Extension of the generalized lasso application in the spatial data analysis

SEPTEMBER 2023

Septian Rahardiantoro

Graduate School of Environmental and Life Science

(Doctor's Course)

OKAYAMA UNIVERSITY

Extension of the generalized lasso application in the spatial data analysis

SEPTEMBER 2023

Septian Rahardiantoro

Abstract

The generalized lasso is a regularization method with great potential to use in spatial analysis. The generalized lasso imposes constraints on the regression coefficients as the ℓ_1 penalty of linear forms, which is represented using a penalty matrix, to include structure or geometry of the coefficients. The different structures of the penalty matrix result in various types of problems and have a variety of applications. The fused lasso, a particular case of the generalized lasso, has been widely applied in spatial data analysis. In this case, the penalty matrix shows a graph structure, so that its each row corresponds to the difference of coefficients between each pair of nodes connected by an edge. It can be used for spatial smoothing and clustering in one- or multi-dimensional neighboring structure of the object, with many applications provided in previous literature. Besides that, the generalized lasso can also be applied in trend filtering, wavelet smoothing, and many more.

In addition, the different predictor matrix in the generalized lasso model, can also lead to various kinds of problems and applications. In fused lasso setting, if the predictor matrix equals to the identical matrix, the problem results in the spatial clustering problem. If the predictor matrix is not identical, the problem becomes spatial modeling. Therefore, because of its flexibility, this study aims to explore and extend the application of generalized lasso in spatial data analysis, especially in spatial clustering and spatial modeling, with some applications to real datasets. Furthermore, this study also considers the methods for selecting optimum tuning parameter of generalized lasso, which controls the trade-off between the goodness-of-fit and the constraints.

In the first part of this study, we conducted some simulation studies to evaluate some methods for selecting optimum tuning parameter of the generalized lasso, namely k-fold cross-validation (k-fold CV), approximate leave-one-out cross-validation (ALOCV), and generalized cross-validation (GCV). We designed the situation of spatial clustering and defined the accuracy measurement of detecting edges with zero differences as an index of edges detection accuracy (IEDA). IEDA combines the sensitivity and positive prediction value (PPV) of detection of the edges with zero differences over several replications. As a result, ALOCV was found to be the recommended method for selecting the optimum tuning parameter compared to k-fold CV, which was suggested by higher IEDA values. If there is only one observation in each location, k-fold CV cannot be feasible, while ALOCV performance in detecting edges with zero difference was appropriate. We also found that GCV can be a substitution for ALOCV when ALOCV is not computable for relatively small tuning parameter or when the noise in the data has large variability since its IEDA performances were almost similar to ALOCV.

The second part of this study discusses about spatial clustering with generalized lasso, which assumed that the predictor matrix is identical. In this part, we provided two applications. In the first application, we examined several methods of selecting the optimum tuning parameter for spatial clustering in the application to Chicago Crime Data, which consists of one or a few observations at each location. As a result, we obtained a suitable result of spatial clustering by using ALOCV, which selected the tuning parameter close to the one suggested by previous literature. In the second application, we extended the application of the generalized lasso for spatio-temporal clustering analysis, to address the issue of determining multiple potential clusters. We proposed a modification of the generalized lasso model adopted for spatio-temporal data, which can be separated into the two generalized lasso problems: trend

filtering on the temporal scale and fused lasso for spatial clustering for each time point. In the trend filtering problem, smoothed temporal pattern is estimated from the average value over all locations at each time point. In the fused lasso problem, in which the average value over all locations has been subtracted from the original responses at each time, clusters are constructed at each time and their relative magnitude can be compared. Therefore, through our proposed method it is possible to see dynamic pattern of clusters as time proceeds. A simulation study was conducted to evaluate the proposed method compared to other approaches in different problems and structures of multiple clusters. The generalized lasso with ALOCV and GCV provided smaller MSE in estimating the temporal and spatial effect compared to unpenalized method, ridge, lasso, and generalized ridge. In temporal effects detection, the generalized lasso with ALOCV and GCV provided relatively smaller and more stable MSE than other methods, for different structure of true risk values. In spatial effects detection, the generalized lasso with ALOCV provided higher IEDA value. The simulation also suggested using a common tuning parameter over all time points in spatial clustering. Then, the proposed method was applied to the weekly Covid-19 data in Japan form March 21, 2020, to September 11, 2021, along with the interpretation of dynamic behavior of multiple clusters. Thus, we obtained information on the clusters of prefectures, and how they are merged or dissolved.

Finally, the last part of this study discusses about generalized lasso for spatial modeling, especially in the case of spatially varying coefficients modeling. The problems in this part assumed that the predictor matrix is not identical. We provided two applications. In the first application, we applied the generalized lasso to fit the spatially varying coefficient model and to cluster regional effects of socio-economics factors that affect the Covid-19 case in Java Island, Indonesia. We considered four numerical socio-economics factors as the predictor variables, namely poverty percentage, Human Development Index (HDI), average of expenditure per month, and Open Unemployment Rate (OUR). In this application, we applied two schemes of grouping regencies: regions by province, and regions defined by K-means clustering of adjacent regencies and Voronoi tessellation. We found that the poverty variable had no effect generally, while the HDI had different effect on the infection rate of Covid-19 between regions on Java Island, that is, the western part of Java Island would have relatively higher HDI effect than the eastern part of Java Island. In the second application, we proposed a generalized lasso with two ℓ_1 penalties to fit a spatially varying coefficient model with numerical and categorical predictor variables. When categorical predictors are involved in the model, the two types of penalties are used: fusion of categories within one categorical predictor in one region, and fusion of adjacent regions for some categories within one categorical predictor. Therefore, in this setting we defined two penalty matrices for pooling regions and for pooling categories. Then, we conducted a simulation study to see the MSE of estimated coefficients and IEDA in the proposed method, compared to existing methods, such as ordinary least square (OLS), generalized ridge, and the generalized lasso with a single penalty matrix either for pooling regions or for pooling categories. We used ALOCV and GCV in the generalized lasso models and LOOCV in generalized ridge for selecting the tuning parameter. Based on our simulation study, our proposed method could estimate coefficients well both for pooling regions and pooling categories, which was suggested by smaller MSE and higher IEDA values, even if the model contains different predictor scales on the model. We also found that the ALOCV and GCV produced similarly good results. As a real data application, the proposed method with ALOCV and GCV were applied in spatially varying coefficient modeling to the house sales price data in Java Island, Indonesia. The predictors consist of two

numerical predictors and seven categorical predictors, of which two categorical predictors were unordered. As a result, ALOCV and GCV provided similar results, and the estimated coefficients for some categories in categorical predictors could be pooled and some of the estimated coefficients could be pooled among provinces.

Contents

1 Introduction

Spatial analysis studies geographical pattern of objects and their relationship using their spatial locations. For example, we may find typical behavior of people in one region compared to the other regions. Or we may generally specify a location at which the infectious disease occurred locally and is likely to be close to the outbreak's source. Thus, location information of the objects, such as their topological, geometric, or geographic properties, is essential in this analysis.

There are many kinds of spatial analysis that have been developed in recent years. Two of them that are quite important are spatial clustering and spatial modeling. Spatial cluster analysis is an essential method for identifying areas with high and low prevalence, which can be used to understand the current conditions and future impacts. For example, it can be applied to the chronic diseases clustering, especially in physical activity and obesity, in U.S. states (Tamura et al., 2014), detecting the air pollution exposure inequities in the United States (Zou et al., 2014), and detecting the cluster of social and environmental inequalities of infant mortality (Padilla et al., 2013). Spatial modeling takes a further step in the analysis, to identify factors that are also suspected of to have a significant effect on the model. For example, it is beneficial to identify the influenced socio-economic factors of Covid-19 cases in Java Island, Indonesia (Rahardiantoro & Sakamoto, 2021), analysis of fungi richness in U.S. states (Zhao & Bondell, 2020), and identify the influenced factors of the house sales price among provinces in Java Island, Indonesia (Rahardiantoro & Sakamoto, 2022b).

Generally, detecting the spatial cluster can be done by evaluating the spatial autocorrelation for the local area (Moran, 1948; Cliff & Ord, 1973; Anselin, 1995) or testing its existence by using some approaches (Kulldorff, 1997; Tango & Takahashi, 2005; Ishioka et al., 2019). Furthermore, spatial regression analysis (Anselin, 1988; Huang, 1984) and geographically weighted regression (Brunsdon et al., 1996) are widely used in spatial modeling. However, some alternative approaches have been proposed in recent years. One method with great potential is the generalized lasso (Tibshirani & Taylor, 2011; Arnold & Tibshirani, 2016). The generalized lasso imposes constraints on the regression coefficients as the ℓ_1 penalty of linear forms to include structure or geometry of the coefficients, and is often applied in spatial data analysis.

The fused lasso is a special case of the generalized lasso and has been widely applied in spatial data analysis. It imposes the ℓ_1 penalty of the difference between coefficients, incorporates the neighborhood structure of the objects flexibly. Basically, we can apply the fused lasso in the case of a one- or multi-dimensional neighboring structure of the object (Tibshirani & Taylor, 2011). For example, a one-dimensional fused lasso can be used for spatial smoothing and hot spot detection for CGH-dataset (Tibshirani & Wang, 2008). A two-dimensional fused lasso can be applied to image denoising, and in the case of irregular graph (Arnold & Tibshirani, 2016; Zhao & Bondell, 2020).

Interestingly, the generalized lasso can be used for both spatial clustering and spatial modeling, depending on the existence of explanatory variables in the case. Therefore, because of its flexibility, this study aims to explore and extend the application of generalized lasso in spatial data analysis, especially in spatial clustering and spatial modeling. Furthermore, this study also considers the methods for selecting optimum tuning parameter of generalized lasso for these applications, because it requires appropriate tuning parameter to obtain the best results.

The chapters are composed as follows: Chapter 2 discusses an overview of the generalized lasso, which consists of its definition, solution to dual problem, and some of its applications. Chapter 3 discusses the methods for selecting the optimum tuning parameter in several cases, along with simulation studies to assess their performances in difference conditions of data. Chapters 4 and 5 show the generalized lasso applications in spatial data analysis in the case of spatial clustering and spatial modeling, respectively. There, a novel application of spatio-temporal clustering analysis will be described in the application to the Covid-19 cases in Japan. Moreover, a new approach to spatial modeling on a spatially varying coefficient model for numerical and categorical auxiliary variables would be described. Finally, Chapter 6 consists of the concluding remarks of this study.

2 The Generalized Lasso

2.1 Definition

Consider a multiple linear regression model $\boldsymbol{y} = \boldsymbol{X\theta} + \boldsymbol{\varepsilon}$, where $\boldsymbol{y} \in \mathbb{R}^n$ is a response vector, $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ is a predictor matrix, $\boldsymbol{\theta} \in \mathbb{R}^p$ is a parameter vector, and $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ is an error vector. In the lasso (Tibshirani, 1996), the estimator of $\boldsymbol{\theta}$ can be obtained by minimizing

$$\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1, \tag{2.1}$$

where $\|\boldsymbol{a}\|_1 = \sum_i |a_i|$ is the ℓ_1-norm and $\|\boldsymbol{a}\|_2 = \sqrt{\sum_i |a_i|^2}$ is the ℓ_2-norm for an arbitrary vector $\boldsymbol{a} = \left(a_1, \dots, a_p\right)^T$ and $\lambda \geq 0$ is a tuning parameter. If we set λ as 0, then equation (2.1) is equivalent to ordinary least square (OLS). If we select a positive λ, the components of $\boldsymbol{\theta}$ are penalized and shrunk toward 0, and some components of $\boldsymbol{\theta}$ are estimated as 0.

The generalized lasso (Tibshirani & Taylor, 2011; Arnold & Tibshirani, 2016) makes constraints on the general structure or geometry in the components of $\boldsymbol{\theta}$ using the penalty matrix $\boldsymbol{D} \in \mathbb{R}^{m\times p}$. The generalized lasso estimator of $\boldsymbol{\theta}$ can be obtained by minimizing

$$\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X\theta}\|_2^2 + \lambda\|\boldsymbol{D\theta}\|_1. \tag{2.2}$$

If we have $\boldsymbol{D} = \boldsymbol{I}$, then the generalized lasso (2.2) becomes the lasso (2.1). For spatial data analysis, the fused lasso on a graph is often used, in which each row of $\boldsymbol{D}$ contains elements -1 and 1 for a pair of adjacent objects, and the remaining elements 0. For instance, the s-th row of $\boldsymbol{D}$ $(s = 1, 2, \dots, m)$ can be expressed as

$$D_s = (0, \dots, -1, \dots, 1, \dots, 0), \tag{2.3}$$

with the j-th element -1 and j'-th element 1 indicating that the j-th and j'-th objects are connected.

2.2 Solution to Dual Problem

Instead of solving (2.2) directly, Tibshirani & Taylor (2011) proposed an algorithm called the dual path algorithm, which actually computes a solution path of the equivalent dual problem of (2.2). For $\boldsymbol{X} = \boldsymbol{I}$, the dual problem of (2.2) is to obtain $\boldsymbol{u} \in \mathbb{R}^m$ minimizing

$$\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{D}^T\boldsymbol{u}\|_2^2 \; s.t \; \|\boldsymbol{u}\|_\infty \leq \lambda, \tag{2.4}$$

where $\|\boldsymbol{u}\|_\infty = \max_{i=1,2,\dots m} |u_i|$. The primal solution of the primal problem (2.2) for given λ is obtained by $\widehat{\boldsymbol{\theta}}_\lambda = \boldsymbol{y} - \boldsymbol{D}^T\widehat{\boldsymbol{u}}_\lambda$ for the solution $\widehat{\boldsymbol{u}}_\lambda$ of (2.4). Similarly, for any general matrix $\boldsymbol{X}$, the dual problem of (2.2) is to obtain $\boldsymbol{u} \in \mathbb{R}^m$ minimizing

$$\frac{1}{2}\|\boldsymbol{X}\boldsymbol{X}^+\boldsymbol{y} - (\boldsymbol{X}^+)^T\boldsymbol{D}^T\boldsymbol{u}\|_2^2 \; s.t \; \|\boldsymbol{u}\|_\infty \le \lambda, \boldsymbol{D}^T\boldsymbol{u} \in \mathrm{row}(\boldsymbol{X}), \tag{2.5}$$

where $\boldsymbol{X}^+$ is the Moore-Penrose inverse of $\boldsymbol{X}$. In this case, the solution $\widehat{\boldsymbol{\theta}}_\lambda$ of the primal problem (2.2) for given λ satisfies $\boldsymbol{X}\widehat{\boldsymbol{\theta}}_\lambda = \boldsymbol{X}\boldsymbol{X}^+\boldsymbol{y} - (\boldsymbol{X}^+)^T\boldsymbol{D}^T\widehat{\boldsymbol{u}}_\lambda$.

The dual path algorithm computes a full solution path of the problem as well as provides the dual solution. The algorithm starts from $\lambda = \infty$ and constructs a solution path to $\lambda = 0$. For each row or coordinate $s = 1, 2, \dots, m$, the "boundary coordinates" is defined as the coordinates such that $\hat{u}_{\lambda,s} = \pm\lambda$, and others as "interior coordinates". Let B denote the set of the boundary coordinates, and v denote the vector of the signs of $\hat{u}_{\lambda,s}$. Then, $\boldsymbol{D}_B$ represents the sub-matrix of $\boldsymbol{D}$ which consists only the rows or coordinates in B, while $\boldsymbol{D}_{-B}$ consists of remaining rows or coordinates except those in B. In the case of $\boldsymbol{X} = \boldsymbol{I}$ with general matrix $\boldsymbol{D}$, the dual solution at l-th iteration for the boundary coordinates is

$$\hat{u}_{\lambda,B} = \lambda v \;\text{ for all } \lambda \in [0, \lambda_l]. \tag{2.6}$$

The dual solution for the interior coordinates at l-th iteration is

$$\hat{u}_{\lambda_l,-B} = (\boldsymbol{D}_{-B}(\boldsymbol{D}_{-B})^T)^+\boldsymbol{D}_{-B}\big(\boldsymbol{y} - \lambda_l(\boldsymbol{D}_B^T v)\big), \tag{2.7}$$

which can be stated more simply as $\hat{u}_{\lambda_l,-B} = a - \lambda_l b$. For any $\lambda \le \lambda_l$, the solution of interior coordinates can be represented as $\hat{u}_{\lambda,-B} = a - \lambda b$ until one hits the boundary. Then, for each interior coordinate s, the "hitting time" t_s^{hit} can be calculated as the solution of $a_s - \lambda b_s = \pm\lambda$, and can be expressed as

$$t_s^{hit} = \frac{a_s}{b_s \pm 1}. \tag{2.8}$$

The next hitting time is hence calculated as

$$T_{l+1}^{hit} = \max_s t_s^{hit}. \tag{2.9}$$

Moreover, the "leaving time" of a boundary coordinate can be calculated as

$$t_s^{leave} = \begin{cases} \dfrac{c_s}{d_s}, & c_s < 0 \text{ and } d_s < 0 \\ 0, & \text{otherwise} \end{cases}, \tag{2.10}$$

where

$$c_s = v_s[\boldsymbol{D}_B(\boldsymbol{I} - (\boldsymbol{D}_{-B})^T(\boldsymbol{D}_{-B}(\boldsymbol{D}_{-B})^T)^+\boldsymbol{D}_{-B})\boldsymbol{y}]_s,$$
$$d_s = v_s[\boldsymbol{D}_B(\boldsymbol{I} - (\boldsymbol{D}_{-B})^T(\boldsymbol{D}_{-B}(\boldsymbol{D}_{-B})^T)^+\boldsymbol{D}_{-B})(\boldsymbol{D}_B)^T v]_s. \tag{2.11}$$

Therefore, the next leaving time is

$$T_{l+1}^{leave} = \max_s t_s^{leave}. \tag{2.12}$$

In summary, the solution path when $\boldsymbol{X} = \boldsymbol{I}$ can be described as in Algorithm 1.

Algorithm 1 Solution Path of the Generalized Lasso when $\boldsymbol{X} = \boldsymbol{I}$.

1. Initiate: $l = 0,\ \lambda = \infty$, $B = \emptyset$, and $v = \emptyset$.
2. While $\lambda_l > 0$:
   a. Compute $\hat{u}_{\lambda_l,-B}$ according to (2.7).
   b. Compute the hitting time according to (2.8) and (2.9).
   c. Compute the leaving time according to (2.10), (2.11), and (2.12).
   d. Let $\lambda_{l+1} = \max\{T_{l+1}^{hit}, T_{l+1}^{leave}\}$. If we obtain $\lambda_{l+1} = T_{l+1}^{hit}$, then B and v are updated by adding the hitting coordinates and its sign respectively, otherwise the leaving coordinates and its sign are removed.
   e. Compute $\hat{u} = \hat{a} - \lambda\hat{b}$ as a solution at $\lambda \in [\lambda_l, \lambda_{l+1}]$. Update $l = l + 1$.

The algorithm above also can be applied in the case of any general matrix $\boldsymbol{X}$, by replacing $\boldsymbol{y}$ and $\boldsymbol{D}$ with $\tilde{\boldsymbol{y}} = \boldsymbol{X}\boldsymbol{X}^+\boldsymbol{y}$ and $\tilde{\boldsymbol{D}} = \boldsymbol{D}\boldsymbol{X}^+$ respectively. If $\mathrm{rank}(\boldsymbol{X}) < p$, the algorithm 1 can still be used by adding an ℓ_2 penalty to the (2.2). Practically, Arnold & Tibshirani (2016) provided fast and stable implementations of the generalized lasso dual path algorithm in the `genlasso` R package, which is fully applied in this study.

2.3 Applications

As mentioned in Chapter 1, the generalized lasso has a wide variety of applications, generally depending on different structures of the penalty matrix $\boldsymbol{D}$ and involvement of the predictor matrix $\boldsymbol{X}$. This is also the reason why highly motivated to learn this method. We discuss some of popular generalized lasso applications.

2.3.1 Fused Lasso

In fused lasso problems (Tibshirani et al., 2005), the ℓ_1 penalty shrinks the coefficients in the neighborhood towards each other. Consider a one-dimensional fused lasso problem, in which objects are ordered and the difference of coefficients between neighbors are penalized. In this case, the penalty matrix $\boldsymbol{D}$ in (2.2) can be expressed as

$$\boldsymbol{D} \in \mathbb{R}^{(p-1)\times p} = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}. \tag{2.13}$$

When $\boldsymbol{X} = \boldsymbol{I}$, this problem is generally applied to spatial smoothing and clustering, which is called as fused-lasso signal approximator (FLSA). The applications are widely found in the case of genomics data, such as in Tibshirani & Wang (2008), Tibshirani & Taylor (2011), and Yang et al. (2016).

Moreover, the fused lasso can also be applied in the case of multi-dimensional structure. Consider the problem of two-dimensional fused lasso on a graph. In this case, each row of penalty matrix $\boldsymbol{D}$ is expressed as (2.3), which corresponds to the difference between nodes connected by an edge. When $\boldsymbol{X} = \boldsymbol{I}$, this problem detects adjacent nodes whose coefficients are estimated to be the same, which can be useful for object clustering. For example, the two-dimensional fused lasso can be applied to regular grid images to denoise the signals on vertically and horizontally adjacent pixels. The two-dimensional fused lasso can be also applied to spatial clustering on irregular graphs, in which adjacent nodes with same coefficient estimates are indicated as a cluster. In addition, for any general matrix $\boldsymbol{X}$, the two-dimensional fused lasso can be applied in the spatially varying coefficient modeling, which allows to have either clustered or separate coefficient estimates according to their locations. In the context of spatial modeling, we consider implementing and extending the spatially varying coefficient modeling to some applications, which was motivated by the study of Zhao & Bondell (2020).

Finally, the fused lasso problem can be also used in the three-dimensional structure of object position, as in human brain image applications using MRI. An example of its application is in Xin et al. (2016) for diagnosis of Alzheimer's disease (AD).

2.3.2 Trend Filtering

In the case $\boldsymbol{X} = \boldsymbol{I}$, if we define $(p-2) \times p$ matrix $\boldsymbol{D}$ as

$$\boldsymbol{D}_{\mathrm{tf},1} = \begin{bmatrix} 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{bmatrix}, \tag{2.14}$$

then the problem is equivalent to linear trend filtering (Kim et al., 2009). For z-th order of trend filtering, the penalty matrix $\boldsymbol{D}_{\mathrm{tf},z}$ can be defined in terms of a composition of individual filters as

$$\boldsymbol{D}_{\mathrm{tf},z} = \boldsymbol{D}\boldsymbol{D}_{\mathrm{tf},z-1} \text{ for } z = 2, 3, \ldots \tag{2.15}$$

where $\boldsymbol{D}$ is the $(p-z-1) \times (p-z)$ version of (2.13). The applications of trend filtering are similar to regression splines and smoothing splines.

2.3.3 Other Applications

Moreover, generalized lasso also can be applied in the wavelet smoothing, the method for signal processing and compression, by solving the lasso problem as minimizing

$$\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{W\theta}\|_2^2+\lambda\|\boldsymbol{\theta}\|_1, \tag{2.16}$$

where $\boldsymbol{W}\in\mathbb{R}^{n\times n}$ is an orthogonal matrix. This problem can be solved by transforming it to the generalized lasso problem as defining $\boldsymbol{\beta}=\boldsymbol{W\theta}$, $\boldsymbol{D}=\boldsymbol{W}^T$, and $\boldsymbol{X}=\boldsymbol{I}$.

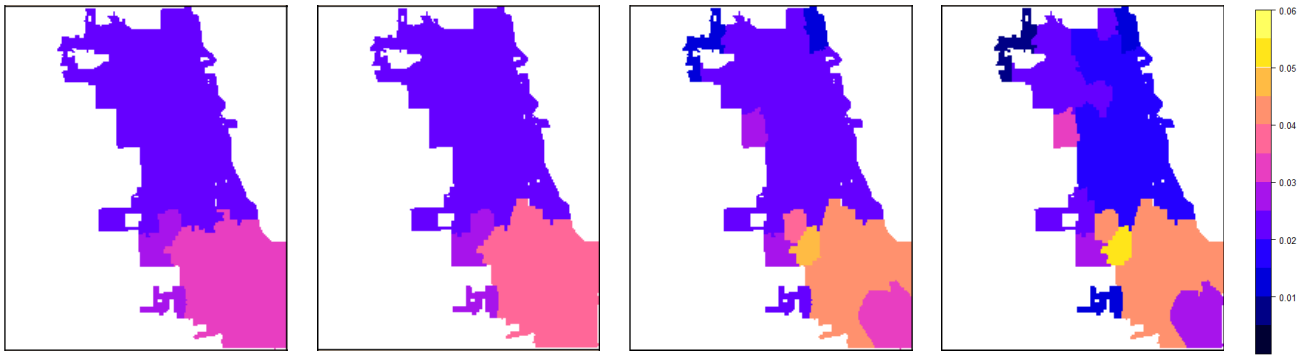
Another application is for outlier detection approach (Tibshirani & Taylor, 2011). Consider a violated observation $\tilde{\boldsymbol{y}}\in\mathbb{R}^n$, with most of them is from the original $\boldsymbol{y}\in\mathbb{R}^n$. To determine which observations are outliers, the generalized lasso can be applied to estimate the coefficient vector $\boldsymbol{\beta}$ by minimizing

$$\frac{1}{2}\left\|\boldsymbol{y}-\tilde{\boldsymbol{X}}\boldsymbol{\beta}\right\|_2^2+\lambda\|\boldsymbol{D\beta}\|_1, \tag{2.17}$$

where $\boldsymbol{\beta}=(\boldsymbol{\alpha},\boldsymbol{\theta})^T$ for $\boldsymbol{\alpha}=\tilde{\boldsymbol{y}}-\boldsymbol{y}$, which suggests the suspected observations as outliers, $\tilde{\boldsymbol{X}}=[\boldsymbol{I}\ \ \boldsymbol{X}]$, and $\boldsymbol{D}=[\boldsymbol{I}\ \ \boldsymbol{0}]$. Then the outliers are determined by the coordinates of $\hat{\boldsymbol{\alpha}}$ that are nonzero, and the estimated coefficients are in $\hat{\boldsymbol{\theta}}$.

3 Tuning Parameter Selection Approaches

One of our crucial themes in this research is to select the optimum tuning parameter λ in the generalized lasso. The choice of inappropriate λ may lead to wrong interpretation of the results. Consider an illustration of clustering problem ($\boldsymbol{X} = \boldsymbol{I}$) in the case study of Chicago crime data (Chicago Police Department, 2014; Arnold & Tibshirani, 2016). Figure 1 shows four different clustering region results with four different λ values $\lambda = \{0.45, 0.25, 0.08, 0.04\}$. It shows that, as λ value gets smaller, the number of regional clusters increases, as indicated by different colors constructed. Therefore, the selection of λ is essential in determining the number of clusters. We consider this case study in more detail in spatial clustering problem in Chapter 4.



(a) (b) (c) (d)

Figure 1. Illustration of clustering regions with generalized lasso for Chicago crime data with (a) $\lambda = 0.45$; (b) $\lambda = 0.25$; (c) $\lambda = 0.08$; (d) $\lambda = 0.04$.

Then, in the following sub-section we will discuss some methods of selecting λ for determining clusters, along with the simulation study to see their performances in various condition of data.

3.1 Some Approaches

A common method that has been often used in selecting λ values is the k-fold cross-validation (CV). However, in some of generalized lasso applications, the k-fold CV may fail to split the data into training and testing data due to the connection between each pair of adjacent objects in the penalty matrix which cannot be separated. Moreover, it is known that k-fold cross-validation produces a biased estimate of the out-of-sample prediction error, especially when both n and p are large and k is small (Rad & Maleki, 2018; Rad et al., 2020). We may reduce the bias issue by applying a large k value and consider using $k = n$ as the special case, which is equal to leave-one-out cross-validation (LOOCV) (Stone, 1974). The LOOCV minimizes the total prediction error when each one observation is predicted as the testing set using the remaining observations as the training set. LOOCV shows better performance, both in numerical and theoretical aspects (Rad et al., 2020). In contrast, for general regularization problems,

LOOCV requires direct computation of the predicted value for each leave-one-out data set, which is computationally expensive. Therefore, the approximate leave-one-out cross-validation (ALOCV) was proposed to reduce computation time ((Rad & Maleki, 2018; Wang et al., 2018). The ALOCV gives an approximation of the leave-one-out predicted values based on the primal and dual formulations of the general regularization problems. Furthermore, in practical computation, we may fail to obtain the ALOCV error for very small λ values. In this case, the generalized cross-validation (GCV) based on the suggestion by Rad & Maleki (2018) can be adopted as an alternative approach. The detail of these methods is described in the following parts.

3.1.1 k-fold Cross-Validation (k-fold CV)

Suppose that we split the data set into k disjoint sets (folds), and let K_c $(c = 1, 2, \dots, k)$ be the set of indices of observations in the c-th fold. The k-fold CV error for a specified λ can be stated as

$$CV_k(\lambda) = \frac{1}{k}\sum_{c=1}^{k}\left(\frac{1}{n_c}\sum_{i\in K_c}\left(y_i - \boldsymbol{x}_i^T\widehat{\boldsymbol{\theta}}_{-c}\right)^2\right), \tag{3.1}$$

Where n_c is the size K_c (indices for the testing set), and $\widehat{\boldsymbol{\theta}}_{-c}$ can be obtained by solving the generalized lasso for the training set, that is, by minimizing

$$\frac{1}{2}\sum_{i\notin K_c}(y_i - \boldsymbol{x}_i^T\boldsymbol{\theta})^2 + \lambda\|\boldsymbol{D\theta}\|_1, \tag{3.2}$$

3.1.2 Leave-One-Out Cross-Validation (LOOCV)

The k-fold CV method requires fitting by (3.2) and computing the prediction error for each division into training and testing sets. The leave-one-out CV is the special case when $k = n$. The leave-one-out CV error for a specified λ can be stated as

$$\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \boldsymbol{x}_i^T\widehat{\boldsymbol{\theta}}_{-i}\right)^2, \tag{3.3}$$

where $\widehat{\boldsymbol{\theta}}_{-i}$ is the solution of the generalized lasso for the leave-one-out sample $\{(x_j, y_j)\}_{j\neq i}$. However, it requires heavy computation for large data sets.

3.1.3 Approximate Leave-One-Out Cross-Validation (ALOCV)

The ALOCV algorithm was proposed by Wang et al. (2018) for general non-differentiable learning problems. In the generalized lasso problem (2.2), for given λ, the ALOCV starts with estimating both primal and dual solutions $\boldsymbol{\theta}$ and $\boldsymbol{u}$, respectively. Then, compute the hat-matrix $\boldsymbol{H}^* = \boldsymbol{A}\boldsymbol{A}^+$, where $\boldsymbol{A} = \boldsymbol{XB}$ for $\boldsymbol{B}$ has columns span the null space of submatrix $\boldsymbol{D}_{-E}$; $E = \{s = 1, 2, \dots, m| \ |\hat{u}_s| = \lambda\}$. The ALOCV error can be obtained by calculating

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \boldsymbol{x}_i^T\widehat{\boldsymbol{\theta}}}{1 - h_{ii}^*}\right)^2, \tag{3.4}$$

where h_{ii}^* is the i-th diagonal component of $\boldsymbol{H}^*$. In detail, the ALOCV algorithm is presented in Algorithm 2.

Algorithm 2 ALOCV for the Generalized Lasso

1. Initiate: $\lambda_l \in \{\lambda_1, \dots, \lambda_L\}$.
2. For each λ_l:
   a. Estimate $\boldsymbol{\theta}$ as a solution of the primal problem (2.2).
   b. Estimate $\boldsymbol{u}$ as a solution of the dual problem according to (2.4) or (2.5).
   c. Construct submatrix $\boldsymbol{D}_{-E}$, where $E = \{s = 1, 2, \dots, m | \, |\hat{u}_s| = \lambda_l\}$.
   d. Construct matrix $\boldsymbol{A} = \boldsymbol{X}\boldsymbol{B}$, where $\boldsymbol{B}$ has columns span the null space of $\boldsymbol{D}_{-E}$.
   e. Compute $\boldsymbol{H}^* = \boldsymbol{A}\boldsymbol{A}^+$.
   f. Calculate the ALOCV error according to (3.4) and save it corresponding to the λ_l.
3. Select λ_l that minimizes the ALOCV error.

3.1.4 Generalized Cross-Validation (GCV)

Practically, we may fail to compute the ALOCV error for very small λ values. Since $h_{ii}^* \to 1$ as $\lambda \to 0$, the denominator $1 - h_{ii}^*$ for some i in (3.4) may become close to zero, and then computation of ALOCV may be unstable. In this case, we adopt the generalized cross-validation (GCV) based on the suggestion by Rad & Maleki (2018), which is to approximate as $h_{ii}^* \approx tr(\boldsymbol{H}^*)/n$, to obtain the following score

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \boldsymbol{x}_i^T\widehat{\boldsymbol{\theta}}}{1 - tr(\boldsymbol{H}^*)/n}\right)^2. \tag{3.5}$$

Then, Algorithm 2 can be applied by changing (3.4) to (3.5) in point 2.f.

3.2 Performance Assessments through Simulation Study

Here, we conduct a simulation study to investigate performance of the tuning parameter selection methods described above in generalized lasso problems. We compare ALOCV and GCV with k-fold CV in the case of spatial clustering ($\boldsymbol{X} = \boldsymbol{I}$).

1	1	1	1	1	1	1	1	1	1
1	1	1	2.5	1	1	1	1	1	1
1	1	2.5	2.5	2.5	1	1	1	1	1
1	2.5	2.5	2.5	2.5	2.5	1	1	1	1
1	1	2.5	2.5	2.5	1	1	1	1	1
1	1	1	2.5	1	1	1	1	1	1
1	1	1	1	1	1	5	5	5	1
1	1	1	1	1	1	5	5	5	1
1	1	1	1	1	1	5	5	5	1
1	1	1	1	1	1	1	1	1	1

Figure 2. The two-dimensional grid simulation data pattern with true $\boldsymbol{\theta}$.

We considered a 10×10 two-dimensional grid as in Figure 2. For each cell $j = 1, \dots, J$ we obtained several observations as $y_{ij} = \theta_j + \varepsilon_{ij}, i = 1, \dots, m$, where the true values $\boldsymbol{\theta} = (\theta_1, \dots, \theta_J)^T$ is shown in Figure 2, and each noise ε_{ij} was generated independently from $\boldsymbol{N}(0, \sigma)$, using the pseudo random number generator of the R software. Based on Figure 2, the true number of edges between adjacent cells (j, j') such that $|\theta_j - \theta_{j'}| = 0$ is $180 - 32 = 148$. We generated 100 replications of the datasets as above.

We considered the following two cases according to the number of observations in each cell. First, when each cell has several observations, we applied the generalized lasso (2.2) for 100 replicated datasets, with λ selected by the five methods: 3-fold CV, 5-fold CV, 10-fold CV, ALOCV, and GCV. We compared the performance of the five methods based on the following four criteria: the minimum CV error, selected λ, the values of the degree of freedom (DF), and the number of edges between adjacent cells (j, j') with zero differences, that is, $|\theta_j - \theta_{j'}| = 0$. We evaluated the degree of freedom (DF) of the model, based on the selected λ. In the case of ℓ_1 penalty, according to Tibshirani & Taylor (2011), the unbiased estimate of the DF for given λ in the generalized lasso (2.2) is defined as,

$$E\big(nullity(\boldsymbol{D}_{-B})\big), \tag{3.6}$$

where $nullity(\boldsymbol{D}_{-B})$ is the dimension of the null space of $\boldsymbol{D}_{-B}$, the reduced rows of the penalty matrix $\boldsymbol{D}$ corresponding to the boundary coordinates set B of a solution of the dual problem (2.4) or (2.5).

Next, when each cell has only one observation, the k-fold CV is not applicable. In this case, we applied the generalized lasso with λ selected by ALOCV and GCV. In both cases, we also computed the $IEDA$ (index of edges detection accuracy) to evaluate the accuracy of detecting edges with zero differences. $IEDA$ can be stated as

$$IEDA = \frac{1}{100}\sum_{z=1}^{100}\frac{2 \times Sens_z^E \times PPV_z^E}{Sens_z^E + PPV_z^E}, \tag{3.7}$$

where $Sens_z^E$ and PPV_z^E indicate the sensitivity and PPV (positive prediction value) to detect the edges with zero differences, respectively, which can be expressed as

$$Sens_z^E = \frac{len\left(\{s: |D_s\widehat{\boldsymbol{\theta}}| = 0\} \cap \{s: |D_s\boldsymbol{\theta}| = 0\}\right)}{len(\{s: |D_s\boldsymbol{\theta}| = 0\})}, \tag{3.8}$$

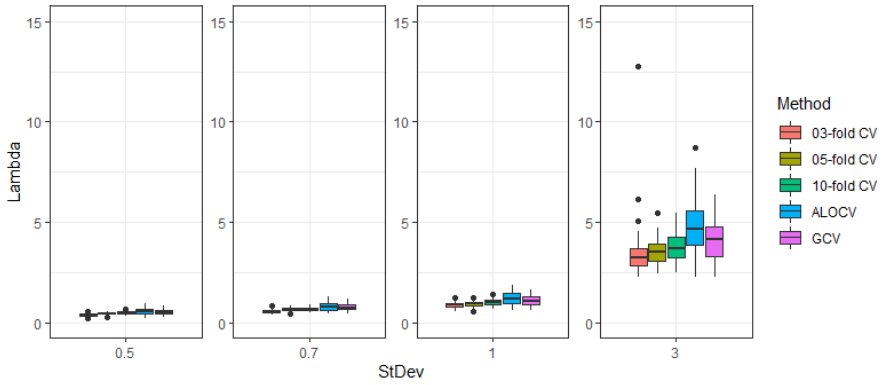
$$PPV_z^E = \frac{len\left(\{s: |D_s\widehat{\boldsymbol{\theta}}| = 0\} \cap \{s: |D_s\boldsymbol{\theta}| = 0\}\right)}{len\left(\{s: |D_s\widehat{\boldsymbol{\theta}}| = 0\}\right)}. \tag{3.9}$$

where len shows the length of a vector, $\{s: |D_s\widehat{\boldsymbol{\theta}}| = 0\}$ is the estimated edges with zero differences, and $\{s: |D_s\boldsymbol{\theta}| = 0\}$ is the actual edges with zero differences. The $IEDA$ value close to 1 means that the estimates can detect edges with zero differences appropriately.

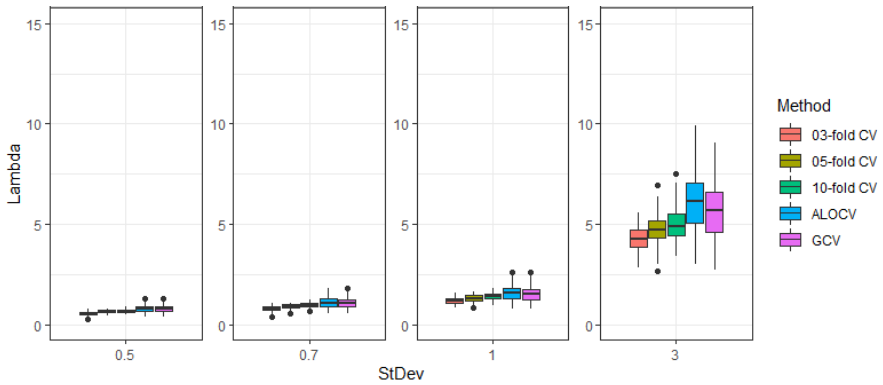
3.2.1 Case of Several Observations in Each Cell

In this case, we considered three kinds of the sample size: $m = 5$, 10 and 20 observations in each cell, thus we have totally $n = 500$, 1000, and 2000 observations, respectively. The tuning parameter λ was searched on the sequence between 0 and 13, and the standard deviation of the noise was set as $\sigma \in \{0.5, 0.7, 1, 3\}$.

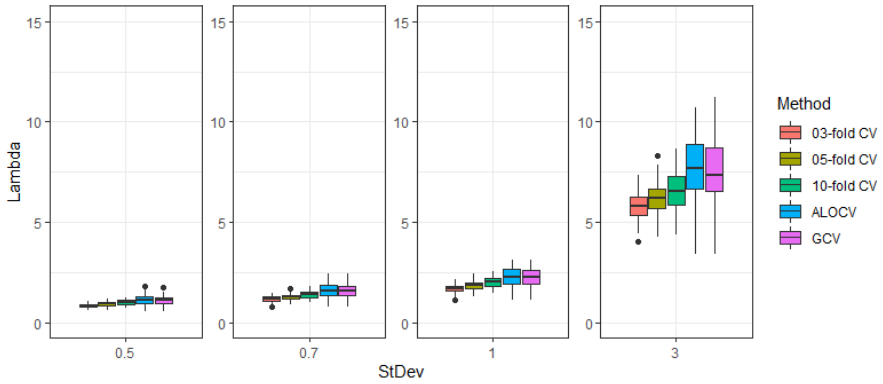
Figure 3 shows the boxplots of selected λ for each σ and n. Based on these results, ALOCV and GCV tended to select higher λ value than k-fold CV, with the highest value obtained in ALOCV, although the range of selected λ for all methods became wider as n increased. The range of selected λ also increased as σ increased. Figure 4 shows the boxplots of DF for each σ and n. DF = 32 is expected. ALOCV and GCV provided smaller DF than k-fold CV, and DF for ALOCV distributed around the number of non-zero edges (32). Figure 5 shows the boxplots of CV error for each σ and n. For all sample sizes n, the error of ALOCV and GCV were slightly smaller than k-fold CV. The range of the CV error for all methods became wider as n increased. Figure 6 shows the boxplots of the number of edges with zero differences of θ's for each σ and n. Here, ALOCV tended to give the largest and closest to the true number (148), which followed by GCV. The range of the number of edges for ALOCV became the widest among all methods when n = 500, while it got narrower as n increased. In contrast, the range of the number of edges for k-fold CV became slightly wider as k and n increase. Table 1 shows the results of $\overline{Sens^E}$ and $\overline{PPV^E}$, the averages of (3.8) and (3.9) for 100 replications, respectively, and the IEDA (3.7). Based on these results, ALOCV tended to have greater IEDA compared to k-fold CV, which followed by GCV. It means that ALOCV selected tuning parameter λ in generalized lasso more appropriately, and if ALOCV is unfeasible, the GCV could be a good alternative. In summary, ALOCV has a good consistency in selecting edges with shrunk difference (Rahardiantoro & Sakamoto, 2022a).



(a)

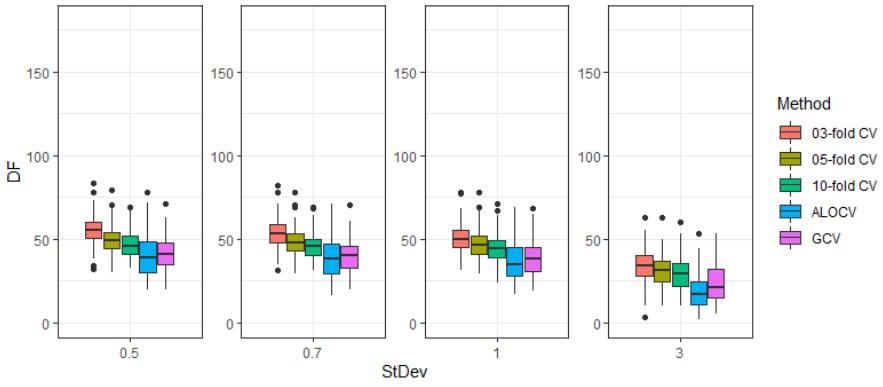


(b)

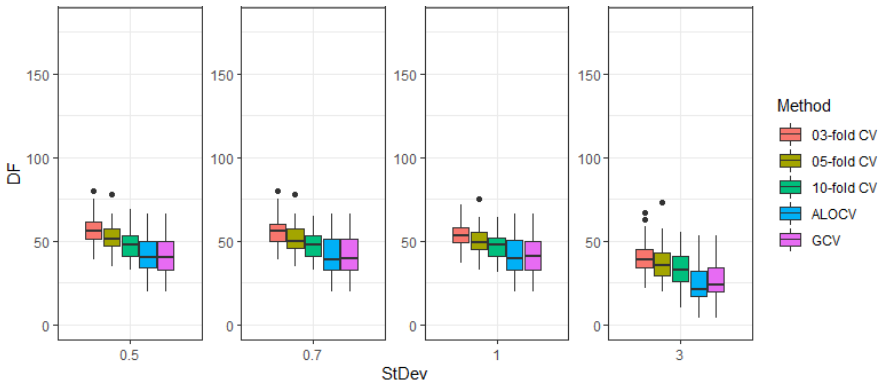


(c)

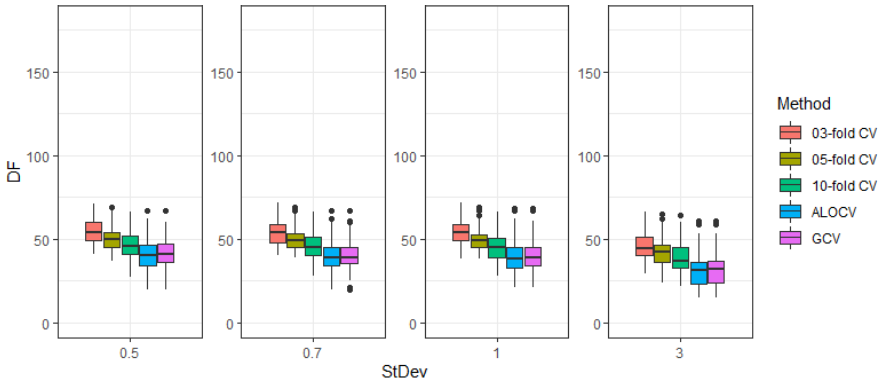
Figure 3. Boxplots of selected λ for (a) $n_1 = 500$, (b) $n_2 = 1000$, and (c) $n_3 = 2000$.



(a)

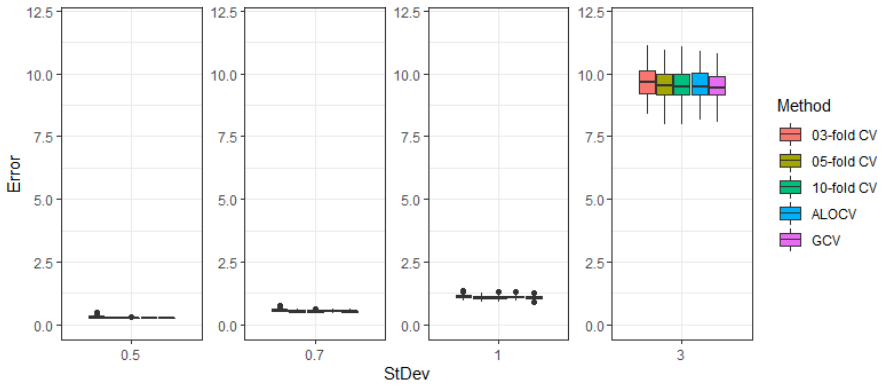


(b)

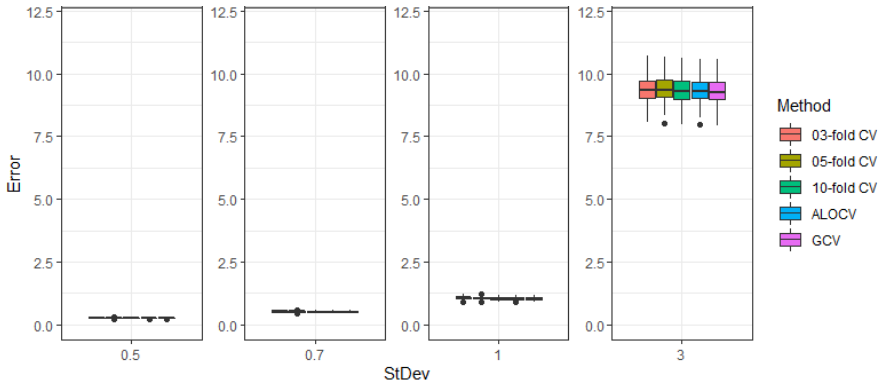


(c)

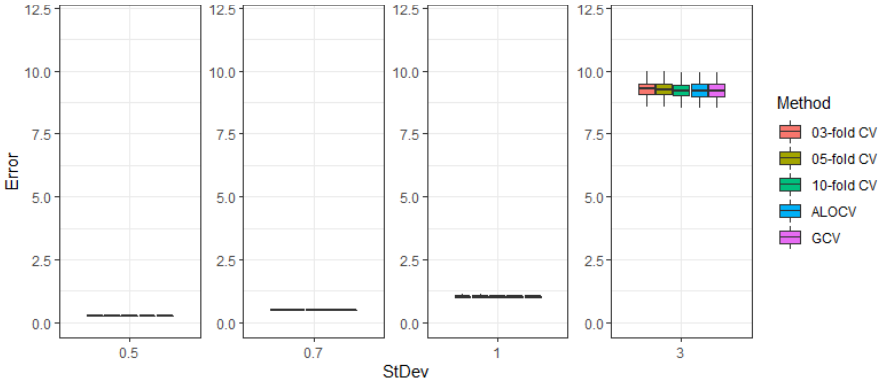
Figure 4. Boxplots of DF for (a) $n_1 = 500$, (b) $n_2 = 1000$, and (c) $n_3 = 2000$.



(a)

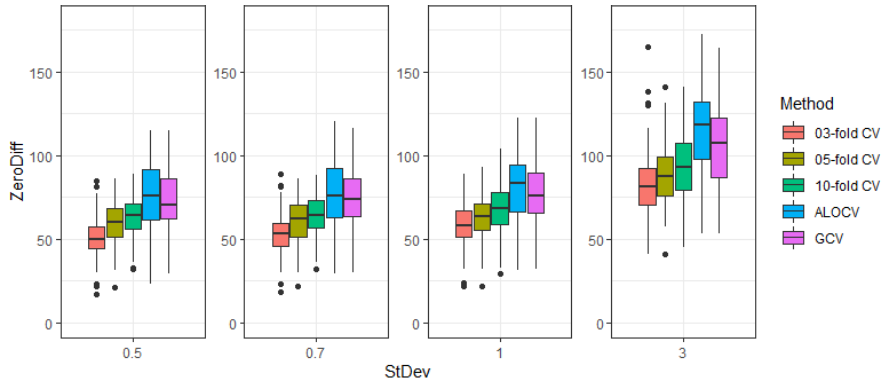


(b)

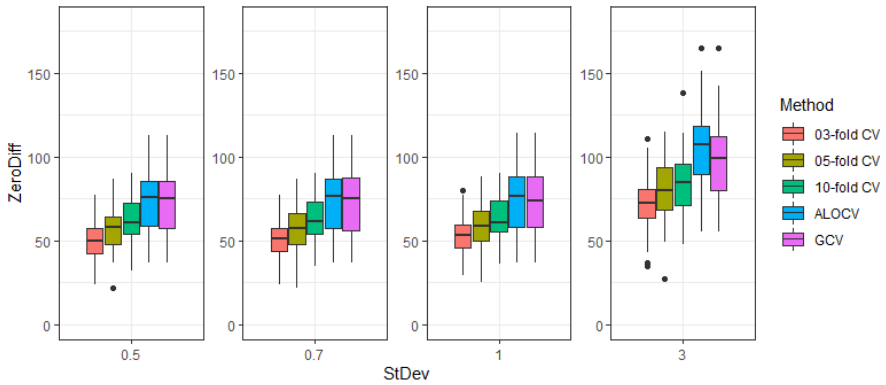


(c)

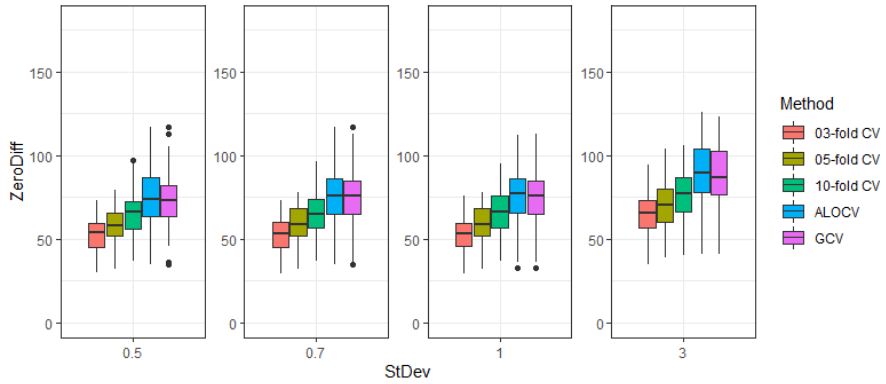
Figure 5. Boxplots of CV error for (a) $n_1 = 500$, (b) $n_2 = 1000$, and (c) $n_3 = 2000$.



(a)



(b)



(c)

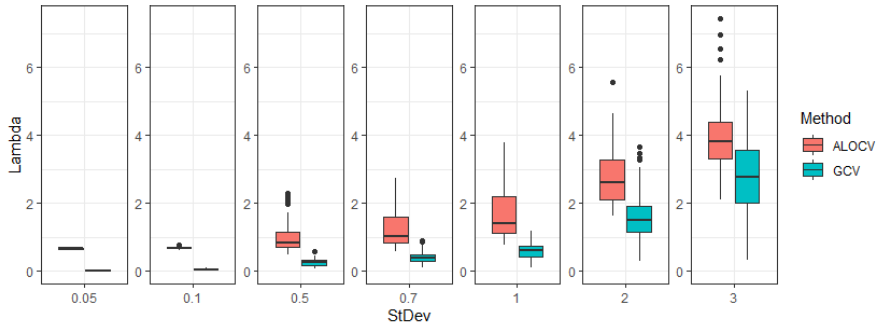
Figure 6. Boxplots of the number of adjacent edges with zero differences of θ's for (a) $n_1 = 500$, (b) $n_2 = 1000$, and (c) $n_3 = 2000$.

Table 1. Result of $\overline{Sens^E}$, $\overline{PPV^E}$, $IEDA$ for 3-fold CV, 5-fold CV, 10-fold CV, ALOCV, and GCV

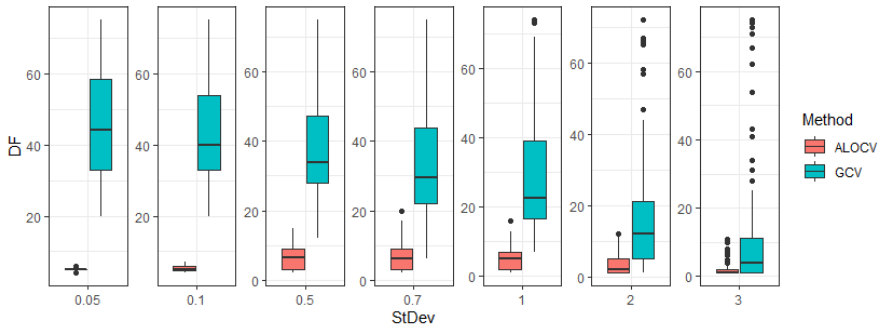
Methods	σ	$n_1 = 500$			$n_2 = 1000$			$n_3 = 2000$		
		$\overline{Sens^E}$	$\overline{PPV^E}$	$IEDA$	$\overline{Sens^E}$	$\overline{PPV^E}$	$IEDA$	$\overline{Sens^E}$	$\overline{PPV^E}$	$IEDA$
3-fold CV	0.5	0.343	1.000	0.505	0.336	1.000	0.498	0.355	1.000	0.520
	0.7	0.361	1.000	0.525	0.342	1.000	0.505	0.358	1.000	0.524
	1	0.390	0.993	0.555	0.357	0.999	0.522	0.361	1.000	0.526
	3	0.512	0.917	0.649	0.457	0.938	0.608	0.427	0.970	0.589
5-fold CV	0.5	0.400	1.000	0.567	0.386	1.000	0.553	0.395	1.000	0.562
	0.7	0.409	1.000	0.575	0.388	1.000	0.554	0.397	1.000	0.565
	1	0.423	0.993	0.588	0.399	0.999	0.565	0.399	1.000	0.566
	3	0.550	0.917	0.681	0.505	0.936	0.649	0.460	0.970	0.619
10-fold CV	0.5	0.426	1.000	0.593	0.422	1.000	0.589	0.438	1.000	0.604
	0.7	0.431	1.000	0.597	0.424	1.000	0.591	0.441	1.000	0.607
	1	0.451	0.993	0.615	0.428	0.999	0.594	0.445	1.000	0.611
	3	0.579	0.913	0.701	0.529	0.934	0.669	0.499	0.970	0.653
ALOCV	0.5	0.507	1.000	0.663	0.493	1.000	0.652	0.502	1.000	0.661
	0.7	0.517	0.999	0.672	0.495	1.000	0.653	0.508	1.000	0.667
	1	0.543	0.992	0.692	0.499	0.999	0.656	0.512	1.000	0.670
	3	0.707	0.907	0.785	0.648	0.925	0.752	0.585	0.965	0.722
GCV	0.5	0.491	1.000	0.651	0.493	1.000	0.651	0.500	1.000	0.659
	0.7	0.504	0.999	0.663	0.494	1.000	0.652	0.509	1.000	0.668
	1	0.514	0.992	0.669	0.495	0.999	0.653	0.510	1.000	0.668
	3	0.647	0.912	0.747	0.614	0.930	0.731	0.573	0.966	0.713

3.2.2 Case of Only One Observation in Each Cell

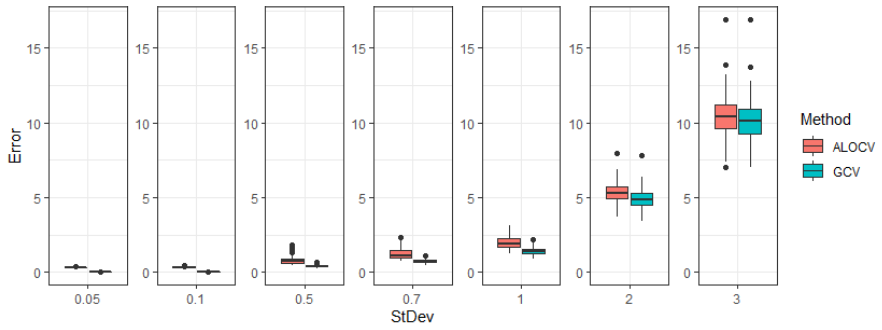
Next, we considered the case of one observation for each cell, so the total sample size was 100. The tuning parameter λ was searched on the sequence between 0 and 4, and the standard deviation of the noise was set as $\sigma \in \{0.05, 0.1, 0.5, 0.7, 1, 2, 3\}$.



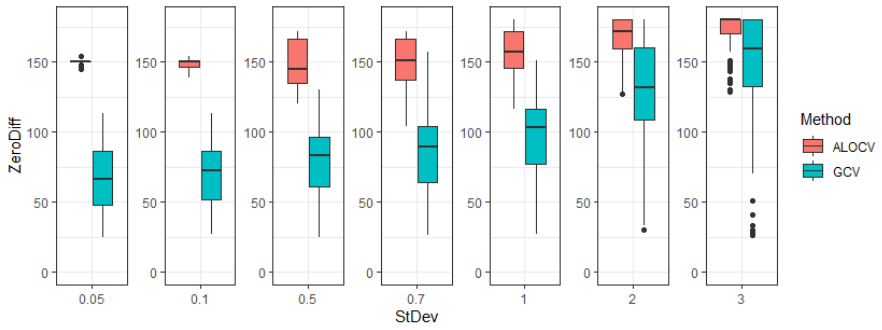
(a)



(b)



(c)



(d)

Figure 7. Boxplots of (a) selected λ, (b) DF values, (c) CV errors, and (d) the number of adjacent edges with zero differences of θ's, for ALOCV and GCV

Table 2. Result of $\overline{Sens^E}$, $\overline{PPV^E}$, $IEDA$ of ALOCV and GCV

σ	ALOCV			GCV		
	$\overline{Sens^E}$	$\overline{PPV^E}$	$IEDA$	$\overline{Sens^E}$	$\overline{PPV^E}$	$IEDA$
0.05	0.934	0.921	0.928	0.452	1.000	0.608
0.1	0.933	0.926	0.929	0.471	1.000	0.625
0.5	0.909	0.911	0.908	0.516	0.984	0.661
0.7	0.907	0.896	0.898	0.545	0.957	0.675
1	0.924	0.874	0.896	0.595	0.928	0.705
2	0.953	0.844	0.893	0.751	0.871	0.785
3	0.965	0.828	0.890	0.818	0.840	0.805

Figure 7 shows the result of the simulation study and Table 2 shows the summary of accuracy on zero difference edges. In Figure 7(a), as the standard deviation of the noise increased, the selected λ tended to be larger, and GCV selected smaller λ than ALOCV. This result was coherent with Figure 7(b), in which GCV provided larger DF than ALOCV. The CV error for GCV was smaller than ALOCV as shown in Figure 7(c). In Figure 7(d), the ALOCV provided the number of zero difference edges very close to the true number (148), while the GCV provided the number of zero difference edges less than the true number, but getting better as σ increases.

Moreover, from the results in Table 2, we can see that ALOCV provided higher $IEDA$ values for smaller σ, and that the $IEDA$ decreased as σ increased. In summary, ALOCV provided the good performances in the detecting edges with zero differences compared to GCV in all of simulation conditions. While, for greater σ, GCV provided the better performances and almost comparable to ALOCV.

4 Spatial Clustering with Generalized Lasso

This chapter discusses the generalized lasso application, especially in the spatial clustering problem. This topic was inspired by a study by Tibshirani & Taylor (2011) and Arnold & Tibshirani (2016), who utilized the generalized lasso for region clustering. In practice, the application of the generalized lasso in the case of spatial clustering does not use explanatory variables in the model, so it is assumed that the matrix $\boldsymbol{X} = \boldsymbol{I}$. We provided two exciting studies on this topic: the region clustering application when the region consists of many observations which applied in Chicago Crime Data (Rahardiantoro & Sakamoto, 2022a) and the spatial-temporal clustering analysis for Covid-19 Cases in Japan (Rahardiantoro & Sakamoto, 2023). We also considered these studies with their optimum tuning parameter selection.

4.1 Spatial Clustering of Chicago Crime Data

4.1.1 Introduction

We applied the generalized lasso for clustering with a spatially varying coefficient problem to Chicago crime data (Chicago Police Department, 2014), which has been also discussed in Arnold & Tibshirani (2016) as an application of fused lasso. The dataset provides the number of burglaries per household over the year 2005 to 2009, spatially aggregated within 2010 census block groups. Although there are totally 2167 blocks in this dataset, we have observations at 2162 blocks (nodes) because 5 blocks with only 1 household have been deleted, and 6995 connections (edges) between neighboring blocks.

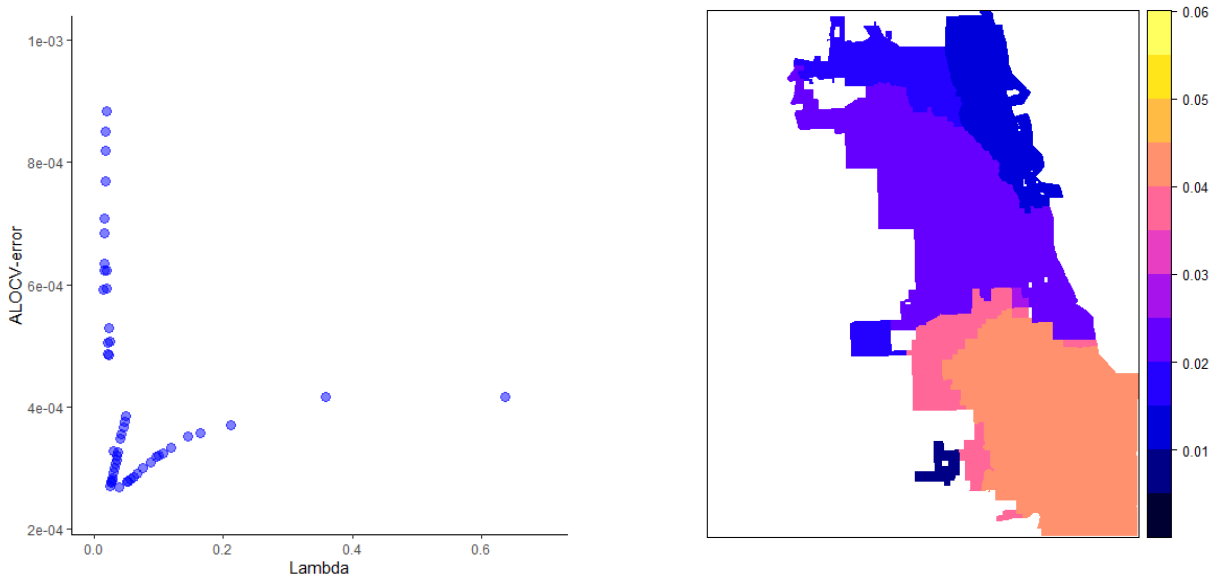
For this case, we consider spatial data with R regions, where the r-th region consists of n_r observations, so that $\sum_{r=1}^{R} n_r = n$. For the purpose of clustering, let $\boldsymbol{X}$ be the R-block diagonal matrix $\tilde{\boldsymbol{I}} = diag\{\mathbf{1}_{n_1}, \dots, \mathbf{1}_{n_R}\}$, where $\mathbf{1}_{n_r} \in \mathbb{R}^{n_r}, r = 1, 2, \dots, R$ is a vector of all ones. It means that (i, r)-th element of $\boldsymbol{X}$ is 1 if y_i belongs to the r-th region, and 0 otherwise. In this case, the number of parameters p is the number of regions R. Therefore, the generalized lasso in this case can be expressed as minimizing

$$\frac{1}{2}\left\|\boldsymbol{y} - \tilde{\boldsymbol{I}}\boldsymbol{\theta}\right\|_2^2 + \lambda\|\boldsymbol{D}\boldsymbol{\theta}\|_1, \tag{4.1}$$

where $\boldsymbol{\theta} \in \mathbb{R}^R$ is a vector of coefficients. The matrix $\boldsymbol{D} \in \mathbb{R}^{m \times R}$ is constructed based on the adjacencies between regions.

4.1.2 Data Analysis

First, we applied the generalized lasso (4.1) to the original dataset directly. In this case $R = n$ and hence $\tilde{\boldsymbol{I}}$ is the diagonal matrix. We can use ALOCV for selecting optimum λ to the dataset with one observation for each node, because it uses only the result of the original fit.



(a) (b)

Figure 8. Fitting the generalized lasso to Chicago crime data (non-clustered). (a) The ALOCV error for the specified λ sequences; (b) Plot of the estimated coefficient at each block, with λ selected by ALOCV

Figure 8 shows the result of using ALOCV to select the optimum λ in fitting the generalized lasso to non-clustered data. In this setting, we have $\boldsymbol{y} \in \mathbb{R}^{2162}$, $\boldsymbol{X} = \boldsymbol{I} \in \mathbb{R}^{2162\times 2162}$, $\boldsymbol{D} \in \mathbb{R}^{6995\times 2162}$. We consider a sequence of 50 tuning parameters λ from 0.01 to 0.7. As shown in Figure 8(a), the optimum λ was selected at 0.0385, and ALOCV error was 2.681×10^{-4}. We estimated the coefficient at each block as shown in Figure 8(b). This result is quite similar with the result described in Arnold & Tibshirani (2016), in which they provided a solution for a particular value $\lambda = 0.037$.

On the other hand, k-fold CV is not feasible in applying generalized lasso to the original dataset, because the elements of $\boldsymbol{\theta}$ are completely separated into those for the training data and the testing data. Therefore, we applied the generalized lasso (4.1) to the dataset clustered into regions in advance, with λ selected by 3-fold CV, 5-fold CV, 10-fold CV and ALOCV. In this setting, we constructed $K = 50$ regions using the K-means method, based on the Euclidian distance between the coordinates of each block census. Then, we used the Voronoi tessellation based on the centroids of these regions to identify the adjacency structure between constructed regions.

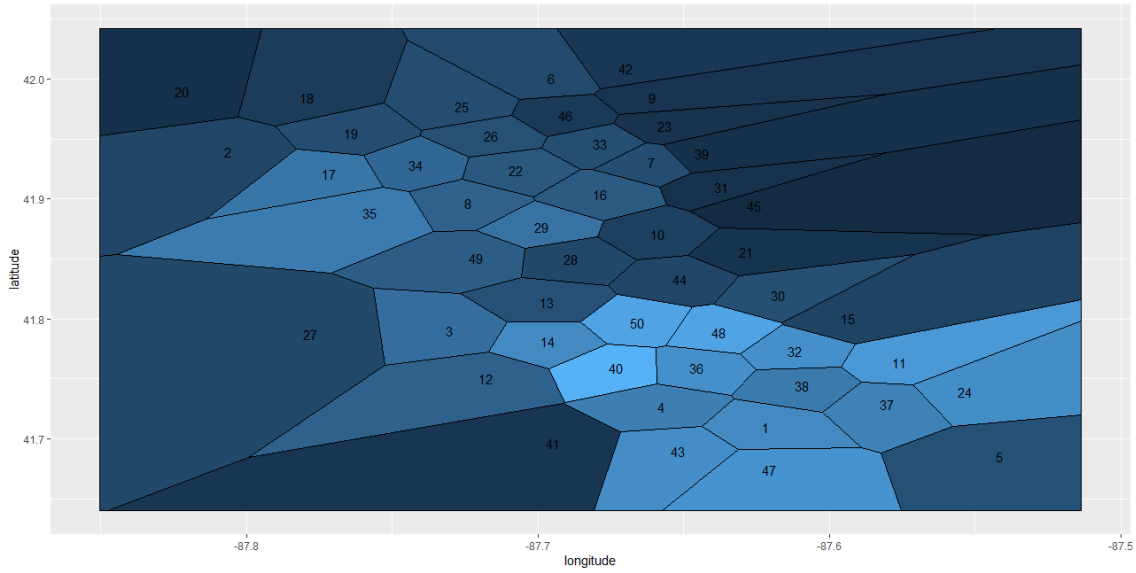


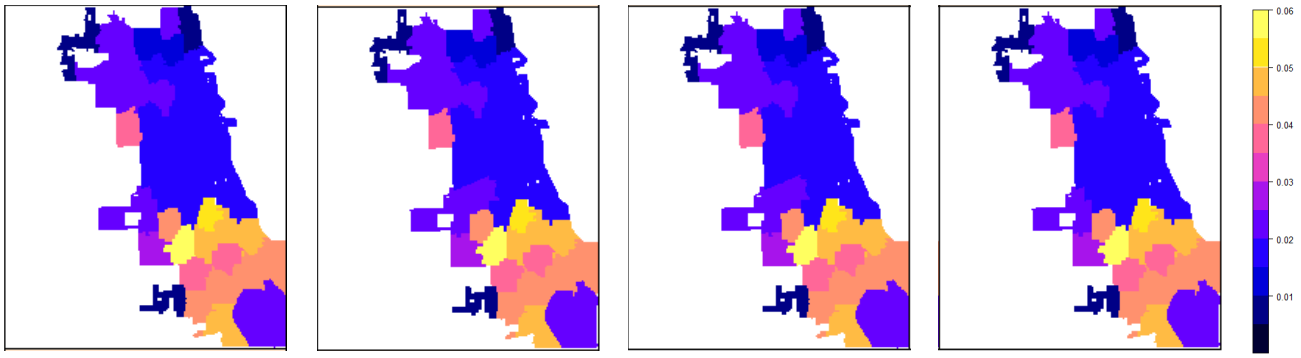
Figure 9. The adjacency of regions constructed by Voronoi tessellation

Figure 9 shows adjacency of regions constructed by Voronoi tessellation. Then, we applied the generalized lasso with formula (4.1) to the data clustered into the regions, with using 3-fold CV, 5-fold CV, 10-fold CV and ALOCV to select λ along to a sequence from 0.01 to 0.7. In this setting, we have $\boldsymbol{y} \in \mathbb{R}^{2162}$, $\boldsymbol{X} = \tilde{\boldsymbol{I}} \in \mathbb{R}^{2162\times 50}$, $\boldsymbol{D} \in \mathbb{R}^{133\times 50}$, since there are 50 regions (nodes) and 133 connections (edges) between regions. Table 3 shows the selected λ and CV errors by each method, and Figure 10 shows the plots of the estimated coefficient for λ selected by each method.

As shown in Table 3, the CV errors obtained by ALOCV is slightly greater than the CV error obtained by k-fold CV. Then, as shown in Figure 10, the plots of the estimated coefficient at each region for λ selected by the four methods were quite similar. However, the estimated coefficients around the southern-east corner for the pre-clustered data were different from those by ALOCV for the original non-clustered data shown in Figure 8(b). We suspect that clustering into 50 regions in advance might have affected the results of estimation and clustering with generalized lasso. Thus, it would be more appropriate to apply the generalized lasso to the original non-clustered Chicago crime data, with using the ALOCV method for selecting λ. In conclusion, we can determine the burglaries risk clusters in Chicago based on the above result, that is, the high-risk cases occurred relatively in the middle to southern side of the city.

Table 3. Selected λ and CV error by each method in fitting the generalized lasso to Chicago crime data clustered into 50 regions

Criteria	Methods			
	3-fold CV	5-fold CV	10-fold CV	ALOCV
λ selected	0.0082	0.0011	0.0011	0.0035
CV error	0.000216	0.000215	0.000213	0.000271



(a) (b) (c) (d)

Figure 10. Fitting the generalized lasso to Chicago crime data clustered into 50 regions: plots of the estimated coefficient at each region, with λ selected by: (a) 3-fold CV; (b) 5-fold CV; (c) 10-fold CV; (d) ALOCV.

4.1.3 Conclusion

The selection of λ by ALOCV for original non-clustered data was reasonable, and the result was in accordance with the conclusion suggested in the preceding literature. In contrast, clustering into regions in advance for making k-fold CV feasible may lead to a wrong result of clustering with a spatially varying coefficient model.

4.2 Spatio-temporal Clustering with an Application of Covid-19 Cases in Japan

4.2.1 Introduction

In the preceding literature on spatio-temporal clustering, ordinary lasso approaches have been mainly used in combination with existing clustering methods. Kamenetsky et al. (2022) proposed the lasso approach to detect the potential cluster using a scan statistic by implementing the sparse matrix representation of the effects of potential clusters. Chen et al. (2018) built separate lasso sub-models at each time point to detect influenced predictors for different historical lags up to 8-time points and included the neighborhood between objects in the specified radius as one of the predictors. However, these methods have limitation in determining multiple potential clusters, because they are highly dependent on the specified radius of the neighborhood.

In this study, we propose a more flexible approach for spatio-temporal clustering, using the generalized lasso framework with two ℓ_1 penalties, in which one penalty corresponds to roughness on the temporal scale, and the other penalty for fusion of adjacent locations at each time point. The proposed model can be separated into the two generalized lasso problems: trend filtering on the temporal scale and fused lasso for spatial clustering at each time point. In the trend filtering problem, smoothed temporal pattern is estimated from the average value over all locations at each time point. In the fused lasso problem, clusters are constructed at each time point and their relative magnitude can be compared. Therefore, our proposed method can reveal dynamic behavior of spatial clusters as time proceeds. One

advantage of our proposed method is its flexibility, that is, we can incorporate adjacencies between objects in the penalty matrix, and it is possible to detect multiple clusters. We evaluate the performance of generalized lasso for detecting the regional cluster compared to existing methods to convince that the generalized lasso can be a good alternative in spatial clustering application through simulation study. We represent a real data application to the Covid-19 cases in Japan.

4.2.2 Proposed Methods: A Spatio-temporal Clustering Analysis with Generalized Lasso

We consider the panel data with locations indexed by $i = 1, 2, \dots, S$ and time points indexed by $t = 1, 2, \dots, T$. So, we have S locations and T time points. In this study, we consider explaining the responses $\boldsymbol{y} = (y_{11}, \dots, y_{ST})^T$ in terms of spatial and temporal effect, which can be expressed as

$$y_{it} = \alpha_t + \beta_{it} + \varepsilon_{it}, i = 1, 2, \dots, S, t = 1, 2, \dots, T, \tag{4.2}$$

where α_t shows the mean temporal effect, and β_{it} shows spatial effect at each time point.

For estimating $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_T)^T$ and $\boldsymbol{\beta} = (\beta_{1t}, \dots, \beta_{St})^T$, we minimize

$$\sum_{t=1}^{T}\sum_{i=1}^{S}(y_{it} - \alpha_t - \beta_{it})^2 + \lambda_T P_T(\boldsymbol{\alpha}) + \sum_{t=1}^{T}\lambda_{S,t} P_S(\boldsymbol{\beta}_t), \tag{4.3}$$

where $P_T(\boldsymbol{\alpha})$ and $P_S(\boldsymbol{\beta}_t)$ indicate the penalty terms of temporal effect and spatial effect, respectively, with corresponding tuning parameters λ_T and $\lambda_{S,t}$. In this case, we use the ℓ_1 penalty term for 1-dimensional trend filtering (Tibshirani & Taylor, 2011)

$$P_T(\boldsymbol{\alpha}) = \sum_{t=3}^{T}|\alpha_t - 2\alpha_{t-1} + \alpha_{t-2}|, \tag{4.4}$$

and the ℓ_1 penalty term for the fused lasso on the graph (Tibshirani & Wang, 2008; Tibshirani & Taylor, 2011)

$$P_S(\boldsymbol{\beta}_t) = \sum_{(i,j)\in\mathcal{E}}|\beta_{it} - \beta_{jt}|, \tag{4.5}$$

where $\mathcal{E}$ is the set of edges on the graph defining adjacency.

To simplify (4.3), we can rewrite the first term as

$$\sum_{t=1}^{T}\sum_{i=1}^{S}(y_{it} - \alpha_t - \beta_{it})^2 = \sum_{t=1}^{T}\sum_{i=1}^{S}\left(y_{it} - \bar{y}_{.t} - \beta_{it} - \bar{\beta}_{.t}\right)^2 + S\sum_{t=1}^{T}\left(\bar{y}_{.t} - \alpha_t - \bar{\beta}_{.t}\right)^2, \tag{4.6}$$

where $\bar{\beta}_{.t} = S^{-1}\sum_{i=1}^{S}\beta_{it}$. If we put the constraint $\bar{\beta}_{.t} = 0$ for identifiability of the parameters, the equation (4.8) can be expressed as

$$\sum_{t=1}^{T}\sum_{i=1}^{S}(y_{it}-\alpha_t-\beta_{it})^2=\sum_{t=1}^{T}\sum_{i=1}^{S}(y_{it}-\bar{y}_{.t}-\beta_{it})^2+S\sum_{t=1}^{T}(\bar{y}_{.t}-\alpha_t)^2, \tag{4.7}$$

Thus, we can express the problem (4.3) as

$$\sum_{t=1}^{T}\sum_{i=1}^{S}(y_{it}-\bar{y}_{.t}-\beta_{it})^2+S\sum_{t=1}^{T}(\bar{y}_{.t}-\alpha_t)^2+\lambda_T P_T(\boldsymbol{\alpha})+\sum_{t=1}^{T}\lambda_{S,t}P_S(\boldsymbol{\beta}_t), \tag{4.8}$$

Therefore, we can solve the problem of minimizing (4.8) as separated minimization on the temporal effect and the spatial effects over time, that is, for estimating $\boldsymbol{\alpha}$ we can only minimize

$$S\sum_{t=1}^{T}(\bar{y}_{.t}-\alpha_t)^2+\lambda_T P_T(\boldsymbol{\alpha}) \tag{4.9}$$

as a 1-dimensional trend filtering problem and for estimating $\boldsymbol{\beta}_t$ $(t=1,2,\dots,T)$, we can only minimize, for each $t=1,..,T$

$$\sum_{t=1}^{T}\sum_{i=1}^{S}(y_{it}-\bar{y}_{.t}-\beta_{it})^2+\sum_{t=1}^{T}\lambda_{S,t}P_S(\boldsymbol{\beta}_t). \tag{4.10}$$

as a fused lasso problem on the graph. In this study, both the problems (4.9) and (4.10) have the form of the generalized lasso, and we can apply the R package `genlasso` (Arnold & Tibshirani, 2016).

4.2.3 Simulation Study

In this simulation study, we investigate the performance of our proposed method with generalized lasso compared to some existing regularization methods. The problem of minimizing (4.3) consists of two penalties, but as explained in the Section 4.2.2, it can be separated into the two generalized lasso problems with each single penalty. Therefore, we compare our proposed methods with the regularization methods which consist of single penalty, such as lasso (Tibshirani, 1996), ridge (Hoerl & Kennard, 1970), and generalized ridge (Zhao & Bondell, 2020). Table 4 shows the corresponding penalties for $P_T(\boldsymbol{\alpha})$ and $P_S(\boldsymbol{\beta}_t)$.

For identifiability issues in lasso and ridge, we make some groups of adjacent time points (or spatial locations), and pool temporal effects α_t (or the spatial effects β_{it}) in the same group. Let $\boldsymbol{\alpha}^*=\left(\alpha_1^*,\dots,\alpha_{T_1}^*\right)^T$ be the vector of pooled temporal effects, $\bar{\boldsymbol{y}}=(\bar{y}_{\cdot 1},\dots,\bar{y}_{\cdot T})^T$, and $\boldsymbol{X}_1\in\mathbb{R}^{T\times T_1}$ be a block-diagonal predictor matrix with a vector ones for each block, representing how the elements are pooled. For example, suppose that we have 12 time points, grouped into $T_1=3$ groups, each containing 4 adjacent time points. In this case, $\alpha_1^*,\alpha_2^*,\alpha_3^*$ are the coefficients for group 1, 2, and 3, respectively, and the predictor matrix can be stated as $\boldsymbol{X}_1=\begin{bmatrix}\mathbf{1}&\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{1}&\mathbf{0}\\ \mathbf{0}&\mathbf{0}&\mathbf{1}\end{bmatrix}$, where $\mathbf{1}=\begin{bmatrix}1\\ \vdots\\ 1\end{bmatrix}$ and $\mathbf{0}=\begin{bmatrix}0\\ \vdots\\ 0\end{bmatrix}$ are vectors of length 4.

Table 4. The penalties $P_T(\boldsymbol{\alpha})$ and $P_S(\boldsymbol{\beta}_t)$ used in the simulation study

Type	$P_T(\boldsymbol{\alpha})$	$P_S(\boldsymbol{\beta}_t)$
Generalized lasso	$\sum_{t=3}^{T}\lvert\alpha_t - 2\alpha_{t-1} + \alpha_{t-2}\rvert$	$\sum_{(i,j)\in\mathcal{E}}\lvert\beta_{it} - \beta_{jt}\rvert$
Generalized ridge	$\sum_{t=3}^{T}(\alpha_t - 2\alpha_{t-1} + \alpha_{t-2})^2$	$\sum_{(i,j)\in\mathcal{E}}(\beta_{it} - \beta_{jt})^2$
Lasso	$\sum_{t=1}^{T_1}\lvert\alpha_t\rvert$	$\sum_{\forall i}^{S_1}\lvert\beta_{it}\rvert$
Ridge	$\sum_{t=1}^{T_1}(\alpha_t)^2$	$\sum_{\forall i}^{S_1}(\beta_{it})^2$

For ridge problems, we can obtain the solution in the close form. Then, the problem of minimizing (4.9) using the ridge penalty for temporal effect is rewritten as

$$S\|\bar{\boldsymbol{y}} - \boldsymbol{X}_1\boldsymbol{\alpha}^*\|_2^2 + \lambda_T\|\boldsymbol{\alpha}^*\|_2^2, \tag{4.11}$$

and the solution of minimizing (4.11) is $\widehat{\boldsymbol{\alpha}^*} = (\boldsymbol{X}_1^T\boldsymbol{X}_1 + \lambda_T\boldsymbol{I})^{-1}\boldsymbol{X}_1^T\bar{\boldsymbol{y}}$.

Similarly, let $\boldsymbol{\beta}_t^* = \left(\beta_{1t}^*, \dots, \beta_{S_1t}^*\right)^T$ be the vector of pooled spatial effect, $\tilde{\boldsymbol{y}}_t$ be the vector of $y_{it} - \bar{y}_{\cdot t}$, and $\boldsymbol{X}_2 \in \mathbb{R}^{S\times S_1}$ be a block-diagonal predictor matrix representing how the elements are pooled. Then, the problem of minimizing (4.10) using the ridge penalty for spatial effect over time is rewritten as

$$\|\tilde{\boldsymbol{y}}_t - \boldsymbol{X}_2\boldsymbol{\beta}_t^*\|_2^2 + \lambda_{S,t}\|\boldsymbol{\beta}_t^*\|_2^2, \tag{4.12}$$

for $t = 1, \dots, T$, and the solution of minimizing (4.12) is $\widehat{\boldsymbol{\beta}_t^*} = \left(\boldsymbol{X}_2^T\boldsymbol{X}_2 + \lambda_{S,t}\boldsymbol{I}\right)^{-1}\boldsymbol{X}_2^T\tilde{\boldsymbol{y}}_t$. For lasso problems, the penalties in (4.11) and (4.12) are replaced with ℓ_1 penalties. In this simulation study, we used the R package "glmnet" to solve the lasso.

In the generalized ridge problem, we can also obtain the solution in the close form as follows. The problem of minimizing (4.9) using the generalized ridge penalty for temporal effect can be expressed in the matrix form as:

$$S\|\bar{\boldsymbol{y}} - \boldsymbol{I}\boldsymbol{\alpha}\|_2^2 + \lambda_T\|\boldsymbol{D}_1\boldsymbol{\alpha}\|_2^2, \tag{4.13}$$

where $\boldsymbol{D}_1 \in \mathbb{R}^{m_1\times T}$ is the penalty matrix forming the second-order difference. The solution for $\boldsymbol{\alpha}$ is written as $\widehat{\boldsymbol{\alpha}} = \left(\boldsymbol{I}^T\boldsymbol{I} + \lambda_T{\boldsymbol{D}_1}^T\boldsymbol{D}_1\right)^{-1}\boldsymbol{I}^T\bar{\boldsymbol{y}}$. In contrast, the problem of minimizing (4.10) using the generalized ridge penalty for spatial effect over time can be expressed in matrix form as:

$$\|\tilde{\boldsymbol{y}}_t - \boldsymbol{I}\boldsymbol{\beta}_t\|_2^2 + \lambda_{S,t}\|\boldsymbol{D}_2\boldsymbol{\beta}_t\|_2^2, \tag{4.14}$$

for $t = 1, \dots, T$, where $\boldsymbol{D}_2 \in \mathbb{R}^{m_2 \times S}$ is the penalty matrix forming the first-order difference on the set of edges $\mathcal{E}$. The solution for $\boldsymbol{\beta}_t$ is written as $\widehat{\boldsymbol{\beta}}_t = \left(\boldsymbol{I}^T\boldsymbol{I} + \lambda_{S,t}{\boldsymbol{D}_2}^T\boldsymbol{D}_2\right)^{-1}\boldsymbol{I}^T\tilde{\boldsymbol{y}}_t$. We also compare the proposed method with the unpenalized estimation in (4.2), that is $\hat{\alpha}_t = \bar{y}_{.t}$ and $\hat{\beta}_{it} = y_{it} - \bar{y}_{.t}$.

We applied LOOCV using the R function "cv.glmnet" to select the tuning parameter in the lasso problem. For ridge and generalized ridge problems, we applied the efficient LOOCV (Meijer, 2010). In the case of minimizing (4.13), an efficient formula of the LOOCV error for given λ_T is represented in a closed form as

$$\frac{1}{T}\sum_{t=1}^{T}\left(\frac{\bar{y}_{\cdot t} - \hat{\alpha}_t}{1 - h_{tt}}\right)^2, \tag{4.15}$$

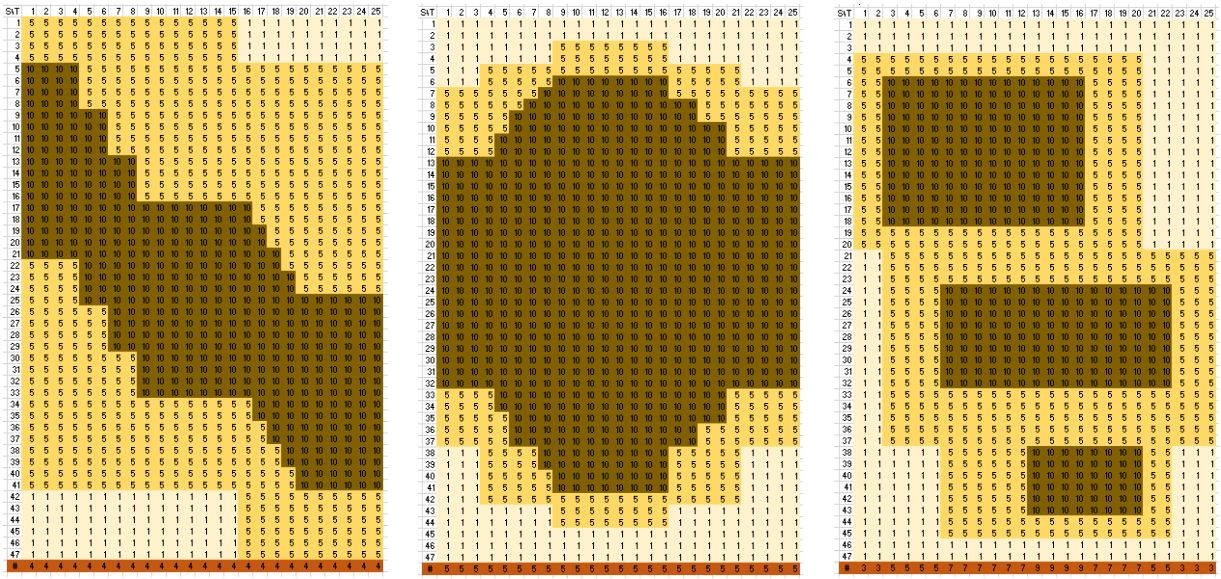
where h_{tt} is the t-th diagonal element of the hat-matrix $\boldsymbol{H}_T = \left(\boldsymbol{I}^T\boldsymbol{I} + \lambda_T{\boldsymbol{D}_1}^T\boldsymbol{D}_1\right)^{-1}$. Then, we select λ_T minimizing LOOCV error (4.15). Similarly, in the case of minimizing (2.14), the LOOCV error for given $\lambda_{S,t}$ can be expressed as

$$\frac{1}{S}\sum_{i=1}^{S}\left(\frac{y_{it} - \bar{y}_{\cdot t} - \hat{\beta}_{it}}{1 - h_{ii}}\right)^2, \tag{4.16}$$

for $t = 1, \dots, T$, where h_{ii} is the i-th diagonal element of the hat-matrix $\boldsymbol{H}_{S,t} = \left(\boldsymbol{I}^T\boldsymbol{I} + \lambda_{S,t}{\boldsymbol{D}_2}^T\boldsymbol{D}_2\right)^{-1}$. We select $\lambda_{S,t}$ minimizing LOOCV error (4.16) for each t, or a common $\lambda_S \equiv \lambda_{S,t}$ minimizing the sum of (4.16) for $t = 1, \dots, T$.

In this simulation study, we assessed the performance of the regularization methods explained above by using the mean square error (MSE) of the coefficients, which indicates the closeness between estimates and the true coefficients. We computed the MSE of the estimated temporal effect and estimated spatial effect. Moreover, to assess the accuracy for detecting clusters in the estimated spatial effect, we used the index of edges detection accuracy ($IEDA$) to evaluate the accuracy of detecting edges with zero differences, that is, zero elements of the vector $\boldsymbol{D}_2\boldsymbol{\beta}_t$.

Because we are motivated by revealing the spread of Covid-19 positive cases in Japan, we constructed data simulating cases for each prefecture in Japan. Japan consists of 47 prefectures, with code 1-47 assigned roughly from north to south, and is grouped into 8 regions: Hokkaido (1), Tohoku (2-7), Kanto (8-14), Chubu (15-23), Kansai (24-30), Chugoku (31-35), Shikoku (36-39), Kyushu & Okinawa (40-47). We suppose that the adjacency between each pair of prefectures is defined based whether they are connected by land, bridges/tunnels, or ocean transportation (National Statistics Center, 2016). We set weekly time points as $T = 25$ to represent about 6 months.



(a) Case 1 (b) Case 2 (c) Case 3

Figure 11. True $\alpha_t + \beta_{it}; i = 1,2, \dots ,47, t = 1,2, \dots ,25$ for each case.
The row label shows the prefecture code, and the column label shows the week.
The last row indicates the number of separated regions for each time point.

We generated new positive cases o_{it} in the i-th prefecture ($i = 1, 2, \dots ,47$) at t-th week ($t = 1, 2, \dots ,25$) from Poisson distribution with mean $\mu_{it} \times N_i$, where $\ln(\mu_{it}) = \alpha_t + \beta_{it} + \varepsilon_{it}$. Where, the noise ε_{it} was generated independently by following a normal distribution with mean 0 and standard deviation 3. N_i is the population in the i-th prefecture, which was obtained form the 2020 Japan's Population Census (Portal Site of Official Statistics of Japan (e-Stat), 2021). We define three cases of the true $\alpha_t + \beta_{it}$ with values 1, 5, and 10 as shown in the Figure 11, to represent different problems and structures of clusters as follows.

a. The Case 1 represents that one aggregated region of higher risk moves as time goes by as in Figure 11(a). In this case, we simulated the cluster of prefectures in Tohoku, Kanto, and Chubu Regions, which have a higher risk steady for four weeks. Then, the higher risk region moves to southwest prefectures within four weeks and become steady on most prefectures in Chubu, Kansai, and Chugoku Regions for eight weeks. After that, the higher risk region moves again to southwest prefectures within four weeks and become steady in Kansai, Chugoku, Shikoku, and Kyushu Regions for remaining five weeks.
b. The Case 2 represents that one aggregated region of higher risk increases and decreases in size as in Figure 11(b). In the first four weeks, the higher risk region keeps steady on several prefectures in Kanto, Chubu, Kansai, and Chugoku Regions. Then, the higher risk region spreads to other surrounding prefectures within four weeks until it becomes steady up to seven regions: Tohoku, Kanto, Chubu, Kansai, Chugoku, Shikoku, and Kyushu Regions within eight weeks. After that, the higher risk region decreases for four weeks, and then returns to the initial size and keeps steady for five weeks.

c. The Case 3 represents that several aggregated regions of higher risk appears and disappears as in Figure 11(c). For the first two weeks, there are no region of higher risk. Then, a higher risk region appears on prefectures in Tohoku, Kanto, and Chubu Regions for 14 weeks. In week 7, the second higher risk region appears on prefectures in Kansai and Chugoku Regions for 16 weeks. Meanwhile, the third higher risk region appears on prefectures in Shikoku and Kyushu Regions from week 13 for 8 weeks.

The regions of adjacent prefectures with higher risk value means that they are clustered. The last row of Figure 11 shows the number of regions separated by adjacency of prefectures and different level of $\alpha_t + \beta_{it}$.

For each case 1, 2, and 3, 100 data sets were replicated. Then, for each data set, we transformed as $y_{it} = \ln\left(\frac{o_{it}}{N_i}\right)$, and fitted several models explained above. For generalized lasso/ridge, we defined the second-order difference penalty matrix $\boldsymbol{D}_1 \in \mathbb{R}^{23\times 25}$ for temporal effect. Moreover, the definition of adjacency between prefectures detects 93 edges, from which we obtained the penalty matrix $\boldsymbol{D}_2 \in \mathbb{R}^{93\times 47}$ for spatial effect. For lasso and ridge, we pooled 4 successive coefficients for temporal effect to obtain $\alpha_1^*, \dots, \alpha_6^*$ (the last one covers five weeks) and pooled coefficients based on 8 prefectural regions of Japan to obtain $\beta_{1t}^*, \dots, \beta_{8t}^*$, so that $T_1 = 6$, $S_1 = 8$, $\boldsymbol{X}_1 \in \mathbb{R}^{25\times 6}$ and $\boldsymbol{X}_2 \in \mathbb{R}^{47\times 8}$.

Figure 12 shows the MSE for the estimate of the temporal effect α_t. In Case 1, the lasso had the smallest MSE than other methods, followed by the generalized lasso with ALOCV and GCV. The true temporal effect α_t was almost constant, and so we guess that the lasso estimates of pooled coefficients might be more advantageous. In Case 2, the generalized lasso with ALOCV mainly provided the smallest MSE, followed by the generalized lasso with GCV. In this case, the MSEs of lasso and ridge were fluctuated highly, especially at the points where true risk values changed. In Case 3, although the MSE of lasso had the smallest value for several weeks, but the generalized lasso with GCV's MSE was the most stable, followed by the generalized lasso with ALOCV. The MSEs of lasso and ridge were also fluctuated when the number of clusters increased or decreased. We guess that pooling of the coefficients might have caused poor performance of lasso and ridge. Generally, in the temporal effect estimation, the generalized lasso with ALOCV and GCV provided relatively smaller and more stable MSE than other methods, for different pattern of true risk values.

Figure 13 shows the average of MSE for the estimates of the spatial effect β_{it} over 47 prefectures for each time point. In the case of using a common tuning parameter λ_S (Figure 13(a)), the generalized lasso with GCV mainly provided the minimum MSE in all cases, followed by the generalized lasso with ALOCV. In Case 1, when the true risk values were steady in weeks 9 to 16, the MSE of lasso was smaller than the generalized lasso with ALOCV. However, in Case 2 and Case 3, when the true risk values changed, the MSEs of other methods were relatively higher than the generalized lasso. The result in the case of using a different tuning parameter $\lambda_{S,t}$ (Figure 13(b)) was slightly different. Mainly, the generalized ridge provided the smallest MSE. The MSEs of the generalized lasso with GCV and ALOCV was higher at several intermediate weeks in Case 1, and at the first and last several weeks in Case 2, but were smaller than lasso and ridge in other cases.

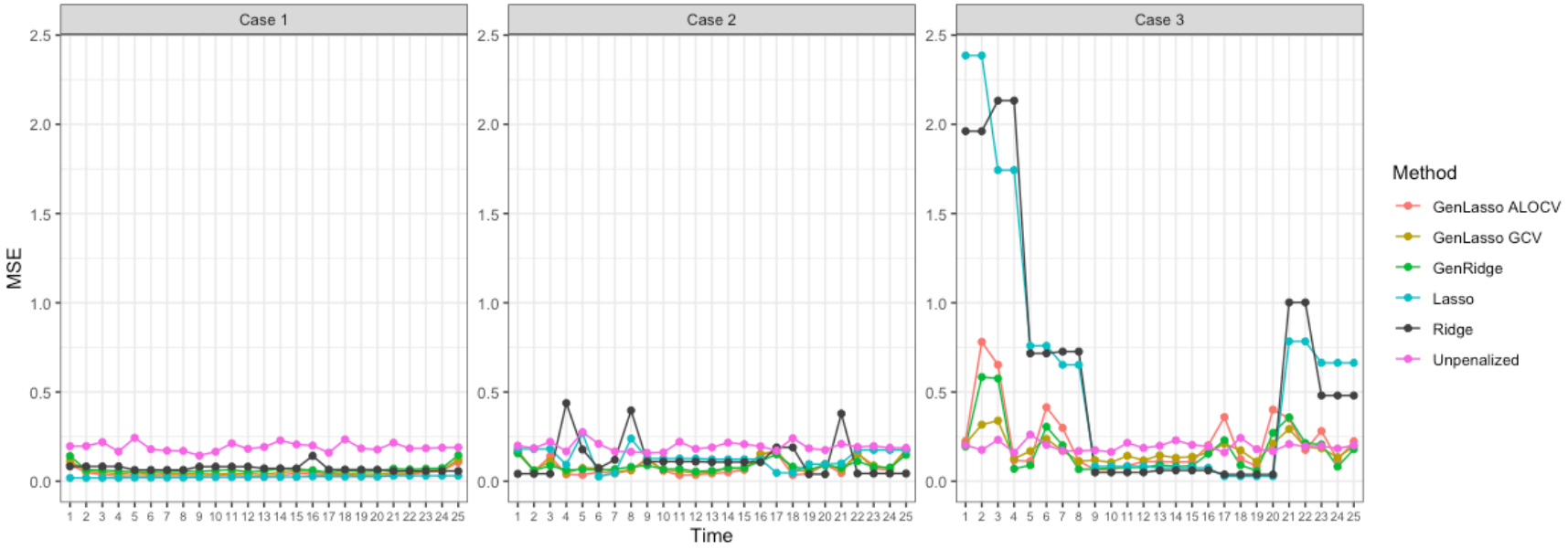
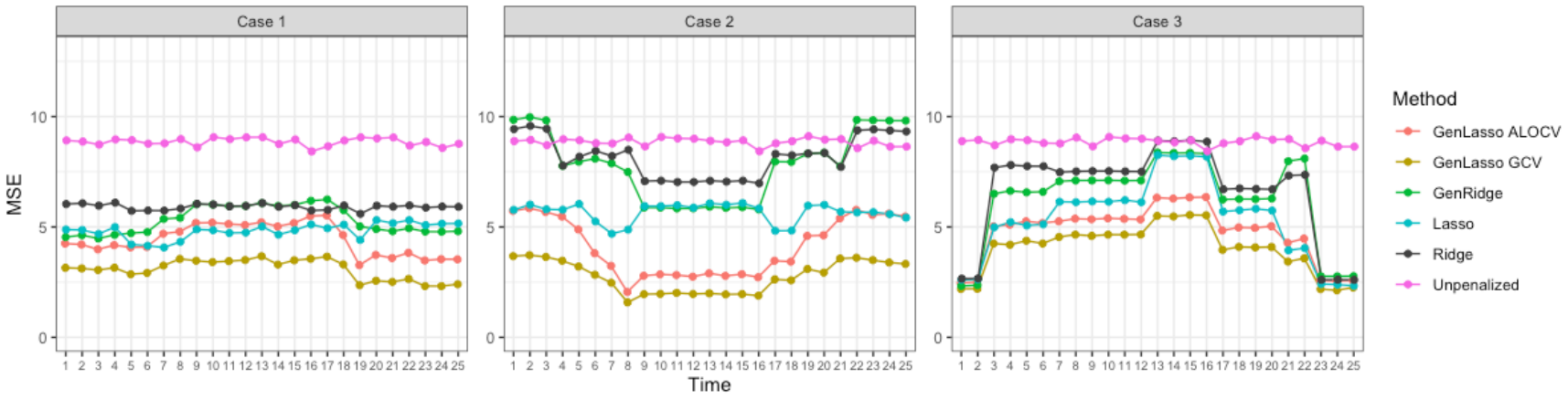
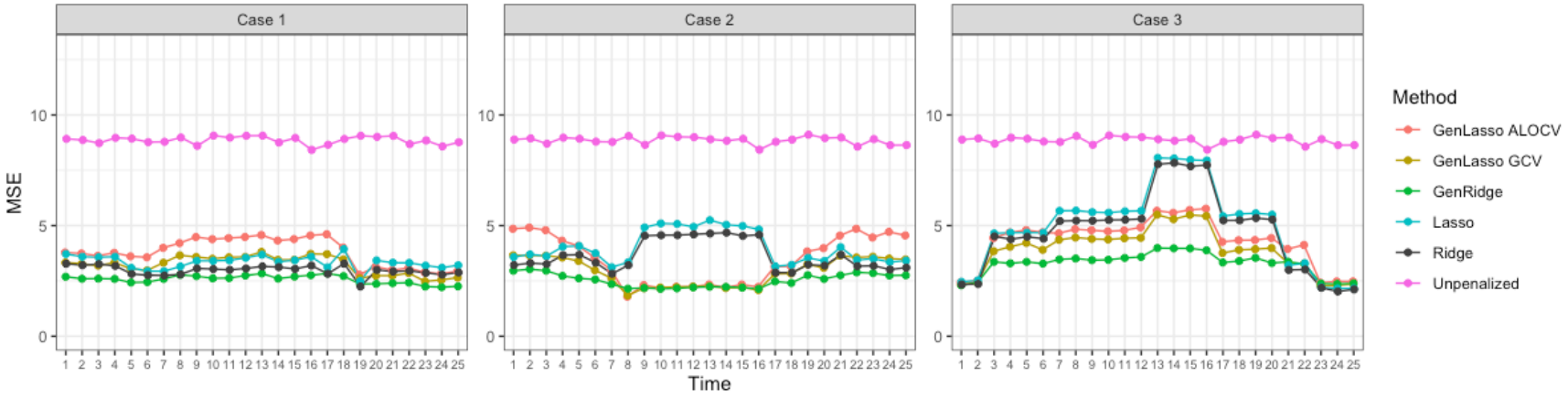


Figure 12. Line-plots of MSE for coefficient α_t each time point from 100 replications. The Case 1, 2, and 3 correspond to the true $\alpha_t + \beta_{it}$ displayed in Figure 11.

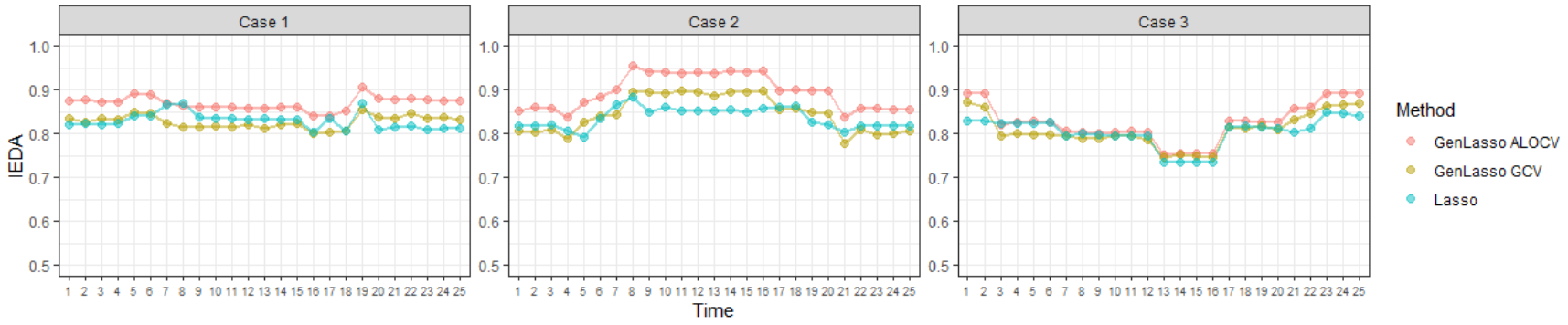


(a)

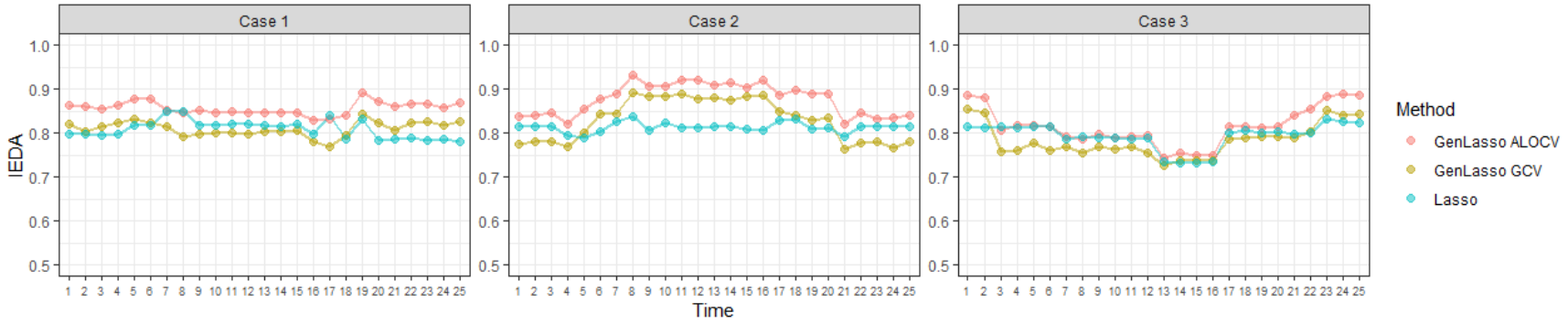


(b)

Figure 13. Line-plots of average of MSE from β_{it} over 47 prefectures for each time point, in the case of using (a) a common λ_S for all time points and (b) different $\lambda_{S,t}$ for each time point. The Cases 1, 2, and 3 correspond to the true $\alpha_t + \beta_{it}$ displayed in Figure 11.



(a)



(b)

Figure 14. Line-plots of $IEDA$ for each time point,
in the case of using (a) a common λ_S for all time points and (b) different $\lambda_{S,t}$ for each time point.
The Cases 1, 2, and 3 correspond to the true $\alpha_t + \beta_{it}$ displayed in Figure 11.

Table 5. Averages of $\overline{Sens^E}$, $\overline{PPV^E}$, and $IEDA$ over all time points

Case	Value	A common tuning parameter ($\lambda_{S,t} \equiv \lambda_S$)			Different tuning parameters ($\lambda_{S,t}$)		
		GenLasso ALOCV	GenLasso GCV	Lasso	GenLasso ALOCV	GenLasso GCV	Lasso
Case 1	$\overline{Sens^E}$	0.904	0.781	0.825	0.865	0.763	0.784
	$\overline{PPV^E}$	0.840	0.888	0.836	0.857	0.885	0.837
	$IEDA$	0.870	0.827	0.829	0.857	0.810	0.809
Case 2	$\overline{Sens^E}$	0.929	0.795	0.833	0.881	0.779	0.791
	$\overline{PPV^E}$	0.869	0.915	0.841	0.884	0.911	0.838
	$IEDA$	0.896	0.847	0.837	0.877	0.831	0.813
Case 3	$\overline{Sens^E}$	0.902	0.842	0.839	0.865	0.790	0.818
	$\overline{PPV^E}$	0.765	0.785	0.775	0.780	0.800	0.775
	$IEDA$	0.825	0.808	0.803	0.815	0.783	0.794

Figure 14 shows the plots of $IEDA$ for all time points in clustering prefectures. We only show the result of the generalized lasso with ALOCV and GCV and lasso because all edges take non-zero differences for other methods. We can see that the generalized lasso with ALOCV outperformed, as indicated by the highest $IEDA$ for most cases in Figure 14(a) and Figure 14(b). The $IEDA$ generally increased when the number of separated regions was small and decreased when the number of separated regions was large. Table 5 shows the averages of $\overline{Sens^E}$, $\overline{PPV^E}$ and $IEDA$ over all time points. The generalized lasso with ALOCV provided higher sensitivity and $IEDA$ than the generalized lasso with GCV and the lasso, although the coefficients of lasso were pooled in advance based on 8 prefectures regions. Moreover, we obtained slightly higher $IEDA$ when using a common tuning parameter than using different tuning parameters for all the cases and methods. If we use a common tuning parameter, the chosen tuning parameter value was not too small, so that the differences between coefficients on the edges tended to shrink to zero, which resulted in more accurate clustering. In contrast, if we use a different tuning parameter at each time point, the tuning parameter chosen varied greatly and was small for many time points. As a result, the differences between coefficients on the edges did not tend to shrink to zero, which decreased the clustering accuracy.

In summary, our simulation study showed that the proposed method performed well in estimating the temporal effect, as suggested by lower MSE. Moreover, the proposed method was also very flexible in detecting multiple clusters, as shown by high $IEDA$ values. The generalized lasso with ALOCV outperformed in detecting clusters, while the generalized lasso with GCV performed well in estimating coefficients.

4.2.4 Real Case Application: Covid-19 Cases in Japan

Since the first confirmed case was detected on January 16, 2020, Japan has experienced 5 major waves of the spread of Covid-19 until September 2021. As of September 11, 2021, the total number of Covid-19 cases in all prefectures in Japan was 1,627,898, with 98% recovered rate (Ministry of Health, Labor, and Welfare, 2021). At that time, the number of Japan's Covid-19 cumulative confirmed cases was the 26th highest in the world (WHO, 2021). In our study, we choose the start point on March 21, 2020, because on that date the total confirmed Covid-19 cases exceeded 1,000, spread in 39 of 47 (83%) prefectures of Japan.

Figure 15 shows daily reported Covid-19 cases in Japan, with (a)-(d) indicating periods of each declaration of emergency status respectively. To correspond with the first wave of Covid-19 spread, the first emergency status was declared on April 7, 2020, first in seven prefectures, and then it was expanded nationwide on April 16, 2020 (Figure 15(a)). The second wave occurred in August 2020, but at that time the government didn't declare an emergency status until the end of the year. After the number of cases decreased in autumn, the third wave occurred at the end of 2020, in which the number of infections reached 230,000 people. The second emergency status was declared for Saitama, Chiba, Tokyo, and Kanagawa on January 8, 2021, and was expanded to 11 prefectures on January 13, 2021. The duration of this emergency status was until March 7, 2021 (Figure 15(b)). The first dose of Covid-19 vaccination was implemented on April 1, 2021, while at that time, Japan was hit by the fourth wave of outbreak. The third emergency declaration was issued for Tokyo, Osaka, Kyoto, and Hyogo on April 25, 2021, and was expanded to five other prefectures on May 16, 2021, which was lifted on June 20, 2021, except in Okinawa (Figure 15(c)). The fifth wave occurred around July to September 2021, during which the Olympic Summer Games was held in Tokyo, and the fourth state of emergency was declared in several prefectures, particularly to prevent spread of the highly contagious Delta variant (Figure 15(d)).

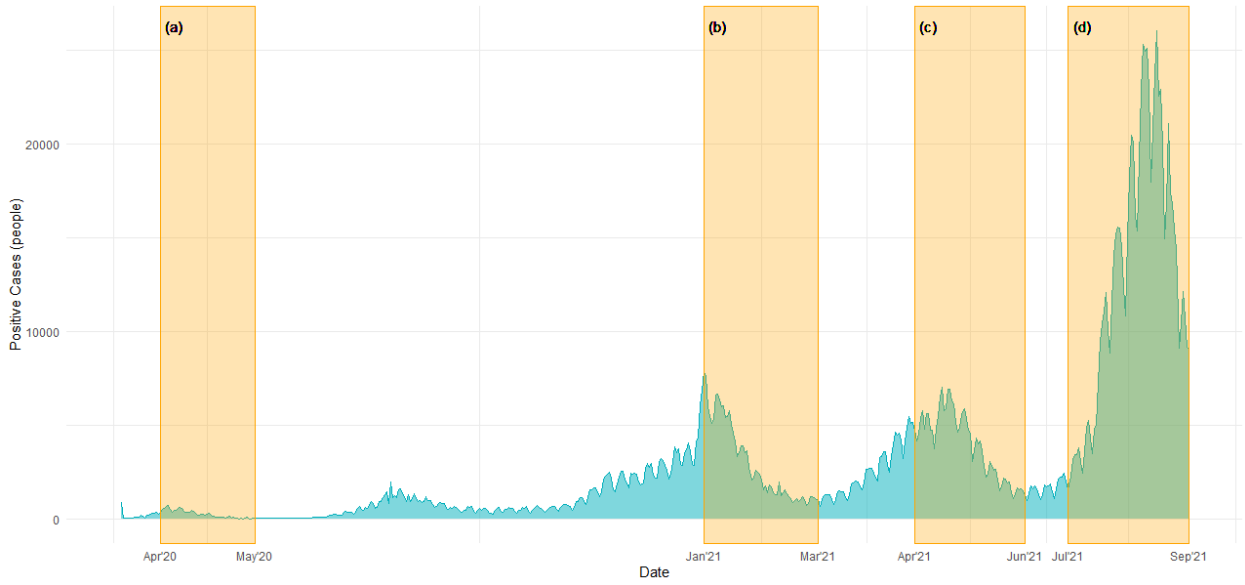


Figure 15. Daily reported Covid-19 positive cases in Japan from March 18, 2020, to September 11, 2021, with emergency status periods (a)-(d)

We applied the minimization problem (4.3) for spatio-temporal analysis, which can be decomposed into the two generalized lasso problems (4.9) and (4.10), with tuning parameter λ_T and $\lambda_{S,t}$ selected by ALOCV and GCV, to understand the temporal effect and prefectural clusters constructed at each time point. We used the weekly Covid-19 positive case data for each prefecture in Japan from March 21, 2020, to September 11, 2021 (the data file covid_jpn_prefecture.csv in Takaya (2021)). Therefore, we have $S = 47$ and $T = 78$. Let y_{it}^* be the number of weekly positive cases in the i-th prefecture and at the t-th week, and N_i be the population in the i-th prefecture. Here, we used the log transformed positive case per population $y_{it} = \log\left(\frac{y_{it}^*}{N_i}\right)$ as the response variable in the generalized lasso problem. The adjacency between each pair of prefectures was introduced as constraints in the same way as in Section 4.2.3, and hence we have $\boldsymbol{D}_1 \in \mathbb{R}^{76\times 78}$ and $\boldsymbol{D}_2 \in \mathbb{R}^{93\times 47}$.

We calculated an unbiased estimate of the DF for λ_T or $\lambda_{S,t}$ to evaluate complexity of the model. According to Tibshirani & Taylor (2011), the DF for given λ in the generalized lasso (4.1) is defined as,

$$df = E[nullity(\boldsymbol{D}_{-E})], \tag{4.17}$$

where $nullity(\boldsymbol{D}_{-E})$ is the dimension of the null space of $\boldsymbol{D}_{-E}$, reduced rows of the penalty matrix $\boldsymbol{D}$ corresponding to the boundary index set E of a solution of the dual problem (2.4) or (2.5). The DF for the ℓ_1 penalty suggests the number of fused groups. In estimating temporal effect with (4.9), we used the formula (4.17) to calculate DF for selected λ_T. In estimating spatial effect for each time point with (4.10), we selected a common λ_S for all $t = 1,2,\ldots,78$, based on the simulation study described in Section 4.2.3, in which a common tuning parameter resulted in higher $IEDA$ values. In this case, the DF was calculated as the average value of DF over all $t = 1,2,\ldots,78$.

a. Result of estimating temporal effect

We considered minimizing (4.9) to estimate the temporal effect α_t. Table 6 contains selected λ_T and the DF based on the proposed method with ALOCV and GCV. We limited the maximum DF to $\frac{3}{4}(78) = 58.5$ to avoid extremely rough temporal effect. The generalized lasso with ALOCV selected higher λ_T and smaller DF than the results of the generalized lasso with GCV.

Table 6. Selected λ_T and DF for estimating the temporal effect α_t

Method	Selected λ_T	DF
GenLasso with ALOCV	0.2759	37
GenLasso with GCV	0.0411	51

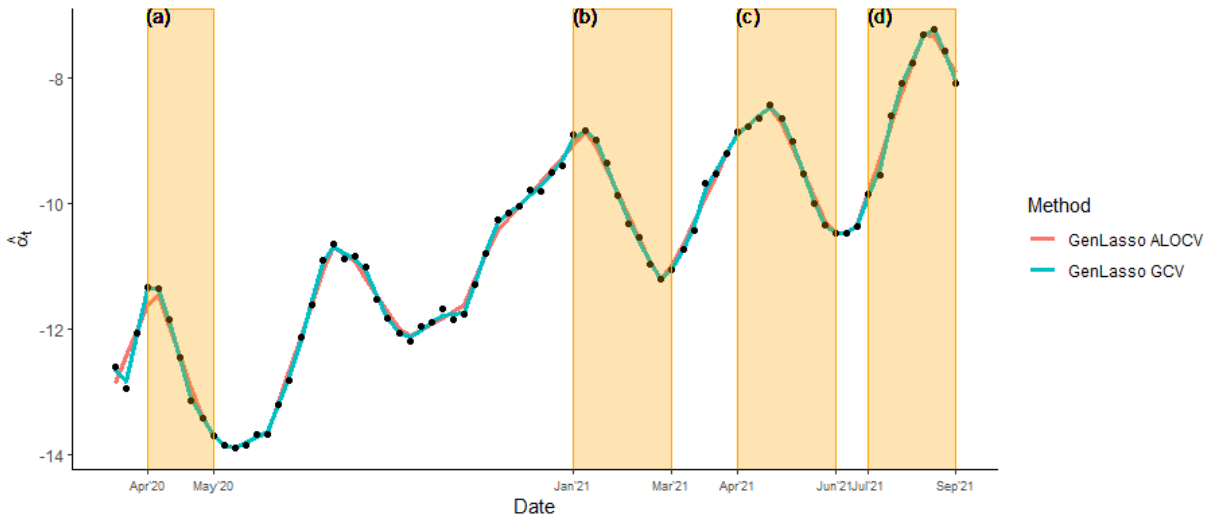


Figure 16. Plot of estimated temporal effect $\hat{\alpha}_t$ based on λ_T selected by using generalized lasso with ALOCV and GCV, with emergency status periods (a)-(d)

Figure 16 shows the estimated temporal effect $\hat{\alpha}_t$ for each t based on selected λ_T using generalized lasso with ALOCV and GCV, with each emergency status period (a)-(d). The break points in the estimated trend should suggest some change of conditions such as emergency status declaration. The estimated trend using ALOCV for ℓ_1 penalty has slightly fewer break points than the one using GCV for ℓ_1 penalty. During the first emergency status period (a), the estimated temporal effect reached the first peak at first and then fell down. After the period (a) ended, it rose quickly and reached the second peak in summer of 2020. During the second emergency status period (b), it reached the third peak at first and then fell down again. During the third emergency status period (c), it slightly increased for a while, reached the fourth peak, and then decreased quickly. During the fourth emergency status period (d), it increased for more than one month and then reached the fifth peak, and then decreased.

b. Result of estimating spatial effect

We considered minimizing (4.10) to estimate the spatial effect β_{it}. In this case, we assumed that $\lambda_S \equiv \lambda_{S,t}$ for all $t = 1,2,\ldots,78$ and selected it by minimizing the sum of ALOCV and GCV errors over all t.

Table 7 contains selected λ_S and the DF. We limited the maximum DF to $\frac{3}{4}(47) = 35.25$ to avoid extremely rough spatial effect. ALOCV was not computable at lower λ_S (less than 3.47) for the reason of division-by-zero issue, described in Section 3.1.4, so that ALOCV selected higher λ_S with lower DF compared to GCV.

Table 7. Selected common λ_S and DF for estimating spatial effect

Method	Selected λ_S	DF
GenLasso with ALOCV	4.1400	1.29
GenLasso with GCV	0.8939	6.64

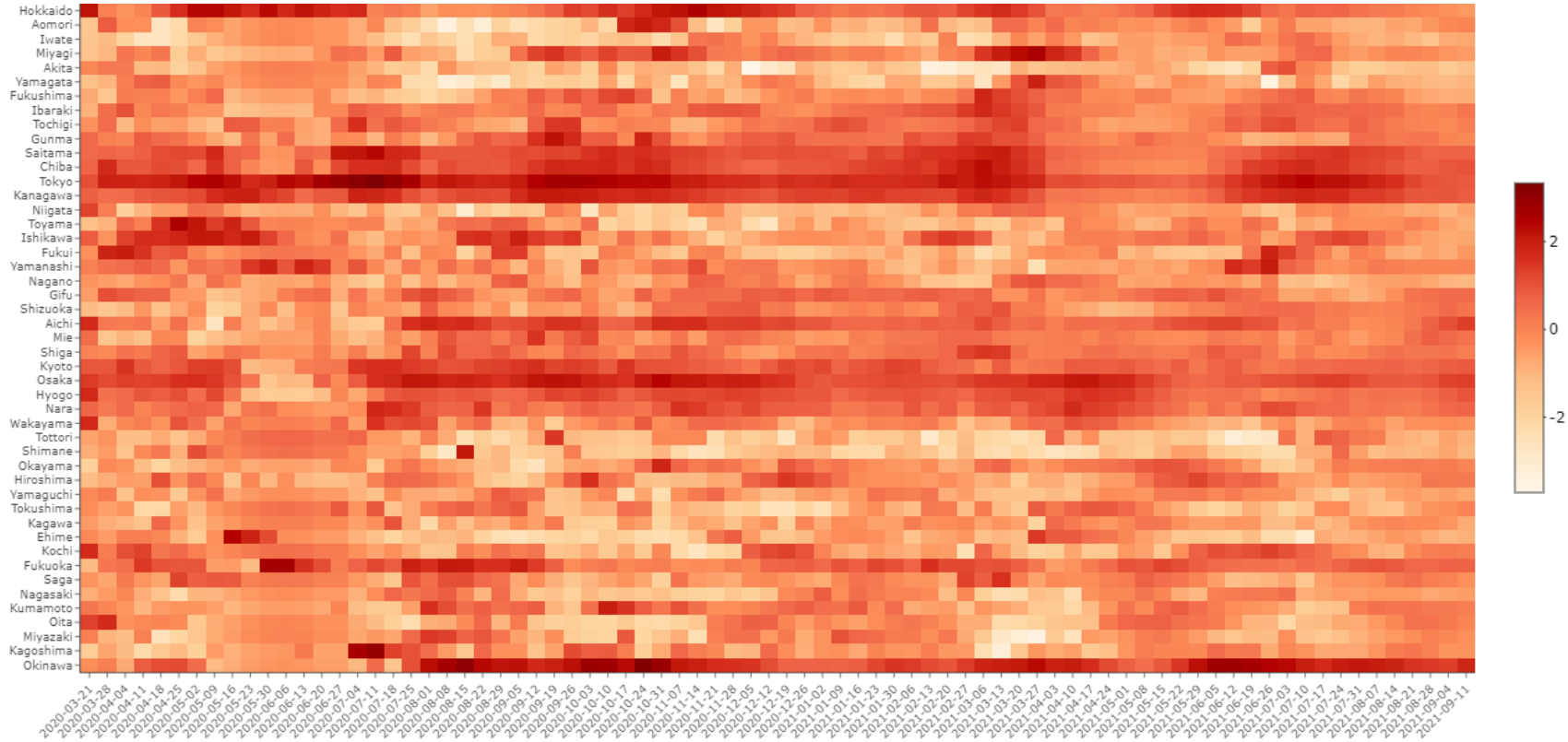


Figure 17. Heatmap of unpenalized estimated spatial effect $\hat{\beta}_{it}$ with a common λ_s.

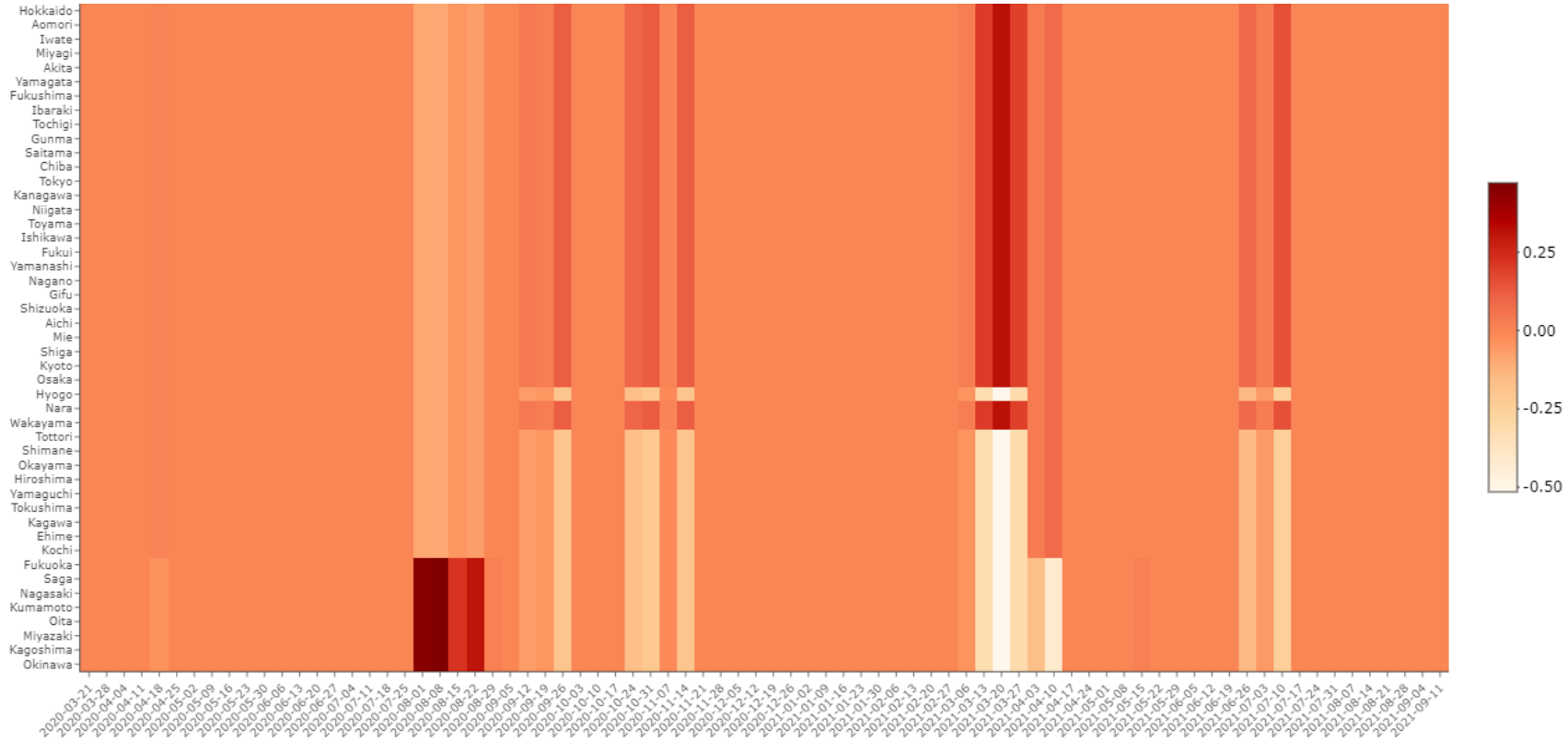


Figure 18. Heatmap of estimated spatial effect $\hat{\beta}_{it}$ with a common λ_S selected by ALOCV.

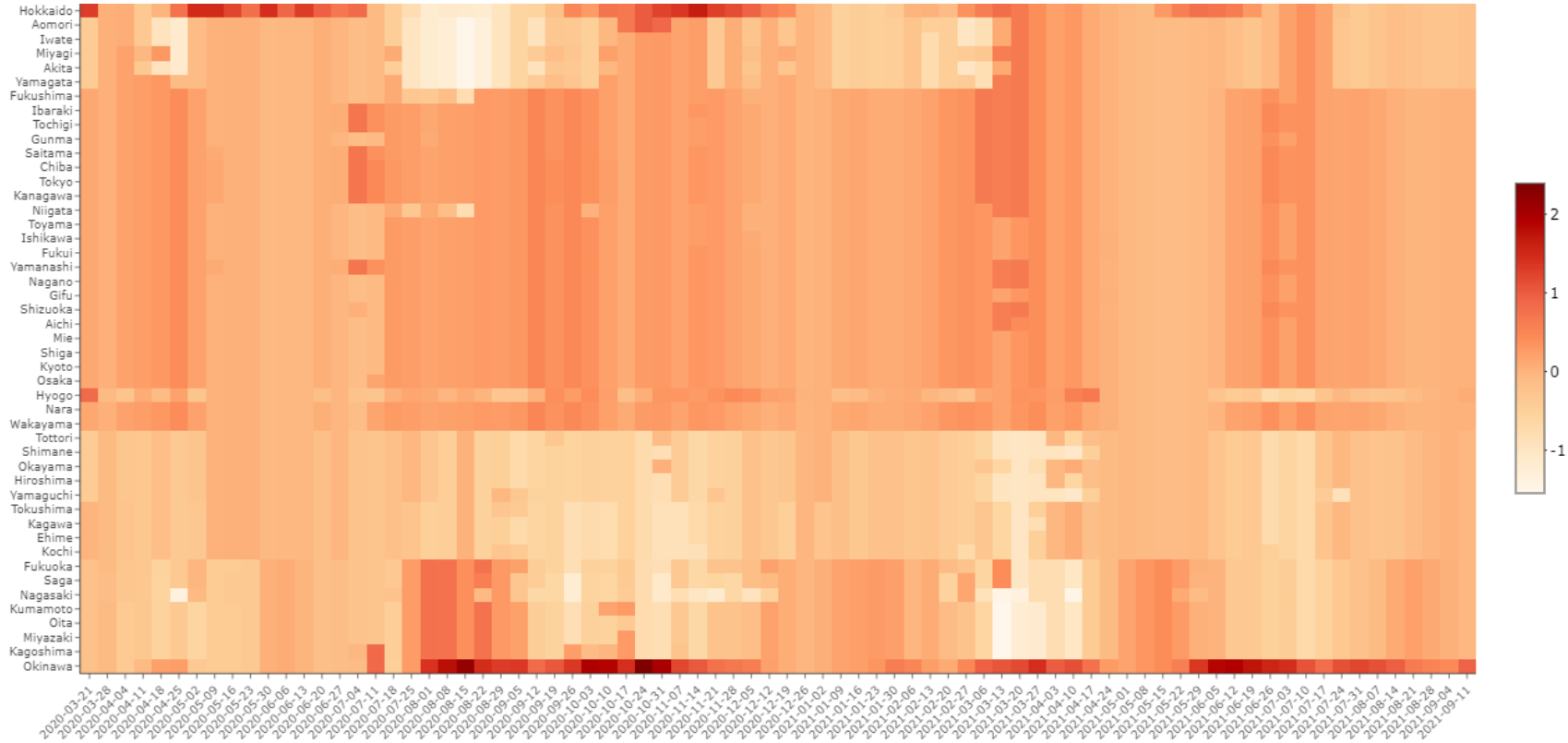


Figure 19. Heatmap of estimated spatial effect $\hat{\beta}_{it}$ with a common λ_s selected by GCV.

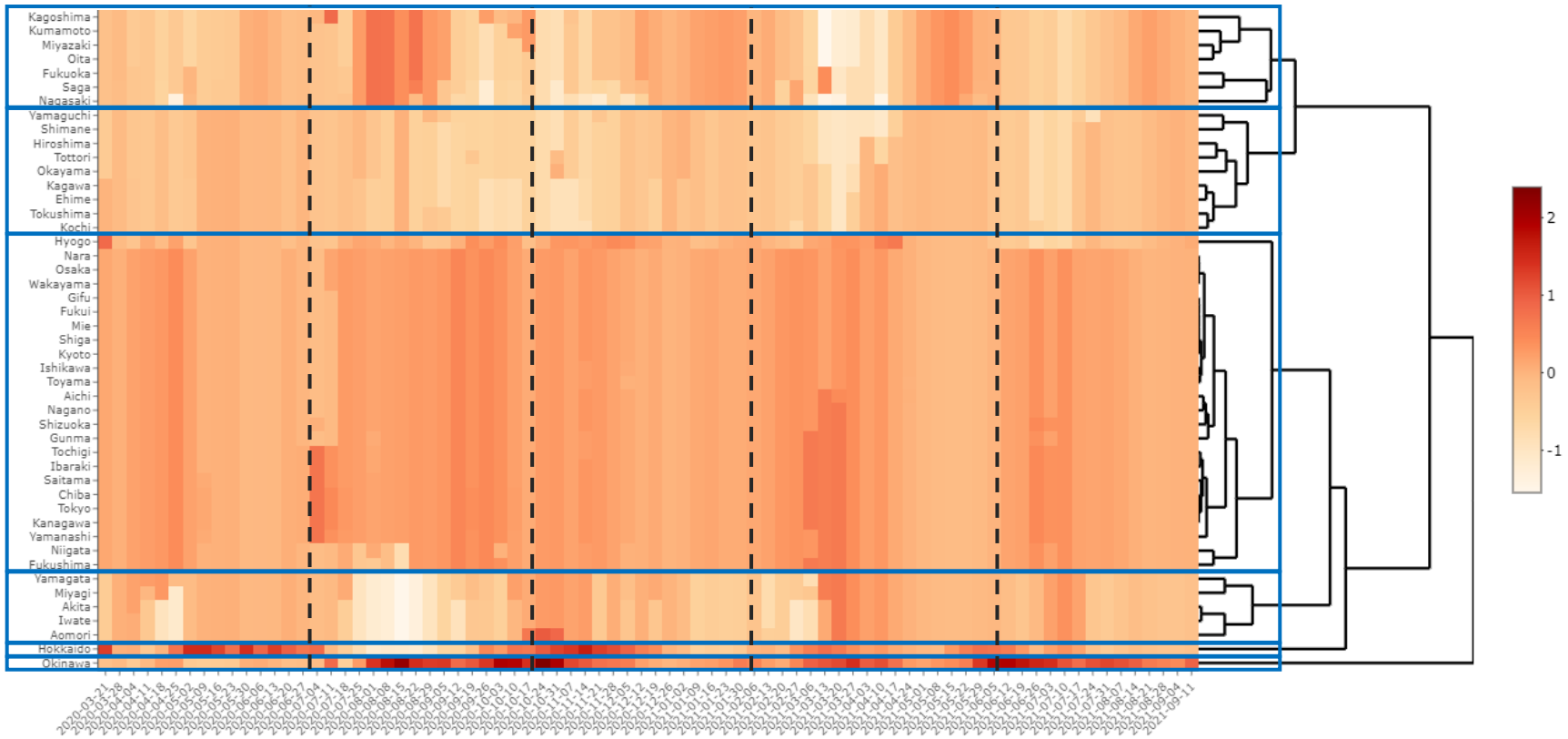


Figure 20. Heatmap of estimated spatial effect $\hat{\beta}_{it}$ with a common λ_S selected by using GCV for generalized lasso, in which prefectures arranged based on agglomerative hierarchical clustering. The blue horizontal boxes indicated separated clusters constructed and black vertical dashed lines indicated separator between waves.

The estimates $\hat{\beta}_{it}$ of spatial effect for these methods of selecting λ_S were plotted in Figure 17, Figure 18, and Figure 19, in which the prefectures are plotted roughly from North (top) to South (bottom), and darker red color indicates higher values. Figure 17 shows unpenalized estimates of the spatial effect $\hat{\beta}_{it} = y_{it} - \bar{y}_{.t}$. Based on this figure, we can see the relative spread of Covid-19 for each prefecture in every week. However, the unpenalized $\hat{\beta}_{it}$ looks very rough, and hence it is very difficult to grasp commonalities and differences of spatial effect between regions for each week. Figure 18 shows the estimated spatial effect $\hat{\beta}_{it}$ with λ_S selected by using ALOCV. It looks very smooth, and one or few clusters covered all prefectures at most of the weeks, and at some weeks the estimated spatial effect had large difference depending on the clustered regions. Figure 19 shows estimated spatial effect $\hat{\beta}_{it}$ with λ_S selected by using GCV. It suggests that there were some clusters of prefectures with the same values. We can see that the largest cluster consisted of most of prefectures during the emergency status periods. However, in other period of weeks, the prefectures were divided into some clusters.

c. Clustering the regions based on the estimated spatial effect

Figure 21 shows the heatmap of the estimated spatial effect $\hat{\beta}_{it}$ with a common λ_S selected by using GCV for generalized lasso as in Figure 19, but prefectures have been arranged based on agglomerative hierarchical clustering. The heatmap after the arrangement can display relative infection risk, that is, how the infection occurred in a specific area and then spread to other areas.

Based on Figure 21, we can detect six major clusters of prefectures from top to bottom: 1) all prefectures in Kyushu region, 2) all prefectures in Chugoku and Shikoku regions, 3) all prefectures in Kanto, Chubu, Kansai regions, and Fukushima prefecture (central part of Japan), 4) prefectures in Tohoku region except Fukushima, 5) Hokkaido prefecture, and 6) Okinawa prefecture. We provide the following interpretation of the dynamic behavior of spatial clusters based on the result of generalized lasso clustering with separating into the five waves that Japan has experienced.

i. First wave (March 21 to June 27, 2020)
   During the first wave of infections, relative infection risk increased gradually in the central part of Japan (cluster 3) and Okinawa (cluster 6) and decreased in the remaining clusters. Then, while the first emergency status had been declared from mid-April to May 2020, relative infection risk was extremely higher in Hokkaido (cluster 5) and decreased gradually in the other clusters.
ii. Second wave (July 4 to October 17, 2020)
   In the second wave of outbreaks, relative infection risk was higher in Kyushu (cluster 1), central Japan (cluster 3), and Okinawa (cluster 6), while lower in the other clusters. In cluster 1, the outbreak reached a peak in August 2020 and then decreased gradually. During the period, the relative risk increased gradually in cluster 3. It was the highest and stagnant in Okinawa.
iii. Third wave (October 24, 2020, to February 6, 2021)
   In the third wave, relative risk was higher in central and northern parts of Japan (clusters 3, 4, 5, and 6). Then, it decreased gradually while the second emergency status had been declared from January to March 2021.
iv. Fourth wave (February 13 to June 5, 2021)
   After higher risk in Kanto and Tohoku regions in March 2021, the fourth wave spread to other regions. While the third emergency status had been declared from April to June 2021, relative

risk was higher in Okinawa (cluster 6) in April and in Kyushu region (cluster 1) in May but was lower in the other clusters.

v. Fifth wave (June 12 to September 11)
In the fifth wave of infections, infection risk increased first in Okinawa (cluster 6), spread into central Japan (cluster 3), Tohoku (cluster 4), and Hokkaido (cluster 5), next into Chugoku and Shikoku (cluster 2), and then into Kyushu (cluster 1).

In summary, we can see that the outbreaks that occurred in central Japan (cluster 3) spread into outer regions such as Chugoku-Shikoku region (cluster 2) and Tohoku region (cluster 4) in one month, and then spread into Kyushu region (cluster 1) a few months late. We can also see that the outbreaks in some regions leaped into Hokkaido (cluster 5) and Okinawa (cluster 6) a few months late.

4.2.5 Conclusion

In this study, we proposed a regularization approach using a modified generalized lasso model with two ℓ_1 penalties for temporal effect and spatial effect. Then, our proposed method can be separated into two generalized lasso problems: trend filtering to estimate smooth temporal effect and fused lasso to detect clusters of spatial location for each time point. Through our proposed method, we can understand dynamic behavior of spatial clusters over time more flexibly, based on relative magnitude of estimated spatial effect at each time point.

To select the appropriate tuning parameters in the generalized lasso, we considered using ALOCV and GCV. Our simulation study suggested that estimation of temporal and spatial effects using generalized lasso with ALOCV and GCV was comparable or superior in terms of MSE to existing regularization methods such as lasso, ridge, and generalized ridge. Also, we showed that the generalized lasso with ALOCV provided higher $IEDA$, the accuracy of detecting edges with non-zero difference. In addition, our simulation study suggested that a common tuning parameter over all time points was preferable in spatial clustering.

Then, through the analysis of weekly Japan's Covid-19 panel data, we illustrated how to understand the spread of Covid-19 infection using our modified generalized lasso model. In estimation of the spatial effect over weeks, the generalized lasso with a common tuning parameter over all time points selected by GCV, provided a reasonable result.

5 Spatial Varying Coefficient Modeling

This chapter discusses the generalized lasso application in the spatial modeling analysis. This study was inspired by Zhao & Bondell (2020) who applied the generalized lasso to fit spatially varying coefficient models. In this case, we incorporate the explanatory variables in the model, so it is assumed that the matrix $\boldsymbol{X} \neq \boldsymbol{I}$. We provided two exciting studies on this topic. First, we applied the generalized lasso to reveal the relationship between socio-economic factors and Covid-19 cases in Java Island, Indonesia, as a spatially varying coefficient modeling. In that case, we also provided the region clustering as the result of the model estimates (Rahardiantoro & Sakamoto, 2021). Second, we proposed the modified generalized lasso to fit spatially varying coefficient models which consist of numerical and categorical explanatory variables. The proposed method was applied in the province-wise analysis of house sales price data on Java Island, Indonesia (Rahardiantoro & Sakamoto, 2022b).

5.1 Spatially Varying Coefficient Modeling between Socio-economic Factors and Covid-19 Cases in Java Island, Indonesia

5.1.1 Background

As of October 5th, 2020, the number of Covid-19 positive cases in Indonesia was greater than 300,000 cases, since the first case was announced by the government at the beginning of February 2020. In Indonesia, more than 60% of Covid-19 cases occurred in Java Island, where the capital city, Jakarta, is located. In this study, we consider revealing the relationship between the socio-economic factors that affect the number of Covid-19 positive cases in Java Island by using a statistical modeling approach. The socio-economic factors which were used namely poverty percentage, Human Development Index (HDI), average of expenditure per month, and Open Unemployment Rate (OUR).

The generalized lasso was applied to provide clustering adjacent regions (cities and regencies) in Java Island into some groups based on the socio-economic factors that are suspected to affect the Covid-19 cases, which could be useful for decision making by government. Our target variable y is the number of Covid-19 cases per 100,000 residents, in each regency in Java Island, Indonesia. The number of Covid-19 cases (as of September 29, 2020) was collected based on the monthly report of each province governments in Java Islands (Pemerintah Provinsi Banten, 2020; Pemerintah Provinsi DKI Jakarta, 2020; Pemerintah Provinsi Jawa Barat, 2020; Pemerintah Provinsi Jawa Tengah, 2020; Pemerintah Provinsi Jawa Timur, 2020; Pemerintah Provinsi D.I. Yogyakarta, 2020). The predictor variables are the recent aggregate calculation based on the statistics office of each province in Java Island (BPS-Statistics of Banten Province, 2020; BPS-Statistics of DKI Jakarta Province, 2020; BPS-Statistics of Jawa Barat Province, 2020; BPS-Statistics of Jawa Tengah Province, 2020; BPS-Statistics of Jawa Timur Province, 2020; BPS-Statistics of Daerah Istimewa Yogyakarta Province, 2020).

5.1.2 Data Description

The data set contains 119 cities and regencies in 6 provinces of Java Islands: Jakarta Province, West Java (Jawa Barat) Province, Central Java (Jawa Tengah) Province, East Java (Jawa Timur) Province,

Yogyakarta Province, and Banten Province. Figure 21 shows the distribution pattern of $\log y$ in each regency in Java Island. The high number of cases occurred in Jakarta Province and surrounding regencies, Semarang (the capital city of East Java), and also Surabaya (the capital city of East Java).

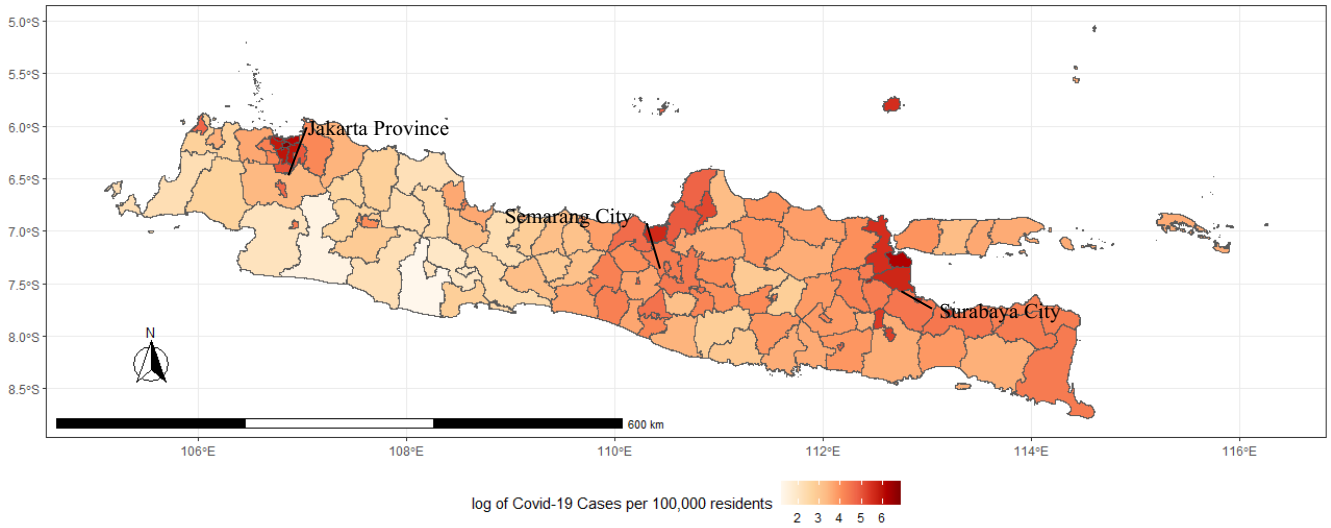


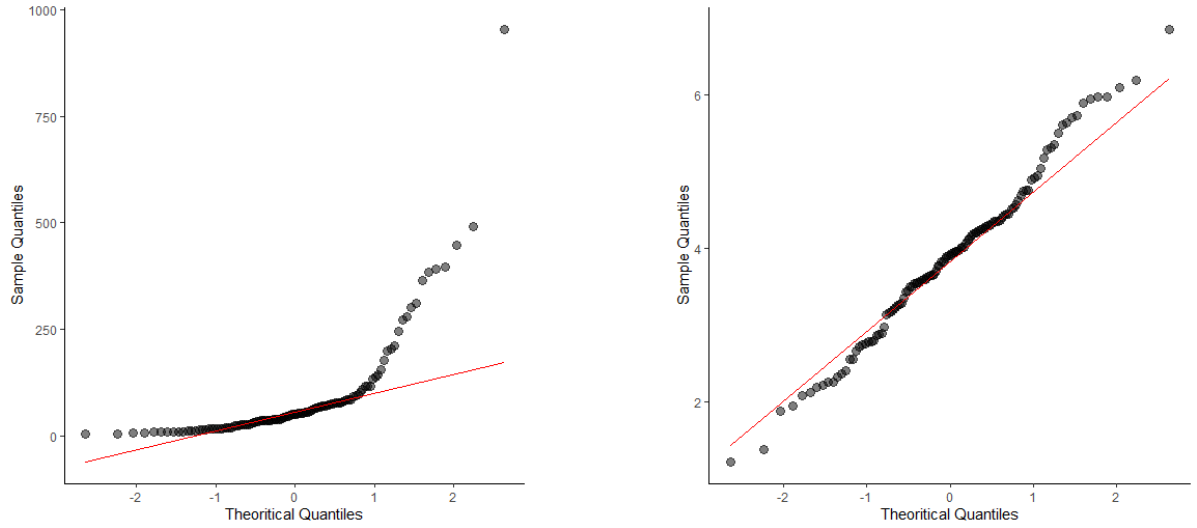
Figure 21. Distribution of log of Covid-19 cases per 100,000 residents in Java Island

For more details, Table 8 provides the summary statistics of each variable of the data set. The highest number of Covid-19 positive rates, 954.5, was observed in Central Jakarta City. It seems that this variable has right tailed distribution. Furthermore, the summary statistics of predictor variables are also listed in this table.

Table 8. Summary statistics of variables

Statistics	Covid-19 Cases per 100,000 residents	Poverty (%)	HDI	Average Expenditure per Month (IDR)	OUR (%)
Min.	3.364	1.680	61.94	646,386	0.950
1st Qu.	24.932	6.625	68.67	863,530	3.565
Median	50.057	9.120	71.75	987,853	4.480
Mean	89.592	9.149	72.61	1,142,366	5.273
3rd Qu.	84.808	11.380	75.58	1,266,295	7.205
Max.	954.521	20.710	86.65	2,625,288	10.650
St. Dev.	127.289	4.036	5.38	400,718	2.283

We use the natural logarithm of y for each regency as the response variable of this study to make the distribution of this variable more symmetric. Figure 22 shows the normal Q-Q plot for the distribution of y before and after natural logarithm transformation.



(a) (b)

Figure 22. Normal Q-Q plot for the distribution of y: (a) before and (b) after natural logarithm transformation

5.1.3 Data Analysis

We analyzed the Covid-19 data set using generalized lasso to fit spatially varying coefficient models (Gelfand et al., 2003). In the application to our data set, we have $n = 119$, $p = 4$. We also included an intercept term in the regression model, but it was not penalized. Before the analysis, we standardized each predictor variable so that the mean is 0 and standard deviation is 1. We divided 119 regencies into some spatial regions in the following two ways: i) 6 regions based on provinces; and ii) 10 regions based on K-means clustering, in which the neighborhood structure was specified by Voronoi tessellation approach. Then, we estimated region-specific coefficients for the intercept and the predictor variables. The details are explained in the following sections.

a. Generalized Lasso with Regions Based on Provinces

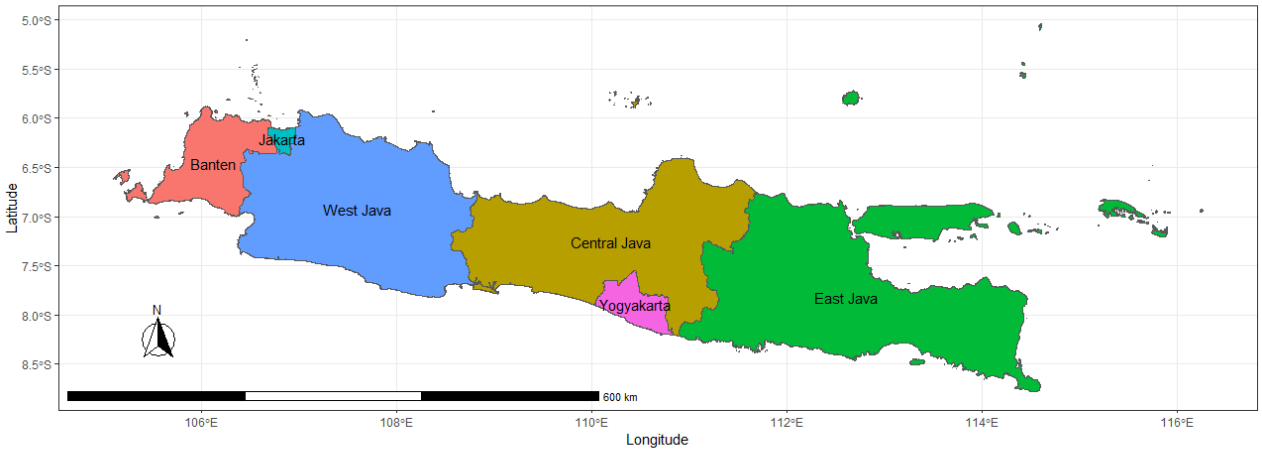


Figure 23. Provinces in Java Island

The 6 provinces in Java Island are represented in the Figure 23. To estimate region-specific coefficients for the intercept and predictor variables, the generalized lasso estimator of $\boldsymbol{\theta}$ can be expressed as follows by minimizing:

$$\sum\left(y_i^{(r)} - \theta_0^{(r)} - x_{i1}^{(r)}\theta_1^{(r)} - x_{i2}^{(r)}\theta_2^{(r)} - x_{i3}^{(r)}\theta_3^{(r)} - x_{i4}^{(r)}\theta_4^{(r)}\right)^2 + \lambda\|\boldsymbol{D\theta}\|_1, \tag{5.1}$$

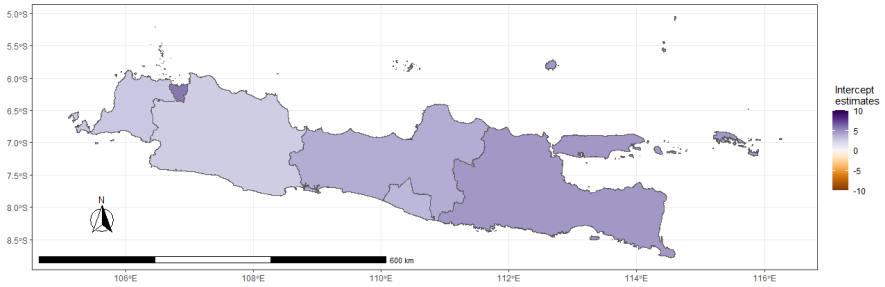
where $x_{i1}^{(r)}$, $x_{i2}^{(r)}$, $x_{i3}^{(r)}$, and $x_{i4}^{(r)}$ represents the predictor variables for poverty, HDI, expenditure, and OUR, respectively, $r = 1,2,\ldots,6$ represents 6 provinces, and $i = 1,2,\ldots$ denotes the i-th regency at each province. We incorporated an intercept and coefficients for predictors at each province, so that $\boldsymbol{y} = \left(\boldsymbol{y}^{(1)T},\ldots,\boldsymbol{y}^{(6)T}\right)^T \in \mathbb{R}^{119}$ is the response vector with $\boldsymbol{y}^{(r)} = \left(y_1^{(r)}, y_2^{(r)},\ldots,\right)^T$ corresponding to r-th province, $\boldsymbol{X} = \left[x_{ij}^{(r)}\right] \in \mathbb{R}^{119\times 30}$ is block-diagonal with r-th block matrix corresponding to r-th province, and $\boldsymbol{\theta} = \left(\boldsymbol{\theta}^{(1)T},\ldots,\boldsymbol{\theta}^{(6)T}\right)^T \in \mathbb{R}^{30}$, $\boldsymbol{\theta}^{(r)} = \left(\theta_0^{(r)},\ldots,\theta_4^{(r)}\right)^T$. For penalty matrix $\boldsymbol{D}$, we considered the penalization of parameters for a common predictor variable between adjacent regions, and each parameter itself, so that $\boldsymbol{D} = \begin{bmatrix}\boldsymbol{I}\\ \widetilde{\boldsymbol{D}}\end{bmatrix}$. The matrix $\widetilde{\boldsymbol{D}}$ is constructed based on adjacencies between provinces, that is,

$$\begin{aligned}\|\widetilde{\boldsymbol{D}}\boldsymbol{\theta}\|_1 = \sum\Big(&\left|\theta_0^{(r)} - \theta_0^{(r')}\right| + \left|\theta_1^{(r)} - \theta_1^{(r')}\right| + \left|\theta_2^{(r)} - \theta_2^{(r')}\right| + \left|\theta_3^{(r)} - \theta_3^{(r')}\right| \\ &+ \left|\theta_4^{(r)} - \theta_4^{(r')}\right|\Big),\end{aligned} \tag{5.2}$$

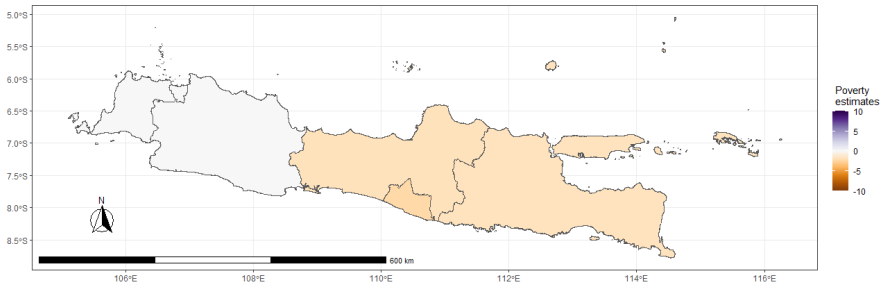
where the summation is taken on the edges of the graph, which corresponds to the adjacent regions (r, r'), so that there are 6 connections that are represented by each row in the matrix. Five 6×6 adjacency matrices of the same form are arranged diagonally to form a block diagonal matrix $\widetilde{\boldsymbol{D}}$, thus $\widetilde{\boldsymbol{D}} \in \mathbb{R}^{30\times 30}$. Therefore, we have $\boldsymbol{D} \in \mathbb{R}^{60\times 30}$.

We applied the generalized lasso regression and selected $\lambda = 0.0199$ by ALOCV minimizing the prediction error. We obtained $R^2 = 0.7955$, and RMSE = 0.4914. Figure 24 shows the spatially varying coefficient estimates of the intercept and the variables poverty, HDI, expenditure, and OUR.

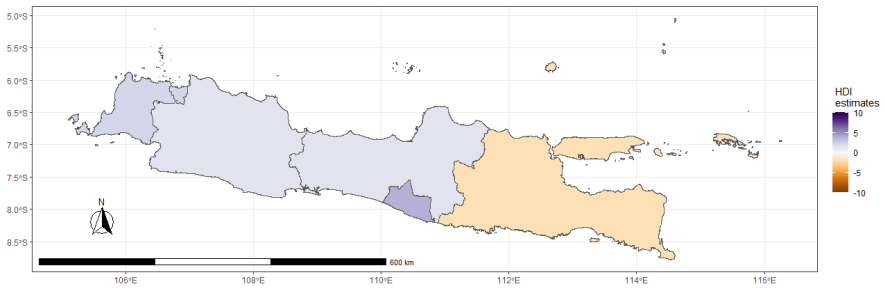
The result shown in Figure 24(a) provided different estimates of the intercept for each province, of which the highest intercept belonged to Jakarta Province, followed by East Java Provinces. The lowest intercept occurred in the West Java Province. For poverty coefficient estimates in Figure 24 (b), we obtained zero estimates for Banten Province, Jakarta Province, and West Java Province, which indicated that poverty in these provinces had no effect to the response variable. The remaining estimates for other provinces were relatively small. Four clusters were constructed on the estimated coefficient for HDI in Figure 24(c): the lowest coefficient occurred in East Java Province; the second cluster included Central Java and West Java Province; the third cluster included Banten Province and Jakarta Province, and the cluster with the highest coefficient occurred in Yogyakarta Province.



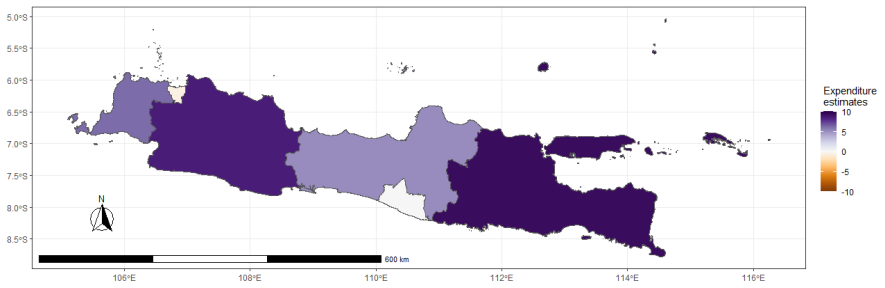
(a)



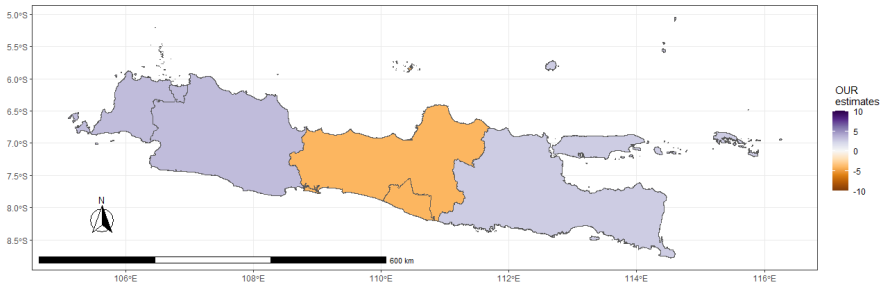
(b)



(c)



(d)



(e)

Figure 24. Plot of coefficient estimates using generalized lasso based on provinces, (a) intercept; (b) poverty; (c) HDI; (d) average of monthly expenditure; (e) OUR

The different estimates also obtained on the estimated coefficient for the monthly average of expenditure in Figure 24(d), of which the highest estimate belonged to East Java Province, followed by West Java Provinces. The two lowest estimates occurred in the Yogyakarta Province and Jakarta Province. On the estimated coefficient for Open Unemployment Rate (OUR) in Figure 24(e), Central Java and Yogyakarta Province had negative relationship to response variable, while the rest provinces had positive relationship to the response variable.

b. Generalized Lasso with Regions Based on K-means Clustering and Voronoi Tessellation

The regions were constructed by using K-means method with $K = 10$ based on Euclidian distance between the centre of each regency in longitude and latitude coordinates. Each region was composed of regencies close to each other. Based on the centroids of the constructed regions, a Voronoi tessellation was used to determine the neighborhood structure, as is shown in Figure 25.

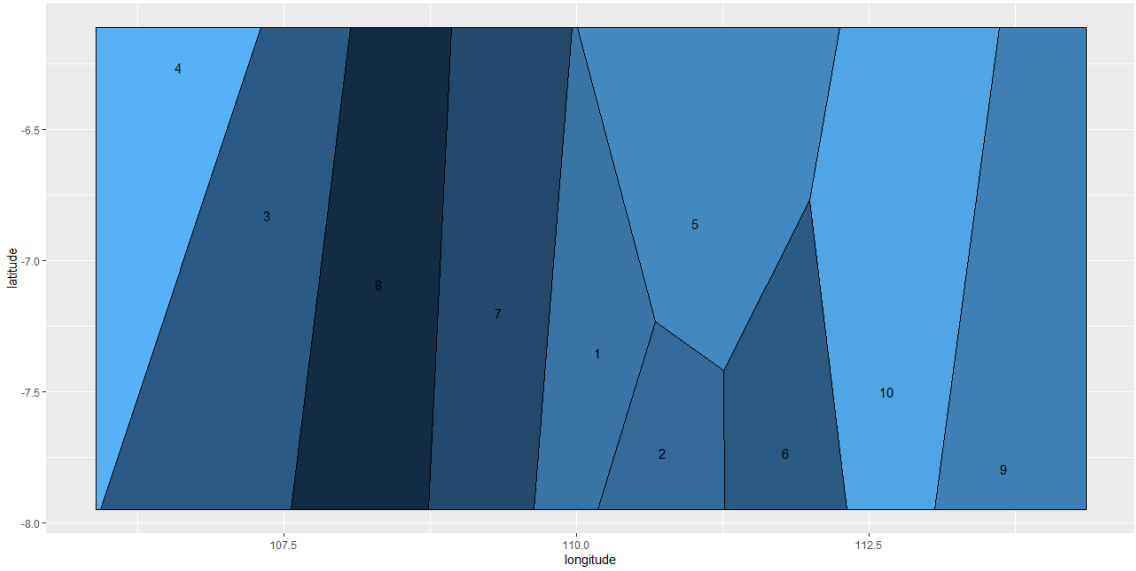
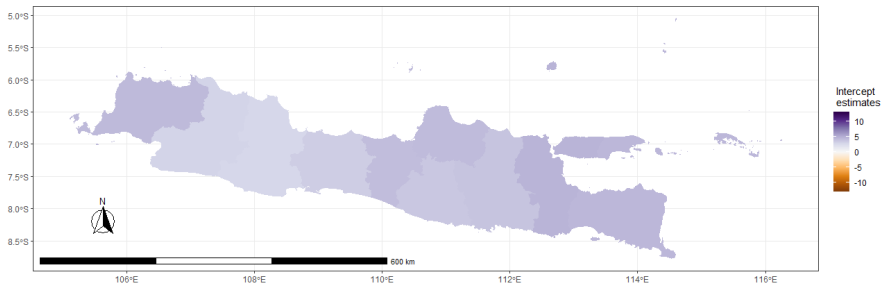


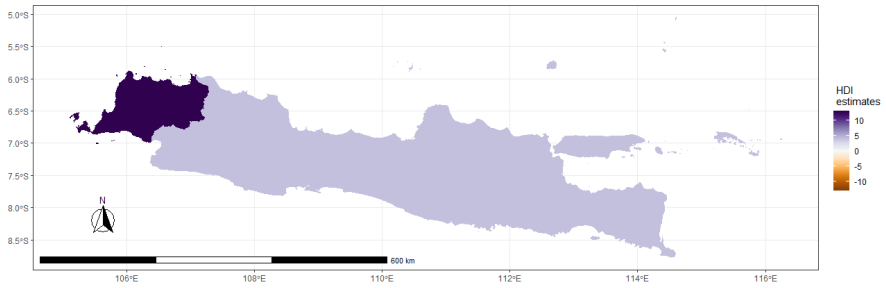
Figure 25. The neighborhood structure determined by Voronoi tessellation for the regions constructed by the K-means clustering ($K = 10$)

We applied the generalized lasso problem (5.1) again, where we have $r = 1,2,\dots,10$ for 10 regions. To identify cluster effects in the coefficients for the intercept and predictor variables, we constructed $\boldsymbol{X} \in \mathbb{R}^{119\times 50}$. Here, we also considered the penalization of parameters for a common predictor variable between adjacent regions, and each parameter itself, where $\boldsymbol{D} = \begin{bmatrix} \boldsymbol{I} \\ \widetilde{\boldsymbol{D}} \end{bmatrix}$; $\widetilde{\boldsymbol{D}} \in \mathbb{R}^{60\times 50}$, similarly as in the analysis based on provinces, so that $\boldsymbol{D} \in \mathbb{R}^{110\times 50}$. Then we selected the $\lambda = 0.1486$ by ALOCV and obtained $R^2 = 0.7536$ and RMSE = 0.5461. Figure 26 shows spatially varying coefficient estimates of the intercept, HDI, and monthly average of expenditure variables based on this analysis. We obtained zero estimates of coefficients for the other variables, poverty and open unemployment rate, over all regions, which indicated that there were no effect of these variables to the response variable over all regions.

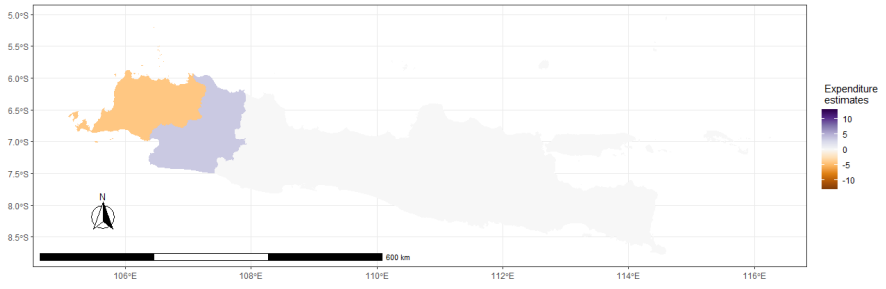
The estimated Intercept as in Figure 26(a) was separated into six clusters, of which we obtained the highest estimate in the cluster around East Java Province, while the lowest estimate in the east region of West Java Province and its surroundings. For HDI as in Figure 26(b), the estimated coefficient was separated into two clusters, of which the higher estimate occurred in Banten, Jakarta, and some regencies in western part of West Java Province. We also obtained some zero estimates in the monthly average of expenditure estimates as in Figure 26(c), while the greater non-zero estimate occurred in some regencies in western part of West Java Province.



(a)



(b)



(c)

Figure 26. Plot of spatially varying coefficient estimates using generalized lasso with neighborhood regions based on *K*-means clustering and Voronoi tessellation, (a) intercept; (b) HDI; (c) average of monthly expenditure

5.1.4 Conclusion

We found that the poverty variable generally had no effect to the response variable, for some provinces in the first model, and for over all regions in the second model. Moreover, the HDI variable provided spatially varying coefficient estimates with the two generalized lasso approaches. This suggests

that the HDI factor would have different effect on the infection rate of Covid-19 between regions on Java Island, that is, the western part of Java Island would have relatively higher HDI effect than the eastern part of Java Island.

5.2 Spatially Varying Coefficient Modeling with Numerical and Categorical Predictor Variables

5.2.1 Introduction

In the previous sub-section, the generalized lasso was applied to fit a spatially varying coefficient model in the application modeling of socio-economic factor to Covid-19 cases in Java Island, Indonesia. All of four predictor variables were in numerical scale. In the generalized lasso for numerical predictor variables, the differences of the coefficients between adjacent regions are regularized using ℓ_1 penalty. In contrast, regularization using the ℓ_1 penalty would be more complicated when there are several coefficients for categorical predictors in the model. Gertheiss & Tutz (2010) proposed shrinkage methods for categorical predictors, which were intended to select categorical predictors in the model as well as to shrink categories within each categorical predictor.

In this part, the goals of applying the spatially varying coefficient model are as follows:

a. fusion of categories within one categorical predictor in one region, and
b. fusion of adjacent regions for some categories within one categorical predictor.

To achieve these goals, we propose a generalized lasso model with two ℓ_1 penalties. Then, we apply the proposed method to house sales price data on Java Island, Indonesia, with a province-based model including both numerical and categorical predictors.

5.2.2 Proposed Method

Recall the region-based data, with regions indexed by $r = 1,2,\dots,R$ and observation within regions indexed by $i = 1,2,\dots,n_r$, with $\sum_{r=1}^{R} n_r = n$. Consider p predictor variables, indexed by $j = 1,2,\dots,p$. In this study, a spatially varying coefficient model is implemented for the response vector $\boldsymbol{y} = \left[y_{ij}^{(r)}\right] \in \mathbb{R}^n$ and the predictor matrix $\boldsymbol{X} = \left[x_{ij}^{(r)}\right] \in \mathbb{R}^{n \times pr}$, as a linear model $\boldsymbol{y} = \boldsymbol{X\theta} + \boldsymbol{\varepsilon}$, where $\boldsymbol{\theta} \in \mathbb{R}^p$ is a vector of parameters, and $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ is a vector of errors. We consider the block-diagonal predictor matrix $\boldsymbol{X}$ that consists of both numerical and categorical predictor variables, where the variables for categorical predictors are arranged as dummy variables. Let the first column of $\boldsymbol{X}$ be the vector of all elements 1, and for categorical predictor with C categories, we have $C - 1$ dummy variables arranged in $\boldsymbol{X}$.

Recall the generalized lasso problem in (2.2). Then, our proposed method can be expressed as minimizing

$$\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X\theta}\|_2^2 + \lambda_1\|\boldsymbol{D}_1\boldsymbol{\theta}\|_1 + \lambda_2\|\boldsymbol{D}_2\boldsymbol{\theta}\|_1 \tag{5.3}$$

where $\boldsymbol{D_1} \in \mathbb{R}^{m_1 \times p}$ is a penalty matrix based on adjacency between regions, with tuning parameter λ_1, and $\boldsymbol{D_2} \in \mathbb{R}^{m_2 \times p}$ is a penalty matrix based on adjacency between categories within one predictor variable, with tuning parameter λ_2.

In more details, each row of $\boldsymbol{D_1}$ contains all zeros except for –1 and 1 in the j-th and j'-th elements, where (j, j') indicate the pair of $\boldsymbol{X}$ columns in adjacent regions. For categorical predictor, the pair of non-zero (j, j') in $\boldsymbol{D_1}$ indicates the same categories in different adjacency regions. Each row of $\boldsymbol{D_2}$ also contains a pair of non-zero elements –1 and 1 in the j-th and j'-th elements, where (j, j') are in the same categorical predictor but different categories, in the same region. We define the pair of (j, j') in $\boldsymbol{D_2}$ based on the scale of categorical predictor. For unordered (nominal) categories we define

$$\left|\theta_j - \theta_{j'}\right| \tag{5.4}$$

where the pair of (j, j') indicate all possible pairs of categories in a common categorical predictor, where located in the same region. For ordered (ordinal) categories we define

$$\left|\theta_j - \theta_{j-1}\right| \tag{5.5}$$

where the pair of $(j, j-1)$ indicate adjacent level of categories in a common categorical predictor, where located in the same region. The $\boldsymbol{D_2}$ also includes rows corresponding to difference from the baseline category, that is, $\left|\theta_j\right|$ $(j = 1,2, \dots, C-1)$ for nominal category case, and $|\theta_1|$ for ordinal category case.

To solve (5.3), consider a grid of fixed $0 < \alpha < 1$, where $\alpha = \frac{\lambda_1}{\lambda_1 + \lambda_2}$ and let $\lambda = \lambda_1 + \lambda_2$. Thus, (5.3) can be simplified as

$$\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X\theta}\|_2^2 + \lambda\left\|\widetilde{\boldsymbol{D}}\boldsymbol{\theta}\right\|_1 \tag{5.6}$$

where $\widetilde{\boldsymbol{D}} = \begin{bmatrix} \alpha \boldsymbol{D_1} \\ (1-\alpha)\boldsymbol{D_2} \end{bmatrix}$. Then, (5.6) can be solved by the dual path algorithm for the generalized lasso, similar to the solution in (2.2). For a specified α, we select the optimum λ using the ALOCV as in Algorithm 2, and GCV for the comparison.

5.2.3 Simulation Study

Before applying the proposed method to the house sales price in Java Island data in following sub-section, we compare different approaches and investigate some characteristics in simulation studies. We combined the idea of numerical experiments in Zhao & Bondell (2020) and Gertheiss & Tutz (2010), to construct the region-based data with numerical and categorical predictor variables. The objectives are to investigate performance about how regions are pooled, and how categories are pooled.

We consider six regions (R1, R2, R3, R4, R5, and R6), which are separated into two regional clusters: cluster 1 = {R1, R2, and R3} and cluster 2 = {R4, R5, and R6}. The pairs of adjacent regions are defined

as: R1 – R2, R1 – R3, R2 – R3, R3 – R4, R4 – R5, and R4 – R6, representing structure of adjacent provinces in Java Island. For each region, we generated 50 observations.

We considered the regression model with three predictors – one numerical and two categorical predictor variables. We set eight categories ($C = 8$) to construct $C - 1 = 7$ dummy variables for both categorical predictors. The categories of the two categorical predictors were randomly generated based on the probability $(0.1, 0.1, 0.2, 0.06, 0.2, 0.1, 0.2, 0.04)^T$ for each observation. Therefore, we have $6 \times (1 + 1 + 7 + 7) = 96$ predictor variables including intercept for each region in the spatially varying coefficient model. The true parameters in the model are set as follows, where the first six elements are intercept from each region, the next six elements are coefficients for the numerical predictor, and the remained elements are for the two categorical predictors (different predictors were delimited by comma and different regions by space):

$$\begin{aligned}\boldsymbol{\theta} = (&2\,2\,2\,5\,5\,5, 1\,1\,1\,5\,5\,5, 0\,0\,0\,0\,0\,0, 1\,1\,1\,2\,2\,2, 1\,1\,1\,2\,2\,2, 1\,1\,1\,2\,2\,2, 1\,1\,1\,2\,2\,2,\\ &1\,1\,1\,-2\,-2\,-2, 1\,1\,1\,-2\,-2\,-2, 0\,0\,0\,0\,0\,0, 1\,1\,1\,2\,2\,2, 1\,1\,1\,2\,2\,2,\\ &-1\,-1\,-1\,2\,2\,2, -1\,-1\,-1\,2\,2\,2, 2\,2\,2\,4\,4\,4, 2\,2\,2\,4\,4\,4)^T\end{aligned}$$

We also considered the different scales of categorical predictors, as specified in three cases, namely: Case 1 for both categorical predictors as unordered (nominal) categories, Case 2 for both categorical predictors as ordered (ordinal) categories, and Case 3 for first categorical predictor as unordered (nominal) categories and the second as ordered (ordinal) categories.

For our settings, we have $\boldsymbol{y} \in \mathbb{R}^{300}$, $\boldsymbol{X} \in \mathbb{R}^{300\times 96}$. The numerical predictor was generated independently from a normal distribution with mean 0 and standard deviation 5 before being standardized and centered. The response variable $\boldsymbol{y}$ was generated by a spatially varying coefficient model $\boldsymbol{y} = \boldsymbol{X\theta} + \boldsymbol{\varepsilon}$, with each element of $\boldsymbol{\varepsilon}$ generated independently from $N(0, 3^2)$. Moreover, we have $\boldsymbol{D}_1 \in \mathbb{R}^{96\times 96}$ for all cases, and $\boldsymbol{D}_2 \in \mathbb{R}^{336\times 96}$, $\boldsymbol{D}_2 \in \mathbb{R}^{84\times 96}$, and $\boldsymbol{D}_2 \in \mathbb{R}^{210\times 96}$ for Case 1, Case 2, and Case 3, respectively. We replicated 100 datasets for each case.

Then, we applied our proposed method (5.3) to the datasets with using ALOCV and GCV for selecting the optimum λ for a specified α. We set $\alpha = 0.1, 0.25, 0.5, 0.75,$ and 0.9, and λ was searched on the interval $(0, 450)$. We compared our methods with the ordinary least square (OLS), the generalized ridge regression (Zhao & Bondell, 2020), and the generalized lasso with a single penalty matrix, either $\boldsymbol{D}_1$ or $\boldsymbol{D}_2$, respectively, with both using ALOCV and GCV for selecting the optimum λ_1 or λ_2 in (5.3). In the generalized ridge regression, we set the constraints on the ℓ_2 norm of coefficients differences in their neighborhood, which is given by:

$$\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X\theta}\|_2^2 + \lambda_1\|\boldsymbol{D}_1\boldsymbol{\theta}\|_2 + \lambda_2\|\boldsymbol{D}_2\boldsymbol{\theta}\|_2. \tag{5.7}$$

Similarly, we considered a grid of fixed $0 < \alpha < 1$, where $\alpha = \frac{\lambda_1}{\lambda_1+\lambda_2}$ and let $\lambda = \lambda_1 + \lambda_2$. Thus, (5.7) can be simplified as

$$\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{X\theta}\|_2^2+\lambda\left\|\widetilde{\boldsymbol{D}}\boldsymbol{\theta}\right\|_2, \tag{5.8}$$

where $\widetilde{\boldsymbol{D}}=\begin{bmatrix}\alpha\boldsymbol{D}_1\\(1-\alpha)\boldsymbol{D}_2\end{bmatrix}$. The solution to generalized ridge (5.8) is: $\widehat{\boldsymbol{\theta}}=\left(\boldsymbol{X}^T\boldsymbol{X}+\lambda\widetilde{\boldsymbol{D}}^T\widetilde{\boldsymbol{D}}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$. For specified α, we select the optimum λ by using LOOCV (van Wieringen, 2015). In this case, LOOCV error can be expressed as

$$\frac{1}{n}\sum_{i=1}^{n}\left(y_i-\boldsymbol{x}_i^T\widehat{\boldsymbol{\theta}}_{-i}\right)^2=\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i-\boldsymbol{x}_i^T\widehat{\boldsymbol{\theta}}}{1-h_{ii}}\right)^2, \tag{5.9}$$

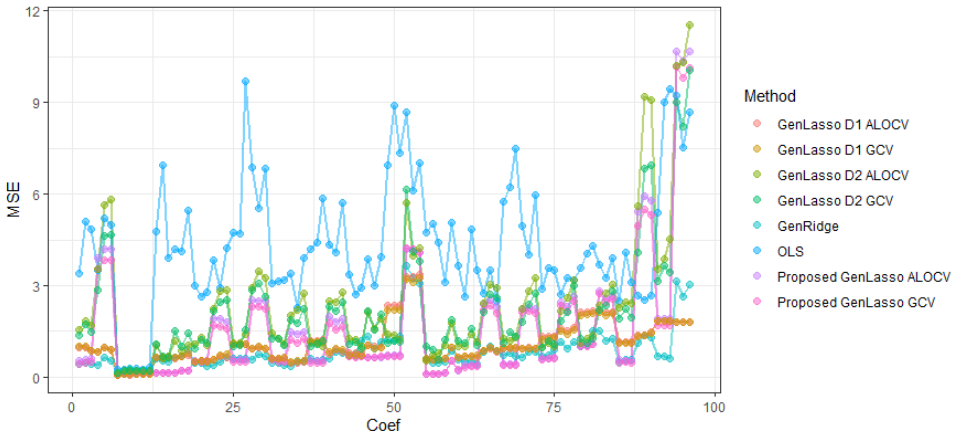
where $\widehat{\boldsymbol{\theta}}_{-i}$ can be stated as

$$\widehat{\boldsymbol{\theta}}_{-i}=\widehat{\boldsymbol{\theta}}-\left(\boldsymbol{X}^T\boldsymbol{X}+\lambda\widetilde{\boldsymbol{D}}^T\widetilde{\boldsymbol{D}}\right)^{-1}\boldsymbol{x}_i\frac{y_i-\boldsymbol{x}_i^T\widehat{\boldsymbol{\theta}}}{1-h_{ii}}, \tag{5.10}$$

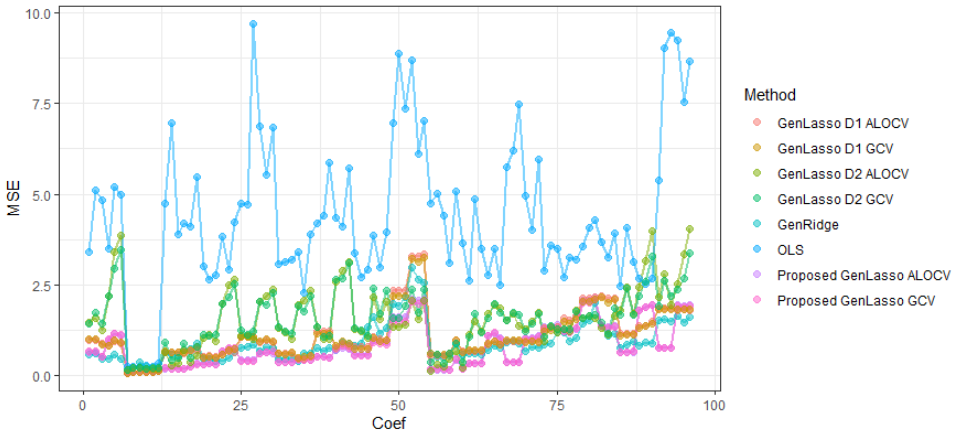
and h_{ii} is the i-th diagonal element of the hat-matrix $\boldsymbol{H}=\boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}+\lambda\widetilde{\boldsymbol{D}}^T\widetilde{\boldsymbol{D}}\right)^{-1}\boldsymbol{X}^T$. We selected λ at specified α with minimum LOOCV error (5.9).

The estimated coefficient vector $\widehat{\boldsymbol{\theta}}$ was compared to the true parameters for each case by using mean square error (MSE). To investigate how regions are pooled and how the categories are pooled, we counted the number of zero elements in $\boldsymbol{D}_1\widehat{\boldsymbol{\theta}}$ and $\boldsymbol{D}_2\widehat{\boldsymbol{\theta}}$, respectively. The true numbers of zero elements are 82 for $\boldsymbol{D}_1\widehat{\boldsymbol{\theta}}$; 108, 60, and 99 for $\boldsymbol{D}_2\widehat{\boldsymbol{\theta}}$ in Case 1, 2, and 3, respectively. We also computed the accuracy of detecting zero elements in $\boldsymbol{D}_1\widehat{\boldsymbol{\theta}}$ and $\boldsymbol{D}_2\widehat{\boldsymbol{\theta}}$ as summarized in Table 9 and Table 10. The results based on 100 datasets are shown as line-plots for MSE in Figure 27 and boxplots for the number of zero elements in Figure 28.

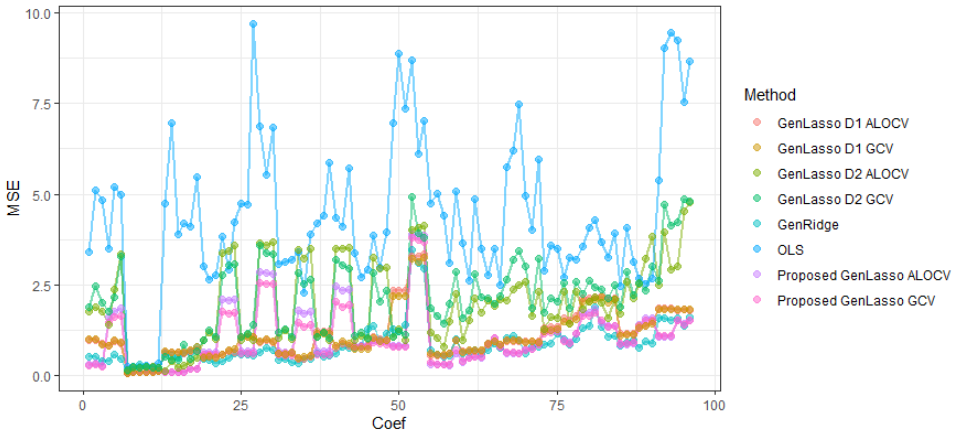
In general, the MSE values of our proposed methods for both ALOCV and GCV were lower compared to OLS and generalized lasso with a single penalty matrix for both using ALOCV and GCV, as shown in Figure 27. The MSE values of our proposed methods were slightly higher in some coefficients corresponding to the regions in cluster 2, especially in Case 1 (Figure 27(a)). In summary, the estimated coefficients of our proposed method are closer to the true coefficients compared to other methods. We can also see that the performance in estimation of the model coefficients of the generalized lasso was not much different between ALOCV and GCV.



(a)

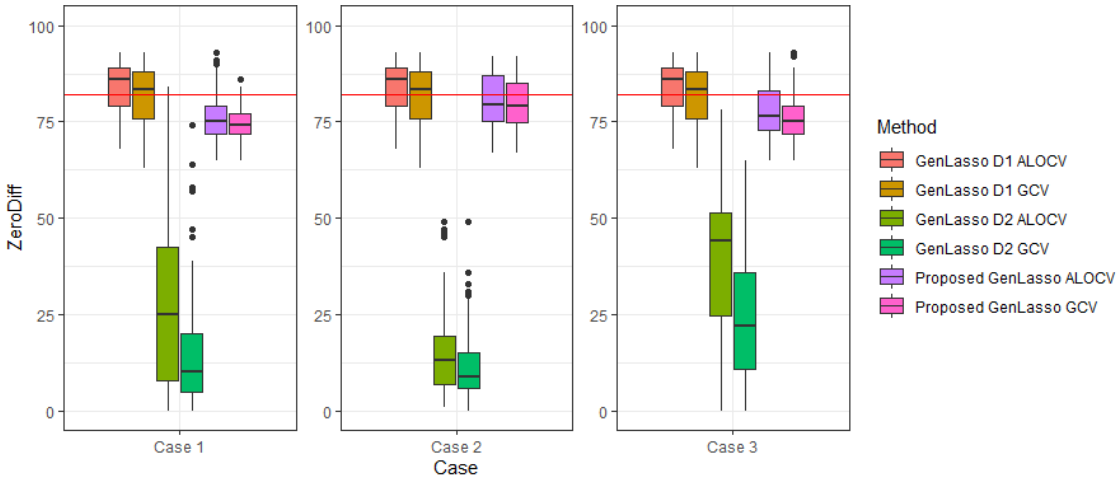


(b)

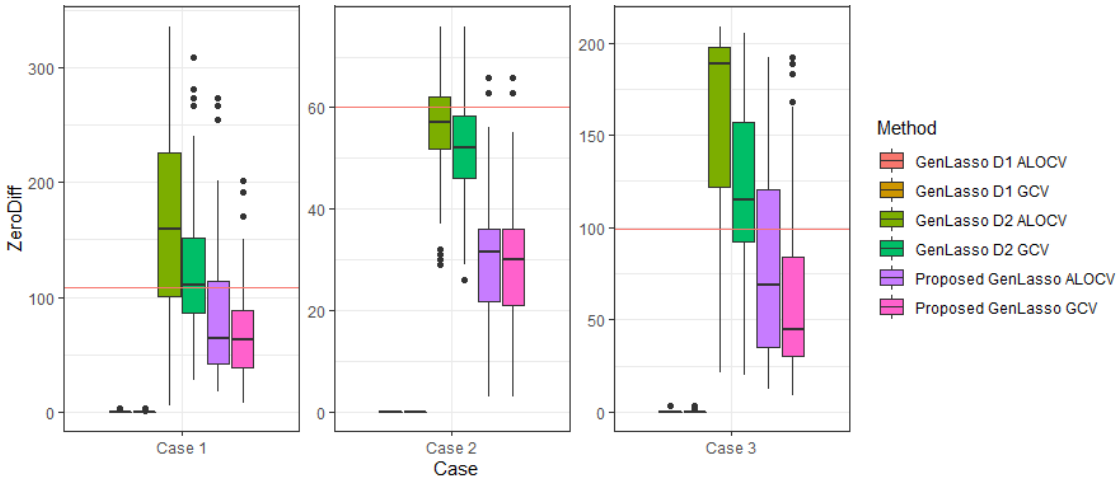


(c)

Figure 27. Line-plots of MSE for each coefficient from 100 replications for each method in (a) Case 1, (b) Case 2, and (c) Case 3



(a)



(b)

Figure 28. Boxplots of the number of edges estimated to have zero differences: (a) for pooling regions and (b) for pooling categories

Since OLS and generalized ridge can't be applied in pooling regions and pooling categories, we only display the number of edges that were estimated to have zero differences in Figure 28 for the proposed generalized lasso using ALOCV and GCV, and the generalized lasso with a single penalty matrix $\boldsymbol{D}_1$ or $\boldsymbol{D}_2$ by using ALOCV and GCV. In all cases, the number of edges between regions that were estimated to have zero differences by our proposed methods was close to the true number of zero-difference edges, 82, as is seen from Figure 28(a). These results were very competitive with the results of generalized lasso with a single penalty matrix $\boldsymbol{D}_1$ as the baseline. The estimated number by the generalized lasso with a single penalty matrix $\boldsymbol{D}_2$ was too small. In Table 9, we summarized the accuracy performances for our proposed methods and the generalized lasso with a single penalty matrix $\boldsymbol{D}_1$ for pooling regions. We obtained higher $IEDA$ values in pooling regions for all cases by our proposed methods, which very close to the generalized lasso with a single penalty matrix $\boldsymbol{D}_1$. The highest $IEDA$ of our proposed methods was

provided in Case 2, in which all categorical predictors were ordered. We can also see, in each case, that ALOCV and GCV provided almost similar $IEDA$ values.

Table 9. Result of $\overline{Sens^E}$, $\overline{PPV^E}$, $IEDA$ of generalized lasso each case for pooling regions

Case	Criteria	Proposed GenLasso ALOCV	Proposed GenLasso GCV	GenLasso $\boldsymbol{D_1}$ ALOCV	GenLasso $\boldsymbol{D_1}$ GCV
Case 1	$\overline{Sens^E}$	0.856	0.844	0.930	0.910
	$\overline{PPV^E}$	0.923	0.927	0.911	0.915
	$IEDA$	0.887	0.883	0.919	0.910
Case 2	$\overline{Sens^E}$	0.902	0.895	0.930	0.910
	$\overline{PPV^E}$	0.921	0.924	0.911	0.915
	$IEDA$	0.910	0.908	0.919	0.910
Case 3	$\overline{Sens^E}$	0.873	0.857	0.930	0.910
	$\overline{PPV^E}$	0.921	0.926	0.911	0.915
	$IEDA$	0.894	0.889	0.919	0.910

Table 10. Result of $\overline{Sens^E}$, $\overline{PPV^E}$, $IEDA$ of generalized lasso for pooling categories

Case	Criteria	Proposed GenLasso ALOCV	Proposed GenLasso GCV	GenLasso $\boldsymbol{D_2}$ ALOCV	GenLasso $\boldsymbol{D_2}$ GCV
Case 1	$\overline{Sens^E}$	0.378	0.326	0.569	0.444
	$\overline{PPV^E}$	0.518	0.526	0.385	0.406
	$IEDA$	0.394	0.379	0.429	0.402
Case 2	$\overline{Sens^E}$	0.483	0.469	0.783	0.728
	$\overline{PPV^E}$	0.972	0.971	0.842	0.851
	$IEDA$	0.622	0.608	0.805	0.777
Case 3	$\overline{Sens^E}$	0.453	0.375	0.778	0.622
	$\overline{PPV^E}$	0.607	0.643	0.498	0.522
	$IEDA$	0.462	0.427	0.585	0.541

The number of edges between categories that were estimated to have zero differences by our proposed methods tended to be slightly smaller than the true number of zero-difference edges for each case, as is seen from Figure 28(b). The number by the generalized lasso with a single penalty matrix $\boldsymbol{D_2}$ were close to the actual values in Case 2, while they tended to be larger in Case 1 and 3. In contrast, the estimated number by the generalized lasso with a single penalty matrix $\boldsymbol{D_1}$ was too small. We summarized the accuracy performances of our proposed methods and the generalized lasso with a single penalty matrix $\boldsymbol{D_2}$ in Table 10. Generally, our proposed methods provided higher PPV, although smaller sensitivity and

$IEDA$, compared to the single penalty model. We also see that ALOCV and GCV provided almost similar results.

In summary, the proposed generalized lasso with two ℓ_1 penalties can be useful for estimating parameters in the spatially varying coefficient model with numerical and categorical predictor variables, and for pooling regions and pooling categories. It was suggested that the proposed method could be more useful for both purposes of pooling regions and pooling categories than using the model with a single penalty. For determining the optimum model, we can use either ALOCV or GCV because it produces similarly good results.

5.2.4 Real Case Application: House Sales Price in Java Island

We used the data of 441 house sales prices on Java Island, Indonesia, which were randomly collected in January 2021 from the website: www.rumah.com (rumah.com, 2021). The response variable used was house sales price in million IDR. There are nine predictor variables, two on numeric scales and seven on categorical scales. Table 11 provides a list of predictor variables and their scales on the data used.

Table 11. List of predictor variables used with their scale

Code	Variable	Scale
X_1	Building Area (m^2)	Numerical
X_2	Surface Area (m^2)	Numerical
X_3	Number of cars that garage will hold	Ordered categorical (0 to 10 with 11 categories)
X_4	Total number of bedrooms	Ordered categorical (1 to 10 with 10 categories)
X_5	Total number of bathrooms	Ordered categorical (1 to 10 with 9 categories)
X_6	Legality certificate type	Ordered categorical (1.Freehold title > 2.building use rights certificate > 3.deed of sale and purchase > 4.others)
X_7	Electrical power	Ordered categorical (450W to 60,000W with 14 categories)
X_8	Presence or absence of adjacency to highway	Binary (1.absence and 2.presence)
X_9	Building status	Binary (1.old construction and 2.new construction)

We considered six administrative provinces in Java Island as regions in this study, namely Banten, Jakarta, West Java, Central Java, East Java, and Yogyakarta. Figure 29 shows the distribution of the price per m^2 of surface area from the data, in which the majority of houses with higher prices are located around Jakarta and some capital cities.

There are some unobserved categories in the categorical predictors for some provinces. Table 12 summarizes the completeness of categories in the categorical variables for all the provinces. In practice, we can still analyze incomplete categories in each province, as is usual in our proposed method. The difference is, in the $\boldsymbol{X}$ matrix structure, there is a value of 0 in the category and location column corresponds to the incomplete one.

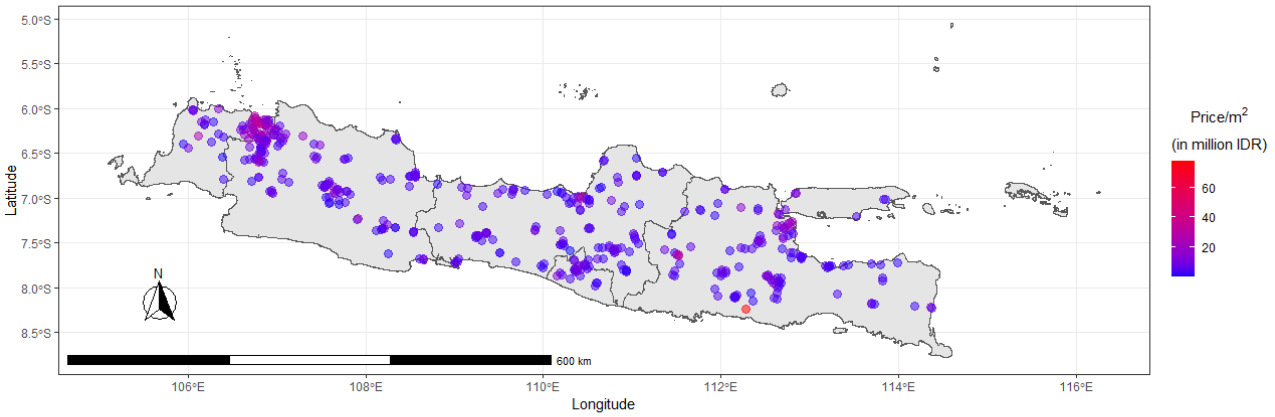


Figure 29. Distribution of house sales price per m^2 (in million IDR) of surface area in Java Island

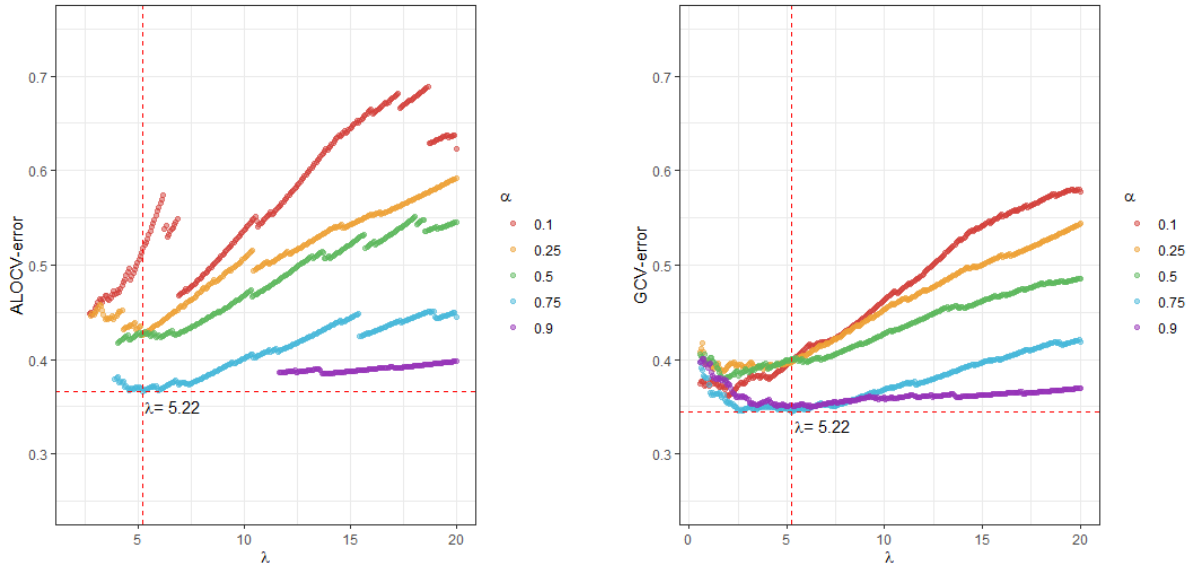
Table 12. Number of observed categories in the categorical variables for each province

Province	Variable						
	X_3	X_4	X_5	X_6	X_7	X_8	X_9
Banten	8	6	6	3	5	2	2
Jakarta	6	8	7	3	9	2	2
West Java	9	9	7	3	9	2	2
Central Java	7	9	8	3	7	2	2
East Java	8	9	8	4	10	2	2
Yogyakarta	4	7	7	3	6	2	2
All provinces	11	10	9	4	14	2	2

We applied the proposed method to the original data that were not fused between categories for each categorical predictor. We constructed $C_j - 1$ dummy variables for j-th categorical predictor variable with C_j categories. We selected the lowest value as the reference category for variables X_3, X_4, X_5, and X_7. For X_6, we chose the category '1. Freehold title' as the reference category because it constitutes the majority. We specified the first category (absence of adjacency to highway and old construction) as the reference category for unordered categorical variables X_8, and X_9, respectively. Then, we arranged the rows of the block-diagonal design matrix $\boldsymbol{X}$ such that the location of observations belongs to the block corresponding to each province. We also included the intercept for each province in the model. Therefore, we have $\boldsymbol{X} \in \mathbb{R}^{441\times288}$, with 48 columns for each province. Furthermore, we have $\boldsymbol{D}_1 \in \mathbb{R}^{288\times288}$ and $\boldsymbol{D}_2 \in \mathbb{R}^{270\times288}$, thus $\widetilde{\boldsymbol{D}} \in \mathbb{R}^{558\times288}$. In the data analysis, we used the log price instead of the original price as the response variable y to make the distribution more symmetric, and standardized the numerical predictors X_1 and X_2.

We specified α as $\alpha = 0.1, 0.25, 0.5, 0.75,$ and 0.9, and for each α, we selected the optimum λ using the ALOCV and GCV. Figure 30 shows the dot plots of the ALOCV errors and GCV errors for each specified α, with $\lambda \leq 20$. In Figure 30(a), the ALOCV error shown discontinuous behavior at some specific λ. This figure also shows that ALOCV was unfeasible on some small λ values and hence the plot is truncated. We can see that the minimum ALOCV error lies on the line with $\alpha = 0.75$, and that $\lambda =$

5.22 was selected. In Figure 30(b), the GCV error pattern is smoother than the ALOCV error. GCV error values could be obtained even for small λ values. Based on the figure, we obtained a minimum GCV error of $\alpha = 0.75$, and $\lambda = 5.22$, which was the same result as ALOCV, even though the GCV error in these results was smaller than the ALOCV error. Therefore, we obtained the optimum model for the generalized lasso (5.3) with $\alpha = 0.75$ and $\lambda = 5.22$.



(a) (b)

Figure 30. (a) ALOCV errors and (b) GCV errors dot plots for specified α and λ

Based on these values, we obtained the parameter estimates of the model in Table 13, in which pooled provinces and categories gave common estimates, and the last row of the table displays estimates when all provinces are pooled. We obtained estimates of the intercept formed into 2 clusters, with coefficients in Banten, Jakarta, and West Java were higher than in other provinces. While the coefficient estimates of the building area (X_1) and surface area (X_2) common to all provinces. The effect of the number of cars that garage will hold (X_3) in East Java and Yogyakarta was slightly different from that in other provinces. For the total number of bedrooms (X_4), the coefficient value tended to increase in line with the increase in the number of bedrooms, and the estimate common to all provinces was provided when the house consisted of more than six bedrooms. We can see the effect of the total of bedrooms in Banten and Jakarta was higher from other provinces. The effect of the total of bathrooms (X_5) was separated to be two: Banten, Jakarta, and West java; Central Java, East Java, and Yogyakarta. We found that the legality certificate type (X_6) had no effect on all categories and provinces. The effect of electrical power (X_7) was separated to be two: Banten, Jakarta, and West java; Central Java, East Java, and Yogyakarta. Finally, we found that the presence or absence of adjacency to highway (X_8) and building status (X_9) had no effect on all categories and provinces, except for the status of buildings in East Java, with a fairly small negative effect for the category of new buildings. Negative estimates for some categories indicate that the predicted house sales price will be lower if the house falls into the corresponding category. Furthermore,

for unobserved combinations of categories in each province, the price can be predicted using the model estimated by the generalized lasso.

In comparison, the results of the model in which all provinces were pooled yielded several different estimates and characteristics. Some of the estimated coefficients had the same value as our proposed model, while in our proposed model the estimated coefficients differed in several provinces. This indicates that our proposed model obtained comprehensive results covering every province condition.

5.2.5 Conclusion

Based on this study, the generalized lasso with two ℓ_1 penalties can be a useful method to model spatially varying coefficients for the region-based data, which consists of numerical and categorical predictors. Based on our simulation, our proposed method can estimate and cluster the coefficients well. In the application to the house sales price data in Java Island, Indonesia, estimated coefficients for some categories in categorical predictors could be pooled, and some of the estimated coefficients among provinces were pooled.

Table 13. Coefficient estimations with pooled categories for a common estimate

Province	Int.	X_1	X_2	X_3 {1}	X_3 {2}	X_3 {3,4}	X_3 {5}	X_3 {6}	X_3 {7}	X_3 {8}	X_3 {9}	X_3 {10}	X_4 {2}	X_4 {3}	X_4 {4}	X_4 {5}	X_4 {≥6}	X_5 {2}	X_5 {3}
Banten	5.993	8.097	1.028	0	0.125	0.194	0.194	0.194	0.194	-0.042	0.000	-0.042	0	0.256	0.568	0.568	0.462	0.298	0.452
Jakarta	5.993	8.097	1.028	0	0.194	0.194	0.194	0.194	0.194	-0.042	-0.042	-0.042	0	0.256	0.568	0.568	0.462	0.298	0.452
West Java	5.993	8.097	1.028	0	0.180	0.194	0.194	0.194	0.194	-0.042	-0.042	-0.042	0	0.256	0.348	0.348	0.462	0.393	0.452
Central Java	5.864	8.097	1.028	0	0.194	0.194	0.801	0.194	0	0	-0.042	-0.042	0	0.256	0.348	0.348	0.462	0.452	0.452
East Java	5.864	8.097	1.028	0	0.397	0.194	0.801	0.194	-0.688	-0.688	-0.688	-0.688	0	0.351	0.348	0.348	0.462	0.452	0.452
Yogyakarta	5.864	8.097	1.028	0	0.194	0.194	0.801	0.194	-0.042	-0.042	-0.042	-0.042	0	0.256	0.348	0.348	0.462	0.452	0.452
All ($\lambda = 1.61$)	6.114	5.522	1.122	0	0.219	0.219	0.403	0.082	-0.189	-0.189	-0.189	-0.189	-0.249	-0.014	0.054	0.164	0.258	0.406	0.548

Province	X_5 {4}	X_5 {5}	X_5 {6,7,8}	X_5 {10}	X_6 {2}	X_6 {3}	X_6 {4}	X_7 {900}	X_7 {1300}	X_7 {2200}	X_7 {3500}	X_7 {4400}	X_7 {5500}	X_7 {6600,7700, 10600}	X_7 {11000}	X_7 {16500}	X_7 {≥23000}	X_8 {2}	X_9 {2}
Banten	0.452	0.452	0.452	0.396	0	0	0	-0.018	0.076	0.439	0.762	0.762	0.762	0.762	0.762	0.762	0.762	0	0
Jakarta	0.452	0.452	0.452	0.396	0	0	0	0	0.194	0.752	0.752	0.762	0.762	0.762	0.762	0.762	0.762	0	0
West Java	0.481	0.452	0.452	0.452	0	0	0	-0.018	0.194	0.439	0.762	0.762	0.716	0.716	0.762	0.762	0.762	0	0
Central Java	0.452	0.452	0.452	0.452	0	0	0	0	0.194	0.573	0.573	0.573	0.573	0.716	0.716	0.762	0.747	0	0
East Java	0.452	0.452	0.452	0.452	0	0	0	0	0.194	0.573	0.747	0.747	0.747	0.747	0.747	0.747	0.747	0	-0.103
Yogyakarta	0.452	0.452	0.452	0.452	0	0	0	0	0.194	0.573	0.747	0.747	0.747	0.747	0.747	0.747	0.747	0	0
All ($\lambda = 1.61$)	0.578	0.578	0.486	0.486	-0.159	-0.159	-0.159	0	0.331	0.718	0.968	0.968	1.083	1.083	1.083	1.083	1.083	0.04	-0.228

6 Concluding Remarks

In this study, we extended some generalized lasso applications to spatial data analysis. There are three important points that should attract attention in this research, namely the method for selecting the optimum tuning parameter λ in generalized lasso, the application of generalized lasso for spatial clustering, and the application of generalized lasso for spatial modeling.

On the first point, based on the simulations that we conducted in the situation of spatial clustering, ALOCV was found to be the recommended method for selecting the optimum tuning parameter λ compared to k-fold CV. If there is only one observation in each location, k-fold CV cannot be feasible, while ALOCV performance in detecting edges with zero difference was appropriate. GCV can be a substitution for ALOCV when ALOCV is not computable for quite small λ, or when the noise in the data has large variability.

On the second point, we provided two applications of the generalized lasso for spatial clustering. In the first application, we tried several methods of selecting the optimum tuning parameter λ for spatial clustering in the application to Chicago Crime Data, which consists of one or a few observations at each location. As a result, we obtained a suitable result of spatial clustering by using ALOCV, which selected the tuning parameter close to the one suggested by previous literature. In the second application, we extended the application of the generalized lasso for spatio-temporal clustering analysis. We proposed a modification of the generalized lasso model adopted for spatio-temporal data, which can be separated into the two generalized lasso problems: trend filtering on the temporal scale and fused lasso for spatial clustering. In the trend filtering problem, smoothed temporal pattern is estimated from the average value over all locations at each time point. In the fused lasso problem, in which the average value over all locations has been subtracted from the original responses at each time, clusters are constructed at each time and their relative magnitude can be compared. Therefore, through our proposed method it is possible to see dynamic pattern of clusters as time proceeds. Our simulation study suggested that estimation of temporal and spatial effects using generalized lasso with ALOCV and GCV was comparable or superior in terms of MSE to existing regularization methods such as lasso, ridge, and generalized ridge. Also, we showed that the generalized lasso with ALOCV provided higher $IEDA$, the accuracy of detecting edges with non-zero difference. In addition, our simulation study suggested that a common tuning parameter over all time points was preferable in spatial clustering. As a real data application, we applied the proposed method for detecting dynamic pattern of clusters on Covid-19 data in Japan. We obtained information on the clusters of prefectures, and how they are merged or dissolved.

On the third point of generalized lasso for spatial modeling, we provided two applications. In the first application, we applied the generalized lasso to fit the spatially varying coefficient model and to cluster regional effects of socio-economics factors that affect the Covid-19 case in Java Island, Indonesia. In this application, we applied two schemes of grouping regencies: regions by province, and regions defined by K-means clustering of adjacent regencies and Voronoi tessellation. We found that the poverty variable had no effect generally, while the HDI had different effect on the infection rate of Covid-19 between regions on Java Island, that is, the western part of Java Island would have relatively higher HDI effect than the eastern part of Java Island. In the second application, we proposed a generalized lasso with two ℓ_1 penalties to fit a spatially varying coefficient model with numerical and categorical predictor variables.

When categorical predictors are involved in the model, the two types of penalties are used: fusion of categories within one categorical predictor in one region, and fusion of adjacent regions for some categories within one categorical predictor. Therefore, in this setting we defined two penalty matrices: $\boldsymbol{D_1}$ for pooling regions and $\boldsymbol{D_2}$ for pooling categories. Based on our simulation study, our proposed method could estimate coefficients well for pooling regions and pooling categories, even if the model contains different predictor scales on the model, compared to OLS, ridge regression, and the generalized lasso with a single penalty matrix either $\boldsymbol{D_1}$ or $\boldsymbol{D_2}$. In the application to the house sales price data in Java Island, Indonesia, estimated coefficients for some categories in categorical predictors could be pooled, and some of the estimated coefficients could be pooled among provinces.

Finally, we discuss future works. First, this study mainly used the `genlasso` package of R software to solve the generalized lasso problems using the dual path algorithm (Arnold & Tibshirani, 2016). However, we may consider using the coordinate descent algorithm as suggested in Yamamura et al. (2021), which suggested to have better estimation accuracy and speed than the algorithm used in `genlasso`. Second, to detect the spatial clusters in the spread of disease as a task in epidemiology studies, the response variable is often observed as count data. The application of modified generalized lasso for count data was proposed by Choi et al. (2018), to which we have a great attention in our future work.

References

Anselin, L. (1988). *Spatial Econometrics: Methods and Models*. Kluwer.

Anselin, L. (1995). Local Indicators of Spatial Association-LISA. *Geographical Analysis*, *27*(2), 93–115. https://doi.org/10.1111/j.1538-4632.1995.tb00338.x

Arnold, T. B., & Tibshirani, R. J. (2016). Efficient Implementations of the Generalized Lasso Dual Path Algorithm. *Journal of Computational and Graphical Statistics*, *25*(1), 1–27. https://doi.org/10.1080/10618600.2015.1008638

BPS-Statistics of Banten Province. (2020). *Banten Province in figures 2020*.

BPS-Statistics of Daerah Istimewa Yogyakarta Province. (2020). *Daerah Istimewa Yogyakarta in figures 2020*.

BPS-Statistics of DKI Jakarta Province. (2020). *DKI Jakarta Province in figures 2020*.

BPS-Statistics of Jawa Barat Province. (2020). *Jawa Barat Province in figures 2020*.

BPS-Statistics of Jawa Tengah Province. (2020). *Jawa Tengah Province in figures 2020*.

BPS-Statistics of Jawa Timur Province. (2020). *Jawa Timur Province in figures 2020*.

Brunsdon, C., Fotheringham, A. S., & Charlton, M. E. (1996). Geographically Weighted Regression: A Method for Exploring Spatial Nonstationarity. *Geographical Analysis*, *28*(4), 281–298. https://doi.org/10.1111/j.1538-4632.1996.tb00936.x

Chen, Y., Ong, J. H. Y., Rajarethinam, J., Yap, G., Ng, L. C., & Cook, A. R. (2018). Neighbourhood level real-time forecasting of dengue cases in tropical urban Singapore. *BMC Medicine*, *16*(1), 129. https://doi.org/10.1186/s12916-018-1108-5

Chicago Police Department. (2014). *City of Chicago Data Portal, Crimes—2001 to Present*.

Choi, H., Song, E., Hwang, S., & Lee, W. (2018). A modified generalized lasso algorithm to detect local spatial clusters for count data. *AStA Advances in Statistical Analysis*, *102*(4), 537–563. https://doi.org/10.1007/s10182-018-0318-7

Cliff, A. D., & Ord, J. K. (1973). *Spatial Autocorrelation*. Pion.

Gelfand, A. E., Kim, H.-J., Sirmans, C. F., & Banerjee, S. (2003). Spatial Modeling with Spatially Varying Coefficient Processes. *Journal of the American Statistical Association*, *98*(462), 387–396. https://doi.org/10.1198/016214503000170

Gertheiss, J., & Tutz, G. (2010). Sparse modeling of categorial explanatory variables. *The Annals of Applied Statistics*, *4*(4). https://doi.org/10.1214/10-AOAS355

Green P, & Silverman B. (1994). *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*. Chapman & Hall/CRC Press.

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*, *12*(1), 55. https://doi.org/10.2307/1267351

Huang, J. S. (1984). THE AUTOREGRESSIVE MOVING AVERAGE MODEL FOR SPATIAL ANALYSIS. *Australian Journal of Statistics*, *26*(2), 169–178. https://doi.org/10.1111/j.1467-842X.1984.tb01231.x

Huang, L., Kulldorff, M., & Gregorio, D. (2007). A Spatial Scan Statistic for Survival Data. *Biometrics*, *63*(1), 109–118. https://doi.org/10.1111/j.1541-0420.2006.00661.x

Ishioka, F., Kawahara, J., Mizuta, M., Minato, S., & Kurihara, K. (2019). Evaluation of hotspot cluster detection using spatial scan statistic based on exact counting. *Japanese Journal of Statistics and Data Science*, *2*(1), 241–262. https://doi.org/10.1007/s42081-018-0030-6

Kamenetsky, M. E., Lee, J., Zhu, J., & Gangnon, R. E. (2022). Regularized spatial and spatio-temporal cluster detection. *Spatial and Spatio-Temporal Epidemiology*, *41*, 100462. https://doi.org/10.1016/j.sste.2021.100462

Kim, S.-J., Koh, K., Boyd, S., & Gorinevsky, D. (2009). ℓ_1 Trend Filtering. *SIAM Review*, *51*(2), 339–360. https://doi.org/10.1137/070690274

Kulldorff, M. (1997). A spatial scan statistic. *Communications in Statistics - Theory and Methods*, *26*(6), 1481–1496. https://doi.org/10.1080/03610929708831995

Meijer, R. (2010). Efficient approximate leave-one-out cross-validation for ridge and lasso [MSc Thesis]. Delft University of Technology.

Ministry of Health, L. and W. (2021, September 6). *Current situation in Japan*. https://www.mhlw.go.jp/stf/covid-19/kokunainohasseijoukyou_00006.html.

Moran, P. A. P. (1948). The interpretation of statistical maps. *Journal of the Royal Statistical Society, Series B*, *10*(2), 243–251.

National Statistics Center. (2016, January 3). *Publication of counted and indexed lists of combined adjacent blocks of prefectures in Japan (in Japanese)*. https://www.nstac.go.jp/services/prefcomp.html.

Padilla, C. M., Deguen, S., Lalloue, B., Blanchard, O., Beaugard, C., Troude, F., Navier, D. Z., & Vieira, V. M. (2013). Cluster analysis of social and environment inequalities of infant mortality. A spatial study in small areas revealed by local disease mapping in France. *Science of The Total Environment*, *454–455*, 433–441. https://doi.org/10.1016/j.scitotenv.2013.03.027

Pemerintah Provinsi Banten. (2020, October 5). *Perkembangan COVID-19 Provinsi Banten*. https://infocorona.bantenprov.go.id/maps.

Pemerintah Provinsi D.I. Yogyakarta. (2020, October 5). *Data terkait COVID-19 di DI Yogyakarta*. https://corona.jogjaprov.go.id/data-statistik.

Pemerintah Provinsi DKI Jakarta. (2020, October 5). *Open data COVID-19 Provinsi DKI Jakarta*. https://riwayat-file-covid-19-dki-jakarta-jakartagis.hub.arcgis.com/.

Pemerintah Provinsi Jawa Barat. (2020, October 5). *Sebaran kasus COVID-19 di Jawa Barat*. https://pikobar.jabarprov.go.id/distribution-case.

Pemerintah Provinsi Jawa Tengah. (2020, October 5). *Tabel sebaran COVID-19 Jawa Tengah*. https://corona.jatengprov.go.id/data.

Pemerintah Provinsi Jawa Timur. (2020, October 5). *Peta sebaran COVID-19 Jatim*. http://infocovid19.jatimprov.go.id/.

Portal Site of Official Statistics of Japan (e-Stat). (2021, October 24). *Population Census 2020*. https://www.e-stat.go.jp/. arXiv:1801.10243

Rad, K. R., & Maleki, A. (2018). *A scalable estimate of the extra-sample prediction error via approximate leave-one-out*.

Rad KR, Zhou W, & Maleki A. (2020). Error bounds in estimating the out-of-sample prediction error using leave-one-out cross validation in high-dimensions. *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, 108.

Rahardiantoro, S., & Sakamoto, W. (2021). Clustering Regions Based on Socio-Economic Factors Which Affected the Number of COVID-19 Cases in Java Island. *Journal of Physics: Conference Series*, *1863*(1), 012014. https://doi.org/10.1088/1742-6596/1863/1/012014

Rahardiantoro, S., & Sakamoto, W. (2022a). Optimum Tuning Parameter Selection in Generalized lasso for Clustering with Spatially Varying Coefficient Models. *IOP Conference Series: Earth and Environmental Science*, *950*(1), 012093. https://doi.org/10.1088/1755-1315/950/1/012093

Rahardiantoro, S., & Sakamoto, W. (2022b). Spatially Varying Coefficient Modeling of Numerical and Categorical Predictor Variables in the Generalized Lasso. *Journal of Environmental Science for Sustainable Society*, *11*(Supplement PP05), 16–19.

Rahardiantoro, S., & Sakamoto, W. (2023). Spatio-temporal clustering analysis using generalized lasso with an application to reveal the spread of Covid-19 cases in Japan. *Comput Stat*. https://doi.org/10.1007/s00180-023-01331-x

Stone M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 111–147.

Takaya, H. (2021). *COVID-19 dataset in Japan, Kaggle Dataset*. https://www.kaggle.com/lisphilar/covid19-dataset-in-japan.

Takemura Y, Ishioka F, & Kurihara, K. (2021). Detection of spatial clusters with high-risk regions by using restricted hierarchical structure. In *Bulletin of the Computational Statistics of Japan* (Vol. 34, Issue 1, pp. 23–43).

Tamura K, Puett R C, Hart J E, Starnes H A, Laden F, & Troped P J. (2014). Spatial clustering of physical activity and obesity in relation to built environment factors among older women in three U.S. states. *BMC Public Health*, *14*(1322).

Tango, T. (2008). A Spatial Scan Statistic with a Restricted Likelihood Ratio. *Japanese Journal of Biometrics*, *29*(2), 75–95. https://doi.org/10.5691/jjb.29.75

Tango, T., & Takahashi, K. (2005). A flexibly shaped spatial scan statistic for detecting clusters. *International Journal of Health Geographics*, *4*(1), 11. https://doi.org/10.1186/1476-072X-4-11

Tibshirani R. (1996). Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol*, *58*(1), 267–288.

Tibshirani, R. J., & Taylor, J. (2011). The solution path of the generalized lasso. *The Annals of Statistics*, *39*(3). https://doi.org/10.1214/11-AOS878

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., & Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(1), 91–108. https://doi.org/10.1111/j.1467-9868.2005.00490.x

Tibshirani, R., & Wang, P. (2008). Spatial smoothing and hot spot detection for CGH data using the fused lasso. *Biostatistics*, *9*(1), 18–29. https://doi.org/10.1093/biostatistics/kxm013

van Wieringen, W. N. (2015). Lecture notes on ridge regression. arXiv:1509.09169

Wang, S., Zhou, W., Maleki, A., Lu, H., & Mirrokni, V. (2018). Approximate Leave-One-Out for High-Dimensional Non-Differentiable Learning Problems. arXiv:1810.02716

World Health Organization (WHO). (2021, October 23). *WHO Coronavirus (COVID-19) Dashboard*. https://covid19.who.int/table.

Xin, B., Kawahara, Y., Wang, Y., Hu, L., & Gao, W. (2016). Efficient Generalized Fused Lasso and Its Applications. *ACM Transactions on Intelligent Systems and Technology*, *7*(4), 1–22. https://doi.org/10.1145/2847421

Yamamura, M., Ohishi, M., & Yanagihara, H. (2021). Spatio-temporal adaptive fused lasso for proportion data. In: Czarnowski, I., Howlett, R.J., Jain, L.C. (eds), *Intelligent Decision Technologies. Smart Innovation, Systems and Technologies*, vol 238 (pp. 479–489), Springer, Singapore. https://doi.org/10.1007/978-981-16-2765-1_40

Yang, T., Liu, J., Gong, P., Zhang, R., Shen, X., & Ye, J. (2016). Absolute fused lasso and its application to genome-wide association studies. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1955–1964.

Zhao, Y., & Bondell, H. (2020). Solution paths for the generalized lasso with applications to spatially varying coefficients regression. *Computational Statistics & Data Analysis*, *142*, 106821. https://doi.org/10.1016/j.csda.2019.106821

Zou, B., Peng, F., Wan, N., Mamady, K., & Wilson, G. J. (2014). Spatial Cluster Detection of Air Pollution Exposure Inequities across the United States. *PLoS ONE*, *9*(3), e91917. https://doi.org/10.1371/journal.pone.0091917

Acknowledgements

First of all, I would like to express my sincere gratitude to Professor Sakamoto Wataru for his warm guidance and insightful advice throughout my Doctor Course studies at Okayama University. My appreciation goes to for useful advice and kindness, so that my research and writing becomes more impressive. He has inspired me to be an independent researcher and set an excellent example as a brilliant statistician. Even though I took an additional year in my doctoral studies, he was still very caring and responsible for the completion of my studies. I really appreciate the highest for the support.

I am grateful to Professor Iizuka Masaya and Professor Ishioka Fumio as my co-supervisors, and Associate Professor Yamamoto Michio of Osaka University as my past co-supervisor, who always give positive energy and optimism. I would like to thank to all member of Environmental Statistics Laboratory, especially to Mr. Tanishima Koki, Mr. Hanai Kekeru, and Ms. Zhao Bolin, who always help me to adapt to life in Japan, both inside and outside the university. I also would like to thank to JICA for supporting my research and development during my study.

Last but not least, I must acknowledge my beloved family. To my parents, thank you for always being so supportive. To my wife and my daughter, I wish to thank them for helping and supporting my university life. In particular to my wife, Maya, thank you for your unwavering love, continuing support, and encouragement throughout this process. No words can express my appreciation for your company and faith in me.