researchers have developed several alternating direction methods of multipliers (ADMM) (Boyd et al., 2011) and path following algorithms (Shen and Huang, 2010; Arnold and Tibshirani, 2016) to solve the graph fused lasso problems, their computation cost could be high for a large general graph $\mathcal{G}$. To solve this problem, Li and Sang (2019) replace the original spatial graph with a common fixed spanning tree graph for all covariates.

Moreover, the SCC model in (2) could not achieve variable selection, and when p is large, it faces the problem of overfitting and is unidentifiable. To achieve variable selection and varying coefficient estimation simultaneously, we propose a new

regularized SCC (RSCC) model with two sources of constraints on the structure of $\mathbf{B}$: one focusing on spatial heterogeneity like (3), and one focusing on variable selection. Specifically, RSCC estimates $\mathbf{B}$ by minimizing the objective function:

$$L(a, \mathbf{B}) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - a - \mathbf{x}_i^T\mathbf{b}_i)^2 + \lambda_1 P_C(\mathbf{B}) + \lambda_2 P_V(\mathbf{B}), \tag{4}$$

where λ_1 and λ_2 are two penalty parameters, P_C is the penalty function to capture the spatially clustered patterns of $\mathbf{B}$, and P_V is the penalty function to select relevant variables. In the following paragraphs, we will discuss the specific design of P_C and P_V to achieve an effective and efficient model.

2.2. The selection of P_C

The straightforward selection of P_C is to directly use the graph fused lasso in (3). However, the computational issue of the graph fused lasso is even aggravated in our model with two regularization terms, and hence a penalty enabling efficient computations is desired for P_C.

Padilla et al. (2018) show that when a graph has a certain simple structure such as a chain or a tree graph structure, one can take advantage of these structures to design efficient algorithms to solve the graph fused lasso problem. We are motivated to consider a similar strategy to replace the original graph in the graph fused lasso penalty with a simple graph chosen in an adaptive fashion, which includes the proximate pairs of samples that are likely to share similar coefficients. Different from selecting a tree graph in the SCC model by Li and Sang (2019), we select to design a chain graph in our model, since, among the many choices of the simple graph, the chain graph could achieve the fastest computation (Padilla et al., 2018). The main advantage of the chain graph lies in the fact that it provides an order of n observations so that the graph fused lasso is reduced to a 1-dimensional fused lasso problem, for which several existing efficient algorithms (Barbero and Sra, 2018; Zhou et al., 2012) can be adapted to solve the problem.

2.2.1. Variable dependent chain graph

Care needs to be taken to generate a chain graph from the original spatial graph $\mathcal{G}$ because naïvely choosing one would severely sacrifice model accuracy. Compared to the original graph, a chain graph may include edges that connect two disconnected vertices in the original graph. When partitioning a chain graph by cutting a set of edges, the number of disjoint connected components will be the number of cutting edges plus 1, which means that the number of piecewise constants in a variable's spatially clustered coefficient is equal to the number of estimated cutting edges plus 1 when using a chain graph in the graph fused lasso. Thus, with a poor choice of chain graph, the number of piecewise constants in a variable's spatially clustered coefficient estimated by the model will be much larger than that with the original graph.

We now briefly introduce two essential tools of graph analysis, the minimum spanning tree (MST) and the depth-first searching (DFS) algorithms, which are used to design a new chain graph for our model.

MST is a subgraph of the original $\mathcal{G}$. By assuming that each edge in $\mathcal{G}$ has a corresponding weight, MST is the spanning tree of $\mathcal{G}$ that connects all vertices of the original graph with no cycles and with minimum total edge weights. Thus, the vertex set of MST includes all n observed locations, and the edge set of MST is an edge subset of size $n-1$ of the original graph. When the weight of each edge reflects the spatial distance of its two vertices, MST is in general capable of preserving spatial proximity information in $\mathcal{G}$.

DFS is a commonly used algorithm to generate a chain structure from a graph, which travels from a root vertex and explores other vertices through edges along each branch as far as possible before backtracking. As proved by Padilla et al. (2018), the graph fused lasso penalty constructed over the DFS-induced chain will not exceed twice the graph fused lasso penalty over the original graph. In our situation, this property helps us control the number of edges connecting observed locations with different coefficient values in the proposed chain graph.

As different variables could have different clustered patterns of coefficients, we propose a variable-dependent chain graph with adaptive learning. For each variable, we generate a specific chain graph $\mathbf{G}_j$: We begin with an initial estimator $\mathbf{B}^{\text{ini}}$ of $\mathbf{B}$, whose choice will be discussed in Section 3.4. For the j^{th} variable, $|b^{\text{ini}}_{i_1,j} - b^{\text{ini}}_{i_2,j}|$ can be used to reflect the variable-specific difference between locations i_1 and i_2, and a smaller value of $|b^{\text{ini}}_{i_1,j} - b^{\text{ini}}_{i_2,j}|$ indicates that i_1 and i_2 are more likely to have the same coefficient in reality. Using $|b^{\text{ini}}_{i_1,j} - b^{\text{ini}}_{i_2,j}|$ as the edge weight, we build a MST of the original graph. Then, we apply the DFS algorithm to this MST to generate the variable-dependent chain graph $\mathcal{G}_j$, referred to as the adaptive MST-DFS-induced chain. Fig. 1 provides an example to generate $\mathcal{G}_j$ with 9 vertices.

Algorithm 1 summarizes the steps to generate the proposed chain graphs. Indeed, theoretically, we will prove in Theorem 3 in Section 4 that under mild assumptions, the number of edges connecting two observed locations with different piecewise constants in the j^{th} spatially varying coefficient in the adaptive MST-DFS-induced chain will not exceed $2(R_j - 1)$ as $n \to \infty$, where R_j is the true number of piecewise constants. This nice property guarantees a good balance between model accuracy and computational efficiency when using the adaptive MST-DFS-induced chain in the graph fused lasso.

2.2.2. Chain graph-based adaptive fused lasso

We now provide the specific form of P_C with $\mathcal{G}_j, j = 1, \ldots, p$ below. For the j^{th} variable, its chain graph $\mathcal{G}_j$ defines a direct path of n samples, i.e. $(j_1 \to j_2 \to \cdots \to j_n)$. To reduce the estimation bias of the lasso-type penalty and improve

Algorithm 1 Variable-dependent Chain Graph.

Input: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, an initial estimator $\mathbf{B}^{\text{ini}}$ of $\mathbf{B}$.
for $j = 1, 2, \ldots, p$: **do**
(1) Construct a MST of $\mathcal{G}$ with $|b^{\text{ini}}_{i_1,j} - b^{\text{ini}}_{i_2,j}|$ as the edge weight of $(i_1, i_2) \in \mathcal{E}$.
(2) Apply the DFS algorithm to the MST in (1) to generate a chain graph, denoted as $\mathcal{G}_j$.
end for
Output: $\mathcal{G}_j$, $j = 1, \ldots, p$.

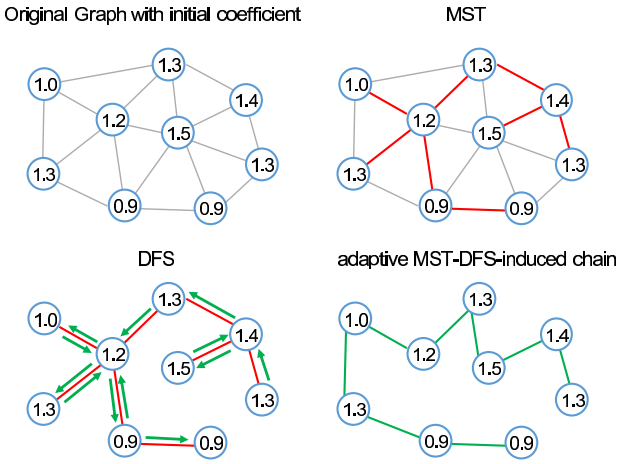

Fig. 1. An example of deriving the adaptive MST-DFS-induced chain from the original graph for a variable: Left-top: The original graph whose values on vertices represent an initial estimation of coefficients; Right-top: For each edge, using the absolute difference between the values of its two vertices as the edge weight to generate an MST from the original graph, showing in red color; Left-bottom: The DFS algorithm is applied on the MST. The green arrows show the path of DFS to explore all vertices of the MST; Right-bottom: The order that DFS reaches the vertices forms a chain graph, which is the adaptive MST-DFS-induced chain. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

variable selection accuracy, we also adopted adaptive weights (Zou, 2006; Viallon et al., 2013) in P_C. Thus, our proposed P_C is a chain graph-based adaptive fused lasso penalty with the form:

$$P_C(\mathbf{B}) = P_C(\mathbf{B}; \mathcal{G}_1, \ldots, \mathcal{G}_p) = \lambda_1 \sum_{j=1}^{p} \sum_{i=1}^{n-1} w_{(j_i, j_{i+1}),j} |b_{j_i,j} - b_{j_{i+1},j}|,$$

where $w_{(j_i,j_{i+1}),j}$ denotes the adaptive weight for the edge $(j_i, j_{j_{i+1}})$ in $\mathcal{G}_j$ determined from an initial estimator $\mathbf{B}^{\text{ini}}$, and $b_{j_i,j}$ denotes the $(j_i, j)^{\text{th}}$ entry of $\mathbf{B}$.

For the sake of notation simplicity, we use $w_{j,i}$ to represent $w_{(j_i,j_{i+1}),j}$. Let $\mathbf{H}_j \in \mathbb{R}^{(n-1)\times n}$ be the incidence matrix of $\mathcal{G}_j$; the $(i, j_i)^{\text{th}}$ entry of $\mathbf{H}_j$ is equal to 1, the $(i, j_{i+1})^{\text{th}}$ entry of $\mathbf{H}_j$ is equal to -1 for $i = 1, 2, \ldots, n-1$, and all the other entries of $\mathbf{H}_j$ are 0. Denote $\mathbf{W}_j \in \mathbb{R}^{(n-1)\times(n-1)}$ as the diagonal matrix with $w_{j,i}$ along the diagonal.

Then, $P_C(\mathbf{B})$ can be written in the matrix form:

$$P_C(\mathbf{B}) = \lambda_1 \sum_{j=1}^{p} \left\| \mathbf{W}_j \mathbf{H}_j \mathbf{B}_j \right\|_1 . \tag{5}$$

2.3. The selection of P_V

We then introduce the form of P_V to impose sparsity regularization to determine the non-zero columns $\mathbf{B}_j$. Note that when a spatial covariate is not a true predictor, the corresponding column of regression coefficients of this covariate becomes a zero vector. By treating the parameters in each $\mathbf{B}_j$ as a group, P_V should help achieve the group selection of parameters and hence identify important spatial covariates to avoid over-fitting in estimation and improve the interpretation of the model. There are many existing methods that focus on group selection (see (Huang et al., 2012) for a selective review of the literature). In this paper, we choose the adaptive group lasso for the form of P_V, since the adaptive group lasso method can achieve the oracle property for the regression problem with group selection (Wang and Leng, 2008).

Specifically, $P_V(\mathbf{B})$ takes the form

$$P_V(\mathbf{B}) = \sum_{j=1}^{p} u_j \left\| \mathbf{B}_j \right\|_2 , \tag{6}$$

where u_j is a weight parameter for adaptive learning determined from an initial estimator $\mathbf{B}^{\text{ini}}$.

2.4. Sparse spatially clustered coefficients

Denote $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^T \in \mathbb{R}^{n\times p}$ as the design matrix. Let $\tilde{\mathbf{X}} = \left[\text{diag}(\mathbf{X}_1), \ldots, \text{diag}(\mathbf{X}_p)\right] \in \mathbb{R}^{n\times(np)}$ where $\mathbf{X}_j \in \mathbb{R}^n$ is the j^{th} column vector of $\mathbf{X}$. Denote $\text{vec}(\mathbf{B}) = (\mathbf{B}_1^T, \ldots, \mathbf{B}_p^T)^T$. By incorporating the penalties P_C in Eq. (5) and P_V in Eq. (6), we have the final matrix form of the objective function which we minimize to obtain an estimator of $\boldsymbol{\theta} = (a, \text{vec}(\mathbf{B})^T)^T \in \mathbb{R}^{np+1}$:

$$L(\boldsymbol{\theta}) = \frac{1}{2n}\left\|\mathbf{Y} - \mathbf{1}a - \tilde{\mathbf{X}}\text{vec}(\mathbf{B})\right\|_2^2 + \lambda_1 \sum_{j=1}^{p} \left\|\mathbf{W}_j\mathbf{H}_j\mathbf{B}_j\right\|_1 + \lambda_2 \sum_{j=1}^{p} u_j \left\|\mathbf{B}_j\right\|_2. \tag{7}$$

It is noticeable that $L(\boldsymbol{\theta})$ is a convex function of $\boldsymbol{\theta}$, which can be efficiently solved by the convex optimization method to be introduced in Section 3.

3. Algorithm

3.1. Proximal gradient method

Taking advantage of the fact that the proximal operators of P_C and P_V can be solved efficiently, we minimize the convex objective function in Eq. (7) using the proximal gradient method.

Denote the first term of $L(\boldsymbol{\theta})$ as $f(\boldsymbol{\theta}) = \frac{1}{2n}\left\|\mathbf{Y} - \mathbf{1}a - \tilde{\mathbf{X}}\text{vec}(\mathbf{B})\right\|_2^2$. Firstly, with a fixed point $\boldsymbol{\theta}^t$, we bound $L(\boldsymbol{\theta})$ by:

$$L(\boldsymbol{\theta}) \le f(\boldsymbol{\theta}^t) + < \boldsymbol{\theta} - \boldsymbol{\theta}^t, \nabla f(\boldsymbol{\theta}^t) > + \frac{l}{2}\left\|\boldsymbol{\theta} - \boldsymbol{\theta}^t\right\|_2^2 + P_{C+V}(\mathbf{B}),$$

where $P_{C+V}(\mathbf{B}) = P_C(\mathbf{B}) + P_V(\mathbf{B})$, $\nabla f(\boldsymbol{\theta}^t)$ is the gradient of $f(\boldsymbol{\theta})$ at $\boldsymbol{\theta}^t$, $< \cdot, \cdot >$ denotes an inner product operator, and l is the Lipschitz constant of $\nabla f(\boldsymbol{\theta})$ that $|\nabla f(\boldsymbol{\theta}^\flat) - \nabla f(\boldsymbol{\theta}^\sharp)| \le l\left\|\boldsymbol{\theta}^\flat - \boldsymbol{\theta}^\sharp\right\|_F$ for any $\boldsymbol{\theta}^\flat, \boldsymbol{\theta}^\sharp \in \mathbb{R}^{n\times p}$. We can choose l as two times of the largest eigenvalue of $(\mathbf{1}, \tilde{\mathbf{X}})^{\mathbf{T}}(\mathbf{1}, \tilde{\mathbf{X}})/\mathbf{n}$ (Beck and Teboulle, 2009).

Secondly, the proximal gradient method updates the value of $\boldsymbol{\theta}$ iteratively with the previous point $\boldsymbol{\theta}^t$ by solving:

$$\boldsymbol{\theta}^{t+1} = \underset{\boldsymbol{\theta}}{\text{argmin}}\, \frac{1}{2}\|\boldsymbol{\theta} - \mathbf{Z}\|_2^2 + \frac{1}{l}P_{C+V}(\mathbf{B}), \tag{8}$$

where $\mathbf{Z} = \boldsymbol{\theta}^t - (1/l)\nabla f(\boldsymbol{\theta}^t)$. We observe that updating $\boldsymbol{\theta}$ via solving Eq. (8) is equivalent to update a and the columns of $\mathbf{B}$ separately via $a^{t+1} = Z_0$ and

$$\mathbf{B}_j^{t+1} = \underset{\mathbf{B}_j}{\text{argmin}}\, \frac{1}{2}\left\|\mathbf{B}_j - \mathbf{Z}_j\right\|_2^2 + \frac{\lambda_1}{l}\left\|\mathbf{W}_j\mathbf{H}_j\mathbf{B}_j\right\|_1 + \frac{\lambda_2 u_j}{l}\left\|\mathbf{B}_j\right\|_2, \tag{9}$$

for $j = 1, 2, \ldots, p$, where $(Z_0, \mathbf{Z}_1^T, \ldots, \mathbf{Z}_p^T)^T = \mathbf{Z}$. Eq. (9) belongs to the proximal operator problem associated with the combination of the weight-included fused lasso and the group lasso.

We further improve the above proximal gradient method with two computational strategies. First, since $L(\boldsymbol{\theta})$ is a convex function, a that minimizes $L(\boldsymbol{\theta})$ should be a solution of $\frac{\partial L(\boldsymbol{\theta})}{\partial \alpha} = 0$. By simple algebra, we have $a = \bar{Y} - \bar{\mathbf{X}}\text{vec}(\mathbf{B})$, where $\bar{Y}$ is the mean value of $\mathbf{Y}$ and $\bar{\mathbf{X}}$ contains the mean value of each column of $\tilde{\mathbf{X}}$. Thus, we update a^{t+1} by

$$a^{t+1} = \bar{Y} - \bar{\mathbf{X}}\text{vec}(\mathbf{B}^{t+1}), \tag{10}$$

instead of $a^{t+1} = Z_0$. Second, we adopt the fast version of proximal gradient methods proposed by Beck and Teboulle (2009), which replaces the fixed value $\boldsymbol{\theta}^t$ in $\mathbf{Z}$ by:

$$\tilde{\boldsymbol{\theta}}^t = \boldsymbol{\theta}^t + \frac{\alpha^{t-1} - 1}{\alpha^t}(\boldsymbol{\theta}^t - \boldsymbol{\theta}^{t-1}). \tag{11}$$

By using this strategy, the number of iterations of proximal gradient methods is proved to improve from $O(1/\epsilon)$ to $O(1/\sqrt{\epsilon})$ to achieve a convergence tolerance of ϵ.

We summarize the details of our proximal gradient method in Algorithm 2.

Algorithm 2 Proximal Gradient Method.

Input: $\mathbf{Y}$, $\mathbf{X}$, $\mathcal{G}_j$, $j = 1, \ldots, p$, λ_1, λ_2, l, ϵ, $\alpha^{-1} = 1$, $\boldsymbol{\theta}^0 = \boldsymbol{\theta}^{-1} = \mathbf{0}$.
for $t = 0, 1, 2, \ldots$: **do**
  Calculate $\alpha^t = 1 + \sqrt{1 + 4(\alpha^{t-1})^2}/2$;
  Construct $\tilde{\boldsymbol{\theta}}^t$ by Eq. (11);
  Calculate $(Z_0, \mathbf{Z}_1^T, \cdots, \mathbf{Z}_p^T) = \mathbf{Z} = \tilde{\boldsymbol{\theta}}^t - (1/l)\nabla f(\tilde{\boldsymbol{\theta}}^t)$;
  Update $\mathbf{B}^{t+1}$ by solving Eq. (8) and get a^{t+1} by Eq. (10);
end for
Stopping criterion: Keep the above iteration until t satisfies $L(\boldsymbol{\theta}^t) - L(\boldsymbol{\theta}^{t+1}) < \epsilon$
Output: $\boldsymbol{\theta}^{t+1}$.

3.2. Solving the proximal operator

In this subsection, we discuss the way to solve Eq. (9) efficiently. We first define the standard form of the proximal operator associated with the adaptive fused lasso and the group lasso as follows: For any vector $z \in \mathbb{R}^n$, the proximal operator associated with the adaptive fused lasso and the group lasso is to solve:

$$\text{prox}_{FL+GL}(\mathbf{z}, \lambda_1, \lambda_2) = \underset{\boldsymbol{\beta}}{\text{argmin}} \, \|\mathbf{z} - \boldsymbol{\beta}\|_2^2 + \lambda_1 \|\mathbf{WH}\boldsymbol{\beta}\|_1 + \lambda_2 \|\boldsymbol{\beta}\|_2 , \tag{12}$$

where $\mathbf{H} \in (n-1) \times n$ is a matrix whose $(i,i)^{\text{th}}$ entry is 1 and $(i, i+1)^{\text{th}}$ entry is -1 for $i = 1, 2, \ldots, n-1$. The other entries of $\mathbf{H}$ are all 0. $\mathbf{W}$ is a diagonal matrix containing the adaptive weights. It is easy to check that Eq. (9) is a specific case of Eq. (12) by reordering the rows of $\mathbf{W}_j\mathbf{H}_j\mathbf{B}_j$ and choosing particular values for $\mathbf{z}$, $\boldsymbol{\beta}$, λ_1, and λ_2. To efficiently solve Eq. (12), we propose the following theorem:

Theorem 1. *Define:*

$$prox_{FL}(\mathbf{z}, \lambda_1) = \underset{\boldsymbol{\beta}}{\text{argmin}} \, \|\mathbf{z} - \boldsymbol{\beta}\|_2^2 + \lambda_1 \|\mathbf{WH}\boldsymbol{\beta}\|_1 ,$$

and

$$prox_{GL}(\mathbf{z}, \lambda_2) = \underset{\boldsymbol{\beta}}{\text{argmin}} \, \|\mathbf{z} - \boldsymbol{\beta}\|_2^2 + \lambda_2 \|\boldsymbol{\beta}\|_2 .$$

$prox_{FL}(\mathbf{z}, \lambda_1)$, $prox_{GL}(\mathbf{z}, \lambda_1)$, and $prox_{FL+GL}(\mathbf{z}, \lambda_1, \lambda_2)$ have the following relation:

$$prox_{FL+GL}(\mathbf{z}, \lambda_1, \lambda_2) = prox_{GL}\big(prox_{FL}(\mathbf{z}, \lambda_1), \lambda_2\big).$$

Theorem 1 can be proved in a similar way to Theorem 1 in Zhou et al. (2012). Theorem 1 shows that Eq. (12) can be computed through a two-step strategy which involves $\text{prox}_{FL}(\mathbf{z}, \lambda_1)$ and $\text{prox}_{GL}(\mathbf{z}, \lambda_2)$ respectively. For $\text{prox}_{GL}(\mathbf{z}, \lambda_2)$, its analytical solution is:

$$\text{prox}_{GL}(\mathbf{z}, \lambda_2) = \left\{ \frac{\max(0, \|\mathbf{z}\|_2 - \lambda_2)}{\|\mathbf{z}\|_2} \right\} \mathbf{z}.$$

For $\text{prox}_{FL}(\mathbf{z}, \lambda_1)$, several efficient algorithms with $O(n)$ computational complexity have been developed based on the ideas of taut string and dynamic programming (Johnson, 2013; Barbero and Sra, 2018). Condat (2013) develops another fast algorithm via the analysis of KKT conditions, which is easy to implement and only requires $n + 7$ storage. Precisely, if m is the number of nonzero entries in $\mathbf{H} \cdot \text{prox}_{FL}(\mathbf{z}, \lambda_1)$, the computational complexity of Condat's method is $O(nm)$. In this paper, we use the weight-included version of Condat's method for its easy implementation. The detailed algorithm is provided in the supplementary material.

Overall Computational Complexity: We now make a summary of the overall computational complexity to solve RSCC. In Algorithm 1, the computational complexity of finding the MST is $O(|\mathcal{E}| \log n)$ where $|\mathcal{E}|$ is the number of edges in the original graph, and the computational complexity for running the DFS algorithm on the MST is $O(n)$. Thus, the computational cost of constructing the adaptive MST-DFS included chains for all variables is $O(p|\mathcal{E}| \log n)$. In Algorithm 2, to achieve a convergence tolerance of ϵ, the number of iterations should be at least $O(1/\sqrt{\epsilon})$. As the smallest computational complexity to solve each small problem in Eq. (9) is $O(n)$ (Barbero and Sra, 2018), the computational cost for Algorithm 2 is $O(np/\sqrt{\epsilon})$. To conclude, the overall computational complexity of RSCC is $O(p|\mathcal{E}| \log n + np/\sqrt{\epsilon})$, which is applicable for large datasets.

3.3. Tuning parameters

There are two tuning parameters in our model: λ_1 for seeking spatially clustered patterns and λ_2 for variable selection. To make a proper choice of their values, we use the generalized information criterion (GIC). GIC is proposed by Fan and Tang

(2013) to select the tuning parameters in penalized generalized linear models. Different from the traditional information criteria such as the Bayes information criterion (BIC), GIC is applicable to the case when the number of parameters increases at most exponentially with the sample size n. Our model considers the case that includes $(np+1)$ parameters in total with $p = O(n^c)$ for some $c \geq 0$, and then we calculate its GIC by:

$$\mathrm{GIC}(\lambda_1, \lambda_2) = \log\left[\frac{1}{n}\sum_{i=1}^{n}(y_i - a - \mathbf{x}_i\mathbf{b}_i)^2\right] + \frac{1}{n}\log\{\log(n)\}\log(n)\cdot \mathrm{df}(\lambda_1, \lambda_2), \tag{13}$$

where $\mathrm{df}(\lambda_1, \lambda_2)$ is the degree of freedom of the model with tuning parameters (λ_1, λ_2) and is estimated by the number of selected edges plus the total number of predicted cutting edges in $\mathcal{G}_1, \ldots, \mathcal{G}_p$. Theorem 3 of Fan and Tang (2013) shows that the GIC defined in Eq. (13) can consistently identify the true model.

In practice, assume that there are m_1 and m_2 candidates for the values of λ_1 and λ_2 respectively, and we need to find the combination of (λ_1, λ_2) leading to the smallest GIC. Instead of running $m_1 m_2$ models with different value combinations of penalty parameters separately, we adopt the strategy of warm starts (Hastie et al., 2015) to sequentially solve $m_1 m_2$ models, which dramatically reduces the overall computing time.

3.4. Initial estimator and adaptive weights

An initial estimator $\mathbf{B}^{\mathrm{ini}}$ of $\mathbf{B}$ is needed for building adaptive MST-DFS-induced chains and determining adaptive weights $\mathbf{W}_j$ and u_j in Eq. (7). With $\mathbf{B}^{\mathrm{ini}}$, u_j and $\mathbf{W}_j$ could be determined by:

$$u_j = (\|\mathbf{B}_j^{\mathrm{ini}}\|_2)^{-\alpha} \text{ and } \mathbf{W}_j = \mathrm{diag}[(\mathbf{H}_j\mathbf{B}_j^{\mathrm{ini}})^{-\alpha}].$$

α is a tuning parameter that is usually set to 1 or 2.

The traditional selection of $\mathbf{B}^{\mathrm{ini}}$ is $\mathbf{B}$'s ordinary least square (OLS) estimator, whereas, in our problem, the number of parameters is larger than the number of observations, and the OLS estimator is not unique. Instead, we run the non-adaptive version of RSCC to get the initial estimator that we will prove is consistent in Theorem 2 under mild conditions. Specifically, we build a common MST-DFS chain graph $\mathcal{G}^*$ for all *variables* by first finding the MST of the original graph with the Euclidean distance between observed locations as the weight of each edge and then run the DFS algorithm on this MST to get the chain graph $\mathcal{G}^*$. By using $u_i = 1$, $\mathbf{W}_j = \mathbf{I}_{n-1}$, and $\mathcal{G}_j = \mathcal{G}^*$, $j = 1, \ldots, p$, we solve Eq. (7) and obtain the initial estimator $\mathbf{B}^{\mathrm{ini}}$.

4. Theorem

We now show that besides fast computing, RSCC achieves theoretical guarantees for estimation. To make a brief overview, we first provide a non-asymptotic bound for the estimation error of our model that depends on the number of truly related variables and the number of edges in graphs connecting locations with different coefficients in Theorem 2. Then we provide Corollary 1 that shows the selection consistency of our estimator. Finally, Theorem 3 shows the relationship between the number of piecewise constants in the spatially clustered coefficients and the number of edges connecting locations with different coefficient values in the variable-dependent chain graph.

For simplification, we assume that the global intercept term a is known and fixed at 0 in the following discussion. All theoretical results are also applicable to the case of unknown a since a can be generalized to a spatially clustered coefficient with only one piecewise constant.

We first introduce the notations used in this section. Use $\mathbf{B}^*$ to denote the true value of $\mathbf{B}$ and $\hat{\mathbf{B}}$ to denote the estimator of $\mathbf{B}$ from Eq. (7). We define a series of index sets $\mathcal{J}, \mathcal{I}_1, \cdots, \mathcal{I}_p$ to indicate the support space of $\mathbf{B}^*$:

$$\begin{aligned} &\mathcal{J} = \{j : \left\|\mathbf{B}_j^*\right\|_2 \neq 0,\ j = 1, 2, \ldots, p\}, \\ &\mathcal{I}_j = \begin{cases} \{i \in \{1, 2, \ldots, n-1\} : \mathbf{H}_{i,j}\mathbf{B}_j^* \neq 0\} & \text{if } j \in \mathcal{J}, \\ \emptyset & \text{if } j \notin \mathcal{J}, \end{cases} \end{aligned} \tag{14}$$

where $\mathbf{H}_{i,j}$ denotes the i^{th} row of $\mathbf{H}_j$. Clearly, $\mathcal{J}$ contains the indexes of truly related variables, and $\mathcal{I}_j$ collects the edges in $\mathcal{G}_j$ that connect locations with different coefficient values on the j^{th} variable. We further define $\tilde{\mathbf{H}}_j = (\mathbf{H}_j^T, \mathbf{1}_n/n)^T \in \mathbb{R}^{n\times n}$. It is easy to check that $\tilde{\mathbf{H}}_j$ is always a full rank matrix. Let $\tilde{\mathbf{H}}_j^{-1}$ denote the inverse matrix of $\tilde{\mathbf{H}}_j$. We decompose $\tilde{\mathbf{H}}_j^{-1} = (\mathbf{H}_j^-, \mathbf{1}_j^-)$ so that $\mathbf{H}_j^- \in \mathbb{R}^{n\times(n-1)}$ contains the first $n-1$ columns of $\tilde{\mathbf{H}}_j^{-1}$, and $\mathbf{1}_j^- \in \mathbb{R}^n$ represents the last column of $\tilde{\mathbf{H}}_j^{-1}$.

To present the theoretical result of our method, we introduce three assumptions:

Assumption 1. *There exists a constant $\gamma > 0$ such that for any $\mathbf{V} \in \mathcal{C}(\mathbf{B}^*; \lambda_1, \lambda_2)$,*

$$\frac{1}{n}\left\|\tilde{\mathbf{X}}\mathrm{vec}(\mathbf{V})\right\|_2^2 \geq \gamma\|\mathbf{V}\|_F^2,$$

where

$$\mathcal{C}(\mathbf{B}^*;\lambda_1,\lambda_2)=\{\mathbf{V}\in\mathbb{R}^{n\times p}:\sum_{j\in\mathcal{J}}\lambda_1\left\|\mathbf{W}_{\mathcal{I}_j^c,j}\mathbf{H}_{\mathcal{I}_j^c,j}\mathbf{V}_j\right\|_1+\sum_{j\in\mathcal{J}^c}\left[\lambda_1\left\|\mathbf{W}_j\mathbf{H}_j\mathbf{V}_j\right\|_1+\lambda_2 u_j\left\|\mathbf{V}_j\right\|_2\right]$$
$$<3\sum_{j\in\mathcal{J}}\left[\lambda_1\left\|\mathbf{W}_{\mathcal{I}_j,j}\mathbf{H}_{\mathcal{I}_j,j}\mathbf{V}_j\right\|_1+\lambda_2 u_j\left\|\mathbf{V}_j\right\|_2\right]\}.$$

$\mathbf{W}_{\mathcal{I}_j,j}$ *denotes the sub-matrix of* $\mathbf{W}_j$ *containing rows corresponding to the indexes in* $\mathcal{I}_j$. $\mathbf{H}_{\mathcal{I}_j,j}$, $\mathbf{W}_{\mathcal{I}_j^c,j}$, *and* $\mathbf{H}_{\mathcal{I}_j^c,j}$ *are defined in the same way.*

Assumption 2. $\epsilon_1,\epsilon_2,\ldots,\epsilon_n$ *are independent sub Gaussian random variables that* $P(|\epsilon_i|>t)\le 2\exp(-t^2/2\sigma_i^2)$.

Assumption 3. $\mathbf{X}$, $\mathbf{W}_j$, *and* μ_j *follow* $\|\mathbf{X}\|_\infty<\infty$, $\max_{j\in\mathcal{J}}\|\mathbf{W}_j^{-1}\|_\infty<\infty$, $\max_{j\in\mathcal{J}}\left\|\mathbf{W}_{\mathcal{I}_j,j}\right\|_\infty<\infty$, $\max_j|u_j^{-1}|<\infty$, *and* $\max_{j\in\mathcal{J}}|u_j|<\infty$.

Assumption 1 requires a restricted strong convexity property at $\mathbf{B}^*$. Assumption 2 requires that the distribution of the error term is not too dispersed. Assumption 3 restricts the ℓ_∞ norms of covariates and weights. All assumptions are commonly adopted in high-dimensional regression model literature (see, e.g., Bühlmann and Van De Geer, 2011; Hastie et al., 2015).

Theorem 2. *Under Assumption 1, for* $\lambda_1/2>\max_{j\in\mathcal{J}}\left\|\frac{1}{n}\boldsymbol{\epsilon}^T\mathrm{diag}(\mathbf{X}_j)\mathbf{H}_j^-\mathbf{W}_j^{-1}\right\|_\infty$ *and* $\lambda_2/2\sqrt{n}>\max_{i,j}|\frac{1}{n}\epsilon_i X_{ij}u_j^{-1}|$,

$$\left\|\hat{\mathbf{B}}-\mathbf{B}^*\right\|_F\le\frac{6(\lambda_1 a_1+\lambda_2 a_2)}{\gamma},\tag{15}$$

and

$$\frac{1}{2n}\left\|\tilde{\mathbf{X}}\mathrm{vec}(\hat{\mathbf{B}}-\mathbf{B}^*)\right\|_2^2\le\frac{18(\lambda_1 a_1+\lambda_2 a_2)^2}{\gamma},$$

where $a_1=\sqrt{|\mathcal{J}|}\max_{j\in\mathcal{J}}\sqrt{|\mathcal{I}_j|}\max_{j\in\mathcal{J}}\left\|\mathbf{W}_{\mathcal{I}_j,j}\right\|_\infty$ *and* $a_2=\sqrt{|\mathcal{J}|}\max_{j\in\mathcal{J}}|u_j|$.

Both a_1 and a_2 include $|\mathcal{J}|$, which shows that the upper bound of the estimating error depends on the number of truly related variables. a_1 also includes $\max_{j\in\mathcal{J}}\sqrt{|\mathcal{I}_j|}$, which is the max number of edges that connect locations with different coefficient values among the variable-dependent graphs.

Under Assumption 2, we can prove that the smallest rates of λ_1 and λ_2 satisfying the requirements of Theorem 2 are $\lambda_1=O(C_1\sqrt{\log(n|\mathcal{J}|)/n})$ and $\lambda_2=O(C_2\sqrt{\log(n)/n})$ as $n\to\infty$, where $C_1=\max_{j\in\mathcal{J}}\|\mathbf{X}_j\|_\infty\max_{j\in\mathcal{J}}\|\mathbf{W}_j^{-1}\|_\infty$ and $C_2=\max_j\|u_j^{-1}\mathbf{X}_j\|_\infty$. Accordingly, we establish Corollary 1 to provide the selection consistency of our estimator.

Corollary 1. *If* $|\mathcal{J}|\max_{j\in\mathcal{J}}|\mathcal{I}_j|=o(n/\log(n))$, *under the Assumptions 1, 2 and 3, there exists* $\delta>0$ *such that*
(i) $\|\hat{\mathbf{B}}_j\|_2<\delta\Leftrightarrow j\notin\mathcal{J}$; *(ii)* $|\mathbf{H}_{i,j}\hat{\mathbf{B}}_j|<\delta\Leftrightarrow i\notin\mathcal{I}_j$, *with probability tending to 1 as* $n\to\infty$.

We now investigate how adaptive learning would improve the convergence rates in Theorem 2 and Corollary 1. First, we consider that $\mathbf{W}_j$ and u_j are built by a consistent estimator of $\mathbf{B}^{\mathrm{ini}}$ following the approach in Section 3.4 with $\alpha=1$. If $\|\mathbf{B}^{\mathrm{ini}}-\mathbf{B}^*\|_F=O(\sqrt{\log(n)/n})$ for example, then $\max_{j\in\mathcal{J}}\|\mathbf{W}^{-1}_{\mathcal{I}_j^c,j}\|_\infty=O(\sqrt{\log(n)/n})$ and $\max_{j\in\mathcal{J}^c}\|u_j^{-1}\|_\infty=O(\sqrt{\log(n)/n})$. Thus, the smallest rate of λ_1 reduces to $O(C_1^*\sqrt{\log(|\mathcal{J}|\max_{j\in\mathcal{J}}|\mathcal{I}_j|)/n})$ to satisfy the requirements of Theorem 2 where $C_1^*=\max_{j\in\mathcal{J}}\|\mathbf{X}_j\|_\infty\cdot\max_{j\in\mathcal{J}}\|\mathbf{W}^{-1}_{\mathcal{I}_j,j}\|_\infty$. Meanwhile, the rate of λ_2 does not change, but $\|\mathbf{X}\|_\infty<\infty$ in Assumption 3 can be relaxed to $\|\mathbf{X}_j\|_\infty<\infty$ for $j\in\mathcal{J}$ and $\|\mathbf{X}_j\|_\infty=O(\sqrt{n/\log(n)})$ for $j\in\mathcal{J}^c$. The results of Corollary 1 will be improved accordingly.

Second, we show the advantage of using the variable-dependent chain graph over the non-adaptive common chain graph. According to the definitions in Eq. (14), the rate of $|\mathcal{J}|$ only depends on $\mathbf{B}^*$, whereas the rate of $|\mathcal{I}_j|$ depends on $\mathbf{B}^*$ as well as $\mathbf{H}_j$ constructed from the adaptive MST-DFS-induced chain $\mathcal{G}_j$. We provide the next Theorem:

Theorem 3. *Denote* R_j *as the true number of piecewise constants in the* j^{th} *variable's spatially clustered coefficient. Suppose that each* $\mathbf{H}_j$ *is constructed from the adaptive MST-DFS-induced chain* $\mathcal{G}_j$ *described in Section 2.2 with the assumptions: (i) The sub-graph of locations with the same coefficient in the* j^{th} *variable is connected in* $\mathcal{G}$; *(ii)* $\mathbf{B}^{\mathrm{ini}}$ *is a consistent estimator of* $\mathbf{B}$. *Then for* $\mathcal{I}_j$ *defined by Eq. (14) with* $\mathbf{H}_j$, $|\mathcal{I}_j|\le 2(R_j-1)$, *and hence* $\max_{j\in\mathcal{J}}|\mathcal{I}_j|\le 2(\max_j R_j-1)$, *with probability tending to 1 as* $n\to\infty$.

Theorem 3 shows that as $n \to \infty$, the value of $|\mathcal{I}_j|$ is only controlled by the true number of piecewise constants in the j^{th} variable's spatially clustered coefficient. Then Corollary 1 holds only if $|\mathcal{J}| \max_{j\in\mathcal{J}} R_j = o(n/\log(n))$. This nice property shows that RSCC could be applied to a large scope of problem theoretically.

The proofs of results in this section are included in the supplementary material.

5. Simulation

For each simulation, we randomly generate 1000 spatial locations $s_1, \ldots, s_{1000}$ from the square domain $[0,1]\times[0,1]$. We then generate $p = 20$ or 100 variables that are not only spatially dependent within each variable but also dependent across variables to mimic many real-world spatial processes. Specifically, the design matrix of p covariates for the simulation is generated from $\mathbf{X} = \mathbf{Z}\boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}\boldsymbol{\Gamma}^T = \boldsymbol{\Sigma}$ with $\boldsymbol{\Sigma}_{ij} = \rho^{|i-j|}$, and $\mathbf{Z} = (z_{ij})_{1000\times p}$ includes p independent realizations generated at 1000 locations from a Gaussian Process GP$(0, C)$ with covariance function $C(z_{i_1 j}, z_{i_2 j}) = \exp(-||s_{i_1} - s_{i_2}||/\phi)$. By this way, the p covariates are correlated according to their indices, and for each covariate, the sample values are correlated according to their locations. The parameters ρ and ϕ affect the strength of the covariate and sample correlation respectively and are set as $\rho = 0.3$ and $\phi = 0.5$. Among p covariates, we randomly select 3 variables with index j_1, j_2 and j_3 as the truly related variables and calculate the response value at the i^{th} location by:

$$y_i = a + \beta_{ij_1}x_{ij_1} + \beta_{ij_2}x_{ij_2} + \beta_{ij_3}x_{ij_3} + \epsilon_i,$$

where ϵ_i is the i.i.d error from $\mathcal{N}(0, 0.1^2)$, and a is the fixed intercept term sampled form $U(-1,1)$. Particularly, β_{ij_1}, β_{ij_2}, and β_{ij_3} are designed with different spatially clustered patterns in the spatial domain: β_{i,j_1} has four piecewise constants with $(1.5, -0.75, 0.75, -1.5)$ as the values of coefficients respectively. β_{i,j_2} has five piecewise constants with $(-0.5, 1, -1.5, -1, 0.5)$ as the values of coefficients respectively. β_{i,j_3} has six piecewise constants with $(-0.5, -1.5, 1, -1, 1.5, 0.5)$ as the values of coefficients respectively. The specific values of β_{ij_1}, β_{ij_2}, and β_{ij_3} for each location are as shown in the first row of Fig. 2.

For model comparison, we consider five compared methods and two oracle methods:

- Elastic-net (EN) (Zou and Hastie, 2005): EN is used as a representative method for constant-coefficient models with variable selection. The penalty parameter and the allocation parameter of EN are determined by BIC. EN becomes the lasso regression when the allocation parameter is tuned as 0.
- SCC (Li and Sang, 2019): The SCC method is a representative method for spatially clustered coefficient models without variable selection.
- Two-step SCC (TS-SCC): Based on the variable selection results of EN, we then use the SCC method to estimate their clustered coefficients.
- GWL (Wheeler, 2009): GWL is a GWR-based variable selection method. It builds a local linear regression with the lasso penalty for each location and achieves location-wise variable selection. We treat a variable as selected by GWL only if it is selected by the model of one location.
- VS-GPSVC (Dambon et al., 2021): VS-GPSVC is a variable selection method for Gaussian process-based SVC models, which achieves jointly selecting the fixed and random effects of variables. To accelerate the computation, we choose the exponential covariance function with a tapering parameter equal to 0.05.
- Oracle$_{\text{VS}}$: Oracle$_{\text{VS}}$ builds the SCC method with known truly related covariates.
- Oracle$_{\text{ALL}}$: Both the truly related covariates and their true clustered patterns are known and the ordinary least square method is used to estimate the coefficients in each clustered pattern.

We use a $K = 5$ nearest neighborhood network built on the Euclidean distance among locations as the original graph $\mathcal{G}$ for the graph-based methods, RSCC, SCC, and TS-SCC. If the constructed network is not connected, we gradually increase K until we get a connected network. For tuning parameter selection, we determine the values of λ_1 and λ_2 in RSCC by selecting the combination with the smallest GIC proposed in Section 3.3 from a 10*10 grid that covers a large range from e^{-8} to e^{0} for both λ_1 and λ_2.

To evaluate the performance of different methods in variable selection and coefficient estimation, we calculate three evaluation metrics: (1) the number of true-positive discoveries (TP); (2) the number of false-positive discoveries (FP); (3) the estimation error between $\hat{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}$ (EE): $\text{EE} = \sqrt{\frac{\left\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right\|_2^2}{np}}$.

To show the efficiency of RSCC, we also record the computational time (Time) of running each method on one core including tuning parameter selection with minute as the unit of time.

Table 1 includes the simulation results. We observe that RSCC achieves better results of all three metrics than those of the five compared methods in both choices of p and could finish in a reasonable time. First, RSCC detects three true variables in all runs and has a low average FP, which demonstrates its superior performance in variable selection. VS-GPSVC performs the second best in variable selection with an FP slightly larger than RSCC. In contrast, EN only selects 2.43 and 1.84 out of 3 truly related variables on average for p = 20 and 100 respectively. The reason that EN fails to select all truly

Table 1
Mean performance (standard deviation) of different methods for variable selection and coefficient estimation in 100 simulations with the variable size $p = 20$ or $p = 100$. The unit of Time is minute.

Method	(n, p) = (1000, 20)				(n, p) = (1000, 100)			
	TP	FP	EE	Time	TP	FP	EE	Time
RSCC	**3.00**	**2.40**	**0.23**	2.44	**3.00**	**5.73**	**0.13**	13.54
	(0.00)	(3.27)	(0.03)	(0.39)	(0.00)	(7.59)	(0.02)	(1.78)
EN	2.43	13.25	0.50	0.00	1.84	48.83	0.22	0.00
	(0.69)	(2.69)	(0.06)	(0.00)	(0.86)	(11.67)	(0.02)	(0.00)
TS-SCC	2.43	13.25	0.39	0.63	1.84	48.83	0.24	1.91
	(0.69)	(2.69)	(0.05)	(0.11)	(0.86)	(11.67)	(0.03)	(0.47)
SCC	3.00	17.00	0.37	0.78	3.00	97.00	0.26	3.96
	(0.00)	(0.00)	(0.04)	(0.02)	(0.00)	(0.00)	(0.03)	(0.07)
GWL	3.00*	17.00*	0.44	6.01	3.00*	97.00*	0.27	26.27
	(0.00)	(0.00)	(0.06)	(0.28)	(0.00)	(0.00)	(0.03)	(2.22)
VS-GPSVC	3.00	3.90	0.32	9.65	2.91	7.49	0.14	520.45
	(0.00)	(1.05)	(0.02)	(0.72)	(0.30)	(2.92)	(0.01)	(7.08)
$\text{Oracle}_{\text{VS}}$	3.00	0.00	0.21	0.14	3.00	0.00	0.10	0.14
	(0.00)	(0.00)	(0.03)	(0.00)	(0.00)	(0.00)	(0.02)	(0.00)
$\text{Oracle}_{\text{ALL}}$	3.00	0.00	0.00	0.00	3.00	0.00	0.00	0.00
	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)

* For GWL, a variable is treated as selected only if it is selected by the model of one location.

related variables is that it does not consider the change of coefficient across locations. The number of FP is also large for EN. GWL has $TP = 3$ and $FP = 17$ for all 100 simulations, which means that every variable is selected by the local linear model of at least one location. Thus, the variable selection results of GWL are not easy to interpret. Second, the EE of RSCC is the smallest among the compared methods and is close to those of $\text{Oracle}_{\text{VS}}$, which demonstrates the effectiveness of RSCC in coefficient estimation. Third, the computational time of RSCC is 2.44 minutes when $p = 20$ and 13.54 minutes when $p = 100$ on average, which is smaller than GWL and VS-GPSVC. We also observe that the computational time of VS-GPSVC increases a lot from 9.65 minutes to 520.35 minutes as p increases from 20 to 100. This result shows that RSCC is an efficient approach and could be used to analyze large spatial data.

Fig. 2 gives an example of the estimated coefficients for different methods with $p = 20$. Overall, RSCC could assign locations with different piecewise constants in the spatially clustered coefficient with different coefficients, and its performance is comparable to that of $\text{Oracle}_{\text{VS}}$. In contrast, SCC without variable selection fails to detect several clusters of regression coefficients for each variable, indicating that including many FP variables in the model may weaken the accuracy of the estimation on the coefficients of truly related variables. TS-SCC performs even worse than SCC as it does not select the third related variable, which shows that a proper variable selection is important to reveal the latent spatial patterns of the coefficients and increase the interpretability of the model. GWL and VS-GPSVC have similar results and can not capture discontinuities in the regression coefficients since both of them assume smoothly varying coefficients, whereas the true regression coefficients in the data generation model are assumed to have spatially clustered patterns. We remark that when true regression coefficients exhibit smoother spatial patterns, GWL and VS-GPSVC might provide more accurate estimations than our method, but they may come with a higher computational cost, especially for large spatial datasets.

Despite its superior performance over the competing methods, we note that RSCC tends to cut a true cluster of observation with the same piecewise constant of a coefficient into several small clusters with different piecewise constants. This phenomenon results from the usage of chain graphs to reduce the computation when clustering the coefficients for large datasets as discussed in Section 2.2. Nevertheless, the number of redundant clusters is not large in general, and as the EE of RSCC is close to $\text{Oracle}_{\text{VS}}$, the estimated values from these redundant clusters are usually close, owing to the use of adaptive variable-dependent weights when constructing chain graphs.

The sensitivity analysis of simulations can be found in the supplementary material. In summary, we find that RSCC is still superior to other methods when the sample or covariate correlation (ϕ or ρ) of spatial covariates is high, and the prediction performance also seems to be robust to the selection of the number of neighbors K used when constructing the K nearest neighborhood graph as the original graph $\mathcal{G}$. We also show that by choosing a relatively large range and grid for (λ_1, λ_2) when finding their best combination via GIC, RSCC could achieve a proper selection of (λ_1, λ_2) with good performance on variable selection and coefficient estimation within a reasonable time.

6. Correlates of COVID-19 vaccination acceptance in the United States

Since early 2020, the COVID-19 pandemic has quickly spread across the world. In December 2020, the United States began a mass vaccination effort after the Food and Drug Administration granted emergency use authorization to the Pfizer-BioNTech vaccine (and shortly thereafter the Moderna and Johnson & Johnson vaccines as well). This mass vaccination effort is widely seen as the most effective population-wide response to COVID-19. However, the effectiveness of this vaccine effort depends crucially on the number of recipients, with herd immunity only possible when enough of the population has been vaccinated (or previously infected). Attempts to achieve widespread vaccine compliance have been complicated

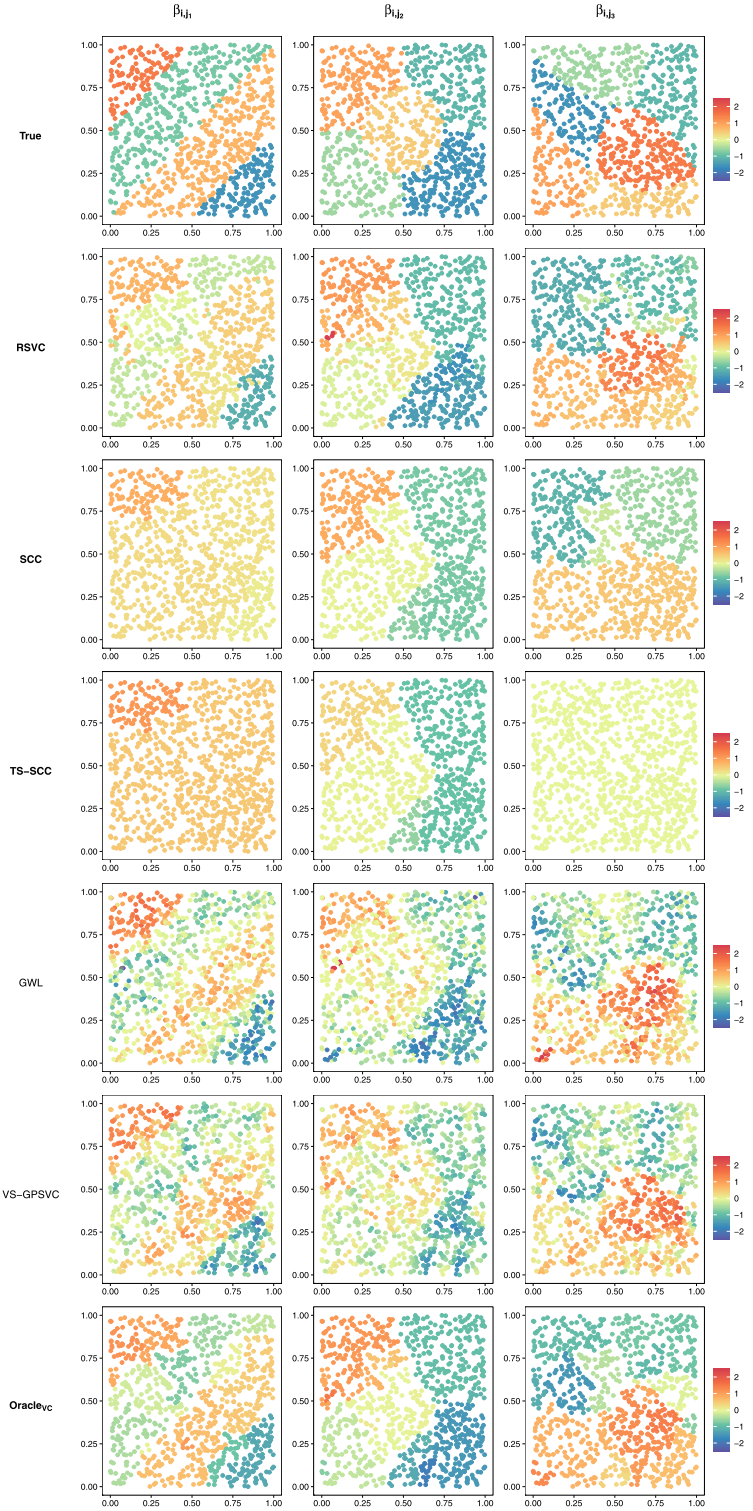

Fig. 2. Estimated coefficients of different methods in one simulation with $(n, p) = (1000, 20)$. β_{i,j_1} has four piecewise constants with $(1.5, -0.75, 0.75, -1.5)$ as the values of coefficients respectively. β_{i,j_2} has five piecewise constants with $(-0.5, 1, -1.5, -1, 0.5)$ as the values of coefficients respectively. β_{i,j_3} has six piecewise constants with $(-0.5, -1.5, 1, -1, 1.5, 0.5)$ as the values of coefficients respectively.

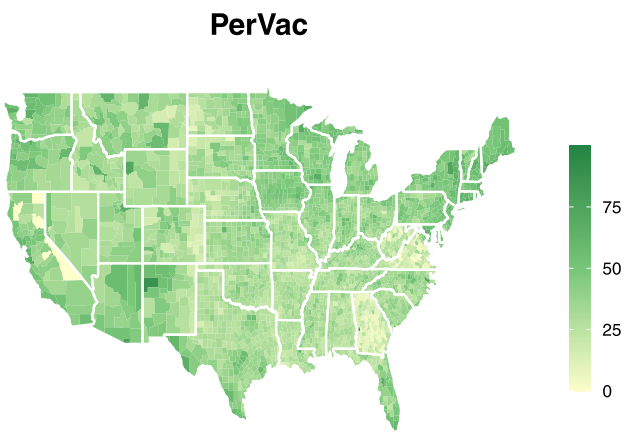

Fig. 3. The percentage of vaccination acceptance (PerVac) of different counties in the United States at July 22, 2021.

by vaccine hesitancy and refusal, often due to personal beliefs or safety concerns within the population (Moghadas et al., 2021). Thus, identifying which social and economic covariates affect the vaccination acceptance of the population and how their effects are become an important subjects of study. While survey research has helped identify reasons why individuals may be hesitant to receive the COVID-19 vaccine, we are also interested in spatial-level patterns of behavior—that is, why cities, counties, and states, vary in their acceptance of (or hesitancy towards) the vaccine. Recently, several studies have shown that COVID-19 vaccination uptake varies systematically across counties. Moreover, this variation is not consistently correlated to expected inputs. For example, Tolbert et al. (2021) find that the correlation between the vaccination rates and COVID-19 impact is not consistent for different regions of the United States, and Mollalo and Tatar (2021) use multiscale GWR to study the varying relationship between vaccine hesitancy and several pre-selected demographic variables.

Building on existing research, we use the RSCC model to both: i) identify the correlates of COVID-19 vaccination acceptance in the United States, and ii) estimate spatial heterogeneity in the effects of these covariates. Our spatial domain is the conterminous United States—that is, we exclude Alaska, Hawaii, and all other islands. As in previous research (Tolbert et al., 2021; Mollalo and Tatar, 2021), we focus on county-level vaccination data in the United States. A county in the United States is a political subdivision of a state that consists of a geographic region with specific boundaries and usually some level of governmental authority. In COVID-19 research, county-level analysis is common given the wide data available at this level (e.g. CDC data), thereby allowing researchers to do nationwide studies with a common unit of analysis (e.g. Chin et al. (2020)). Moreover, counties in different parts of the United States have distinct cultural, social, and economic practices, which may influence how these contextual factors translate into vaccination acceptance among the mass public.

To undertake our analysis we gather data on all 3075 counties (i.e., subdivisions of U.S. states) in the conterminous United States. We collect the percentage of people who have received the full-dose vaccine before July 22, 2021 from the Center for Disease Control and Prevention (CDC), since as of July 2021 U.S. residents had access to COVID-19 vaccinations for several months. Scaling the number of vaccinated people by the total population of each county, we generate our response variable (PerVac) as shown in Fig. 3. As potential inputs, we consider 37 county-level social and economic variables that potentially affect vaccination acceptance. These variables are collected from the 2019 American Community Survey 5-Year Data (ACS5), which reflects the social and economic conditions before COVID-19 spread. We also include the cumulative COVID-19 cases as an additional covariate. The descriptions and pre-processing of variables are given in the supplementary material. By using RSCC, we are able to simultaneously select the subset of variables that are significant determinants of PerVac, find spatially varying relationships between them, and identify potential spatially clustered patterns among counties. In addition to spatially varying coefficients on covariates, we also allow the intercept to vary across space in each of the models to capture the spatial dependence that is unexplained by covariates. To run RSCC, we build a graph $\mathcal{G}$ of counties by their bordering information: If two counties have an adjoining boundary, there is an edge between their corresponding vertices in $\mathcal{G}$. $\mathcal{G}$ is used as the original graph in RSCC to further construct variable-dependent chain graphs.

Table 2 summarizes the range of coefficients for different variables estimated by RSCC. First, 14 variables are selected by RSCC, with 11 of them demonstrating spatially varying coefficient effects. This suggests that models restricting these covariates' effects to a single common parameter neglect important spatial variation in these regression coefficients. By neglecting this variation, these restricted coefficient models necessarily explain less of the variation in the outcome and provide a less complete account of the relationship between the inputs and outcome than can be obtained from our RSCC results. Second, most of the spatially varying coefficients are always in the same direction—Ln_Population, Age65Above, Edu.Bachelor, NoEarningHouseHold, SalaryHouseHold, TotalLaborForce, and Ln_CumulCases positively affect PerVac, and Poverty.Above1.5 negatively affects PerVac across all counties. Interestingly, however, there are three covariates (Race.White, Ln_HouseUnits, and Ln_MedianHouseValue) where the ranges of the coefficients cross zero. For these three variables, researchers using mean coefficients (such as in ordinary least squares) would not only risk neglecting this heterogeneity, but also could in-

Table 2
Range of estimated coefficients of 14 variables selected by RSCC.

Variable	Coefficient
Ln_Population	0.002 ∼ 0.019
Age65Above	0.056 ∼ 0.178
Race.White	-0.015 ∼ 0.046
Race.AmIndian	0.311
Race.Hispanic	0.208
Edu.HighSchool	0.308
Edu.Bachelor	0.685 ∼ 1.003
Poverty.Above1.5	-0.128 ∼ -0.094
NoEarningHousehold	0.469 ∼ 0.542
SalaryHousehold	0.327 ∼ 0.383
TotalLaborForce	0.313 ∼ 0.435
Ln_HouseUnits	-14.396 ∼ 7.432
Ln_MedianHouseValue	-1.330 ∼ 2.280
Ln_CumulCases	1.075 ∼ 1.567

Table 3
Performance of different methods for the vaccination acceptance in counties of the United States.

Metric	RSCC	EN	TS-SCC	SCC	GWL	VS-GPSVC
PE (%)	**6.45**	7.64	7.63	7.64	7.07	6.95
#Selected	**14**	33	33	38	38*	25/12**

* For GWL, a variable is treated as selected only if it is selected by the model of one county.
** VS-GPSVC detects that 25 variables have fixed effects and 12 variables have random effects. Among them, 8 variable have both the fixed and random effects.

correctly conclude that the effects of a covariate to be indistinguishable from zero and fail to include it in the model as a result.

To better represent the spatially clustered patterns of these covariate effects, we plot the estimated coefficients county map for SalaryHousehold, Ln_HouseUnits, and Ln_CumulCases in Fig. 4. There are several noticeable within-state and across-state patterns in the results. For SalaryHousehold (the top map), we observe that the estimated coefficients divide the United States into several regions; the west and mid-west regions as well as the eastern region near North Carolina and Virginia have the highest impact, followed by the northeast region and the south and mid-east region. For Ln_HouseUnits (the center map), we can see how the sign of the coefficient changes across different regions. For example, we observe that in much (though not all) of California there is a largely negative effect, while in other parts of the country (Florida, Georgia, etc.) there is a consistently positive effect. This plot also clearly demonstrates that the latent boundaries of the spatially clustered patterns of coefficients occasionally coincide with state boundary lines (e.g. Texas and Louisiana). Recall that RSCC does not utilize any state-level information, meaning that this result reflects the ability of RSCC to successfully discover abrupt changes across some states, which may be due to state-level variation in vaccination policies. Note as well, however, that these patterns do not strictly conform to state boundaries (e.g. Wyoming has both counties with positive and negative coefficients), which suggests that state-level variables alone would be insufficient to account for the variation observed here, and in our more flexible RSCC model, we are able to identify that parts of some states (e.g. southeast California) may be more similar to counties in neighboring states (e.g. Arizona) in terms of how the inputs affect vaccination uptake. For Ln_CumulCases (the bottom map), we could find that the regions on the east part of the United States have a higher coefficient than the south and the middle parts. All these spatially clustered patterns discovered by RSCC provide new insights for scientists about the relationship between vaccination acceptance and social and economic variables and could motivate future research into systematically explaining the heterogeneity of these covariate effects across counties.

Lastly, since model comparison metrics based on estimation results are not available due to unknown true spatial regression coefficients, we compare the performance of different methods based on prediction although it is not primarily of interest in this areal unit data analysis. Specifically, we randomly select 5% of the counties to serve as the test dataset and use the remaining samples as the training dataset to build a model for each method. Then we estimate the coefficient of each observation in the test dataset by using the mean of its bordering counties' coefficients in the training dataset and predict its PerVac. Finally, the mean squared prediction error (PE) in the test dataset for each method is calculated as the evaluation metric. Table 3 includes PE and the number of selected variables (#Selected) for RSCC and compared methods. RSCC has the smallest PE among all methods while selecting a much smaller number of variables than EN and GWL, indicating that RSCC is capable of achieving superior predictive performance with a much more parsimonious model.

7. Discussion

The paper proposes a new regularized spatially clustered coefficient regression model, called the RSCC model, to select important variables and estimate the spatially clustered coefficients simultaneously. Moving forward, RSCC could be further

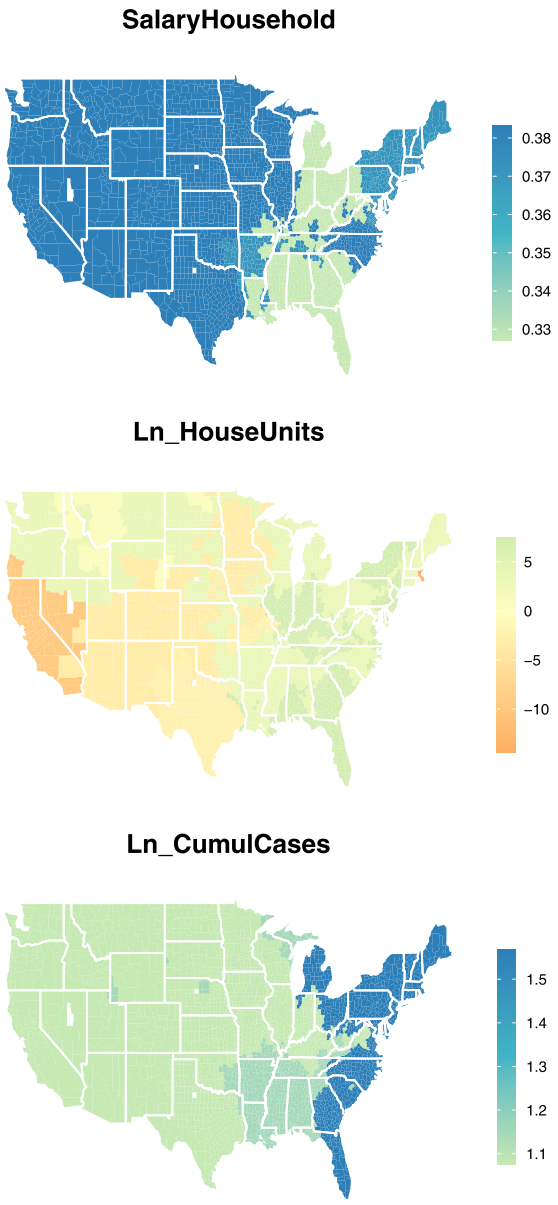

Fig. 4. Real data results: estimated coefficients for SalaryHousehold, Ln_HouseUnits, and Ln_CumulCases. The counties rendered in white have missing values in the covariates and are excluded from our study.

refined in several ways. First, the regularization term of RSCC puts constraints on the size of the coefficients, and hence the results can be sensitive to the method of standardization of covariates. Second, the RSCC estimator does not come with an uncertainty measure, thereby making statistical inference difficult, a common issue shared by regularization-based approaches. We will investigate these research topics in future work.

In practice, RSCC could handle large spatial data with $np \approx 10^6$ on a single core within at most several hours. For a spatial dataset with larger n and p, our current computational method needs to be further improved to handle both storage and computing problems—for example, with the use of parallel computing.

RSCC could also be applied and extended to many other statistical contexts and biological problems. First, one could also easily extend RSCC to the variable selection problems in non-Gaussian outcomes, and multivariate regression models with spatially clustered coefficients. The algorithm and theoretical results in this paper can be applied in a similar way to derive such extensions. Second, RSCC only requires a graph to incorporate the relational information among observations, and thus its usage is not restricted to the spatial problem. For example, RSCC can be extended to study the change of relationship between genes in single-cell RNA sequencing (scRNAseq) data. Cells included in one scRNAseq analysis are always heterogeneous and belong to different cell types. For different cell types, the relationship between the target gene

and other genes varies a lot (Hamey et al., 2017). Fortunately, since the process of cell differentiation is continuous, the gene expression levels of different cells are usually located on a continuous low-dimensional manifold in high-dimensional space, and cells with the same subtype are close in the manifold. Then RSCC can be built with the neighborhood graph of cells in this manifold to find cell clusters and the relevant gene to the target gene simultaneously.

Acknowledgements

Yan Zhong was supported by the National Key R&D Program of China (No. 2021YFA1000100 and 2021YFA1000101). Huiyan Sang is supported by National Science Foundation (No. DMS-1854655 and DMS-2210456). Scott J. Cook was supported by the National Science Foundation (No. DMS-1925119). Scott J. Cook and Paul M. Kellstedt were supported by a College of Liberal Arts COVID-19 Innovation Grant from Texas A&M University.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csda.2022.107581.

References

Arnold, T.B., Tibshirani, R.J., 2016. Efficient implementations of the generalized lasso dual path algorithm. J. Comput. Graph. Stat. 25, 1–27.
Barbero, A., Sra, S., 2018. Modular proximal optimization for multidimensional total-variation regularization. J. Mach. Learn. Res. 19, 2232–2313.
Beck, A., Teboulle, M., 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. 2, 183–202.
Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J., et al., 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends® Mach. Learn. 3, 1–122.
Bühlmann, P., Van De Geer, S., 2011. Statistics for High-Dimensional Data: Methods, Theory and Applications. Springer Science & Business Media.
Chin, T., Kahn, R., Li, R., Chen, J.T., Krieger, N., Buckee, C.O., Balsari, S., Kiang, M.V., 2020. US County-Level Characteristics to Inform Equitable Covid-19 Response. MedRxiv.
Choiruddin, A., Coeurjolly, J.-F., Letué, F., et al., 2018. Convex and non-convex regularization methods for spatial point processes intensity estimation. Electron. J. Stat. 12, 1210–1255.
Condat, L., 2013. A direct algorithm for 1-d total variation denoising. IEEE Signal Process. Lett. 20, 1054–1057.
Dambon, J.A., Sigrist, F., Furrer, R., 2021. Joint variable selection of both fixed and random effects for gaussian process-based spatially varying coefficient models. Preprint. arXiv:2101.01932.
Fan, Y., Tang, C.Y., 2013. Tuning parameter selection in high dimensional penalized likelihood. J. R. Stat. Soc., Ser. B 75, 531–552.
Feng, W., Sarkar, A., Lim, C.Y., Maiti, T., 2016. Variable selection for binary spatial regression: penalized quasi-likelihood approach. Biometrics 72, 1164–1172.
Fotheringham, A.S., Brunsdon, C., Charlton, M., 2003. Geographically Weighted Regression: the Analysis of Spatially Varying Relationships. John Wiley & Sons.
Gelfand, A.E., Kim, H.-J., Sirmans, C., Banerjee, S., 2003. Spatial modeling with spatially varying coefficient processes. J. Am. Stat. Assoc. 98, 387–396.
Hamey, F.K., Nestorowa, S., Kinston, S.J., Kent, D.G., Wilson, N.K., Göttgens, B., 2017. Reconstructing blood stem cell regulatory network models from single-cell molecular profiles. Proc. Natl. Acad. Sci. 114, 5822–5829.
Hastie, T., Tibshirani, R., Wainwright, M., 2015. Statistical Learning with Sparsity: the Lasso and Generalizations. Chapman and Hall/CRC.
Hoefling, H., 2010. A path algorithm for the fused lasso signal approximator. J. Comput. Graph. Stat. 19, 984–1006.
Huang, J., Breheny, P., Ma, S., 2012. A selective review of group selection in high-dimensional models. Stat. Sci. 27.
Johnson, N.A., 2013. A dynamic programming algorithm for the fused lasso and l 0-segmentation. J. Comput. Graph. Stat. 22, 246–260.
Kim, M., Wang, L., 2021. Generalized spatially varying coefficient models. J. Comput. Graph. Stat. 30, 1–10.
Lee, D.-T., 1980. Two-dimensional Voronoi diagrams in the L_p-metric. J. ACM 27, 604–618.
Lee, J., Kamenetsky, M.E., Gangnon, R.E., Zhu, J., 2021. Clustered spatio-temporal varying coefficient regression model. Stat. Med. 40, 465–480.
Li, K., Lam, N.S., 2018. Geographically weighted elastic net: a variable-selection and modeling method under the spatially nonstationary condition. Ann. Assoc. Am. Geogr. 108, 1582–1600.
Li, F., Sang, H., 2019. Spatial homogeneity pursuit of regression coefficients for large datasets. J. Am. Stat. Assoc., 1–21.
Luo, Z., Sang, H., Mallick, B., 2021. A bayesian contiguous partitioning method for learning clustered latent variables. J. Mach. Learn. Res. 22.
Moghadas, S.M., Vilches, T.N., Zhang, K., Wells, C.R., Shoukat, A., Singer, B.H., Meyers, L.A., Neuzil, K.M., Langley, J.M., Fitzpatrick, M.C., et al., 2021. The Impact of Vaccination on Covid-19 Outbreaks in the United States. medrXiv.
Mollalo, A., Tatar, M., 2021. Spatial modeling of covid-19 vaccine hesitancy in the united states. Int. J. Environ. Res. Public Health 18, 9488.
Padilla, O.H.M., Sharpnack, J., Scott, J.G., Tibshirani, R.J., 2018. The dfs fused lasso: linear-time denoising over general graphs. J. Mach. Learn. Res. 18 (176), 1–36.
Reich, B.J., Fuentes, M., Herring, A.H., Evenson, K.R., 2010. Bayesian variable selection for multivariate spatially varying coefficient regression. Biometrics 66, 772–782.
Sass, D., Li, B., Reich, B.J., 2021. Flexible and fast spatial return level estimation via a spatially-fused penalty. J. Comput. Graph. Stat., 1–35.
Shen, X., Huang, H.-C., 2010. Grouping pursuit through a regularization solution surface. J. Am. Stat. Assoc. 105, 727–739.
Shin, Y.E., Sang, H., Liu, D., Ferguson, T.A., Song, P.X., 2019. Autologistic network model on binary data for disease progression study. Biometrics 75, 1310–1320.
Smith, M., Fahrmeir, L., 2007. Spatial bayesian variable selection with application to functional magnetic resonance imaging. J. Am. Stat. Assoc. 102, 417–431.
Thurman, A.L., Fu, R., Guan, Y., Zhu, J., 2015. Regularized estimating equations for model selection of clustered spatial point processes. Stat. Sin., 173–188.
Tolbert, J., Orgera, K., Garfield, R., Kates, J., Artiga, S., 2021. Vaccination is Local: Covid-19 Vaccination Rates Vary by County and Key Characteristics. KFF.
Viallon, V., Lambert-Lacroix, S., Höfling, H., Picard, F., 2013. Adaptive Generalized Fused-Lasso: Asymptotic Properties and Applications.
Wang, H., Leng, C., 2008. A note on adaptive group lasso. Comput. Stat. Data Anal. 52, 5277–5286.
Wheeler, D.C., 2009. Simultaneous coefficient penalization and model selection in geographically weighted regression: the geographically weighted lasso. Environ. Plan. A 41, 722–742.
Zhang, X., Liu, J., Zhu, Z., 2019. Distributed linear model clustering over networks: a tree-based fused-lasso admm approach. Preprint. arXiv:1905.11549.
Zhou, J., Liu, J., Narayan, V.A., Ye, J., 2012. Modeling disease progression via fused sparse group lasso. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 1095–1103.
Zhu, J., Huang, H.-C., Reyes, P.E., 2010. On selection of spatial linear models for lattice data. J. R. Stat. Soc., Ser. B 72, 389–402.
Zou, H., 2006. The adaptive lasso and its oracle properties. J. Am. Stat. Assoc. 101, 1418–1429.
Zou, H., Hastie, T., 2005. Regularization and variable selection via the elastic net. J. R. Stat. Soc., Ser. B 67, 301–320.