STATISTICAL INFERENCE FOR STRUCTURED SPATIAL AND TEMPORAL POINT DATA

A Dissertation

by

LIHAO YIN

Submitted to the Graduate and Professional School of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Huiyan Sang
Committee Members,	Alaa Elwany
	David Jones
	Mikyoung Jun
Head of Department,	Brani Vidakovic

May 2022

Major Subject: Statistics

ABSTRACT

The availability of large-scale spatial and temporal data has fueled increasing interest in statistical modelling and analysis. With the recent development of data collection and data storage techniques, the observation scopes can sometimes involve a extremely vast range or an explosive amount of cases. Then this always leads to an inevitable focus that there tend to be some heterogeneous properties among observations. Thus, the research was conducted to explain the variability in spatial or temporal data considering the correlation of observations.

We first considered the intensity estimation problem for large spatial point patterns on complex domains in $\mathbb{R}^2$ (e.g., domains with irregular boundaries, sharp concavities, and/or interior holes due to geographic constraints) and linear networks, where many existing spatial point process models suffer from the problems of “leakage" and computation. We proposed an efficient intensity estimation algorithm to estimate the spatially varying intensity function and to study the varying relationship between intensity and explanatory variables on complex domains. The method is built upon a graph regularization technique and hence can be flexibly applied to point patterns on complex domains such as regions with irregular boundaries and holes, or linear networks. An efficient proximal gradient optimization algorithm is proposed to handle large spatial point patterns. Numerical studies were conducted to illustrate the performance of the method. Besides, we apply the method to study and visualize the intensity patterns of the accidents on the Western Australia road network, and the spatial variations in the effects of income, lights condition, and population density on the Toronto homicides occurrences.

In addition, the spatial inhomogeneity occurred in various scenarios, especially for the data laying in a vast-scale space. we further established a spatially adaptive sampling design approach based in an estimation of the spatially varying underlying contamination distribution. This part of research was motivated by an Arsenic exposure data which were collected through drinking water in private wells across the Iowa state. From the public and environmental health management perspective, it is critical to allocate the limited resources to establish an effective arsenic sampling and testing plan for health risk mitigation. we propose a statistical regularization method to automatically detect spatial clusters of the underlying contamination risk from the currently available private well arsenic testing data in the USA, Iowa. This approach allows us to develop a sampling design method that is adaptive to the changes in the contamination risk across the identified clusters.

Finally, we further looked into the cluster issues in structured temporal point data. How to cluster event sequences from heterogeneous point processes is a challenging task, especially when event sequences are repeatedly observed and associated with multiple event types. To solve this problem, we proposed an efficient model-based clustering framework, based on a novel multivariate mixture of functional point processes (MFPP). The proposed model generated event sequences from a multi-level log-Gaussian Cox process, which allows to uncover complex inner patterns among sequences, by imposing multiple latent random effects. We prove the identifiability of our mixture model and developed an effective semi-parametric Exponential-Solution (ES) algorithm to the proposed model. The effectiveness of the proposed framework is demonstrated through simulation studies and real data analyses.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to all those who gave me the possibility to complete this dissertation. I want to appreciate my advisor, Dr. Huiyan Sang, for her guidance, patience and support throughout this research. Thank my committee members, Dr. Alaa Elwany, Dr. David Jones and Dr. Mikyoung Jun for their advice in this dissertation.

I also thank Dr. Ganggang Xu, Dr. Yongtao Guan and Dr. Suise Dai for their valuable advice and professional help. Their suggestions and ideas helped me tremendously in my research and writing of this dissertation. Thank Dr. Eun Sug Park for her support.

Thanks also to my friends and the department faculty and staff for making my time at Texas A&M University a great experience. Especially I want to thank Dr. Jianhua Huang, who provided me with the access to studying in Texas A&M University.

Finally, I would like to thank my parents for their continuing encouragement, and my girlfriend, Shikun Wang for her everlasting love.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis (or) dissertation committee consisting of Professor Huiyan Sang of the Department of Statistics.

The data analyzed in Chapter III was provided by Dr. Susie Dai. The part of analysis in Chapter III was contributed by Professor Huiyan Sang.

All other work conducted for the thesis (or) dissertation was completed by the student independently.

Funding Sources

Graduate study was supported by a fellowship from Texas A&M University and a dissertation research fellowship from Professor Huiyan Sang.

NOMENCLATURE

u	Spatial coordinate
s, t	Timestamp
$\lambda(\cdot)$	Intensity function
C	Number of clusters
ω	Cluster indicator
$\mathbb{G}$	Connection graph
$\mathbb{E}$	Edge set
$\mathbb{V}$	Vertex set
H	Oriented incidence matrix corresponding to $\mathbb{E}$
MST	Minimum spanning tree
k-NN	k-Nearest neighbor graph
ϵ-NN	ϵ-Radius nearest neighbor graph

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION

With recent technological advancement, large-scale, high-resolution, and irregularly data concerning real-time human scattered activities can be collected through various venues nowadays. For example, social media platforms such as Twitter and Facebook produce a myriad of user generated content on a daily basis, and Taxi service companies such as Uber and Lyft maintain pick-up/drop-off records of all taxi drivers. The complexity and magnitude of these new data call for new innovative statistical modeling tools. We plan to answer this call by proposing a series of new nonparameteric and semi-parametric point process models for spatial or temporal point patterns of structured human activities. To illustrate, we first introduce two scenarios that the team has access to as motivating examples.

Events on Complex Domains We collected two motivating datasets on complex domains. The first one is the traffic accident locations on the Western Australia road network shown in the right panel of Figure 1.1, where the interest lies in studying the spatial variation of accident occurrences. The left panel in Figure 1.1 shows the homicides locations that occurred in Toronto, where the city boundary has a very irregular shape especially near Toronto islands. The Toronto data set also includes several additional covariates such as the records of average income, night lights, and population density. Therefore, the questions of interest include not only the intensity of crime events but also the relationships between crime intensity and regional characteristics. In particular, for a large city like Toronto, we may expect that such relationships may vary, and in some places rather abruptly, across the study domain.

Sampling Design for Spatial Observations The motivating dataset was collected by the University of Iowa State Hygienic Laboratory from July 1st, 2015 to June 16th, 2020, which contains water Arsenic concentrations in totally 14,570 sampled wells across Iowa state. The spots of wells were highly unevenly distributed (Figure 1.2). Based on the risk categories, we characterize the wells that contain higher than 0.01 mg/L arsenic as high risk wells, and use a binary variable to denote whether a well is at high risk. Obviously the arsenic risk varies in different regions of Iowa.

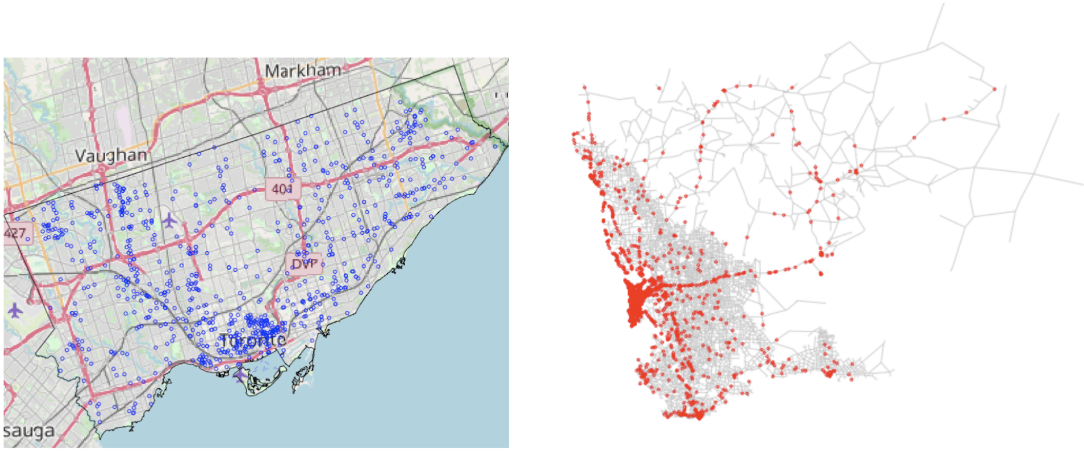

Figure 1.1: Left: map of homicide locations in Toronto during $2000-2014$; Right: traffic networks and road accidents on Western Australia in 2011;

We aimed to detect this spatially varying arsenic exposure risk, and then establish an effective arsenic sampling and testing plan for health risk mitigation.

Repeatedly Observed Event Sequences The motivating data was collected using Twitter API and consists of posting times of 500 university official accounts from April 15, to May 14th, 2021. Figure 1.3 displays posting time stamps of seven selected accounts in five consecutive days. While the daily posting patterns vary across different accounts, the posting date seems to also play an important role. Specifically, all accounts cascade a barrage of postings on April 16th while few postings appear on April 18th. Lastly, each posting is associated with a specific type of activity, namely, tweet, retweet, or reply. Our main interest is to cluster these multi-category, dynamic posting patterns into subgroups.

As we can see in these motivating examples, spatial and temporal point pattern data contain rich information about human activities or events, and have become increasingly prevalent in many disciplines. However, the development of new statistical tools for handling such highly structured data much behind the data availability. The proposed research aims to narrow this gap by achieving following specific aims.

In the Chapter 2, we aim to find solutions to the intensity estimation problem for spatial point

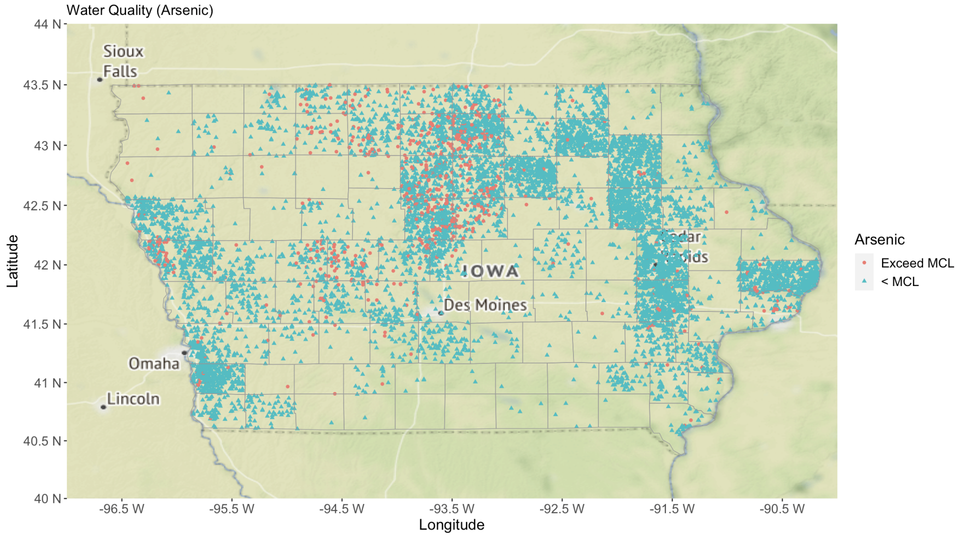

Figure 1.2: Spatial distribution of the Arsenic contamination presence/absence observations.

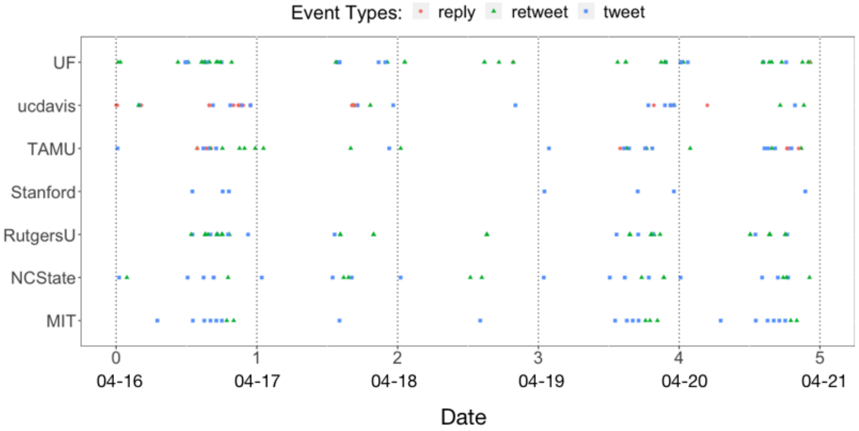

Figure 1.3: The activities of selected accounts on Twitter.

patterns on complex domains in $\mathbb{R}^2$ (e.g., domains with irregular boundaries, sharp concavities, and/or interior holes due to geographic constraints) and linear networks, where many existing spatial point process models suffer from the problems of “leakage" and computation. we developed a simple yet effective approach based on a fused lasso regularization method on a graph for the estimation of piece-wise constant spatial intensity functions. We propose penalties on regression coefficients to encourage sparsity on the differences among regression coefficients that are close in space. The fused lasso methods have gained increasing popularity owning to its flexibility of learning clustered structures. The work in this chapter was previously published and can refer to Yin and Sang (2021) for more details.

In the Chapter 3, we proposed to cluster the Iowa into several sub-regions, so that the arsenic exposure risks are homogeneous in each sub-region. In light of Chapter 2, the varying coefficient model and the graph-fused lasso regularization demonstrated their merits to cluster some irregular space with piece-wise constant properties. In the same way, we proposed a logistic model with spatially varying coeffients for the binary outcomes across the Iowa state. The work in this chapter was previously published and can refer to Yin et al. (2021a) for more details.

In the Chapter 4, we further looked into the cluster issues in structured temporal event sequences which are repeatedly observed. An important goal in studying human activity patterns is to identify user groups displaying similar behavioral patterns. One can further look into each cluster to better understand the underlying cause for certain activity patterns and to make some necessary adjustments (e.g. behavioral interventions). The goal of this project is to develop a unified approach to model human activity patterns and simultaneously form user clusters accordingly. The work in this chapter was previously published and can refer to Yin et al. (2021b) for more details.

2. FUSED SPATIAL POINT PROCESS INTENSITY ESTIMATION WITH VARYING COEFFICIENTS ON COMPLEX CONSTRAINED DOMAINS

2.1 Introduction

Numerous problems in geosciences, social sciences, ecology, and urban planning nowadays involve extensive amounts of spatial point pattern data recording event occurrence. Examples include locations of invasive species, pick-up locations of Taxi trips, addresses of 911 calls, and traffic accidents on roads, to name a few. In many such applications, the primary problem of interest is to characterize the probability of event occurrence. In the presence of additional covariates information, another problem of interest is to study the effect of these covarites on event occurrence probability, considering the spatial dependence of observations. Spatial point process models have been widely used for the analysis of point patterns, in which the intensity function, denoted as $\rho(u)$, is used to describe the likelihood for an event to occur at location u.

In practice, many spatial point patterns data are collected over complex domains with irregular boundaries, peninsulas, interior holes, or network geographical structures. In this chapter, we consider two motivating data examples on such complex domains. The first one is the traffic accident locations on the Western Australia road network shown in the right panel of Figure 1.1, where the interest lies in studying the spatial variation of accident occurrences. The left panel in Figure 1.1 shows the homicides locations that occurred in Toronto, where the city boundary has a very irregular shape especially near Toronto islands. The Toronto data set also includes several additional covariates such as the records of average income, night lights, and population density. Therefore, the questions of interest include not only the intensity of crime events but also the relationships between crime intensity and regional characteristics. In particular, for a large city like Toronto, we may expect that such relationships may vary, and in some places rather abruptly, across the study domain.

Thus far, many methods have been introduced to model the first-order spatially varying intensity function $\rho(u)$. Popular point process models include the spatial Poisson point processes, the log-Gaussian Cox Processes, and the Gibbs point processes. See a review by Møller and Waagepetersen (2007). Intensity estimations of these models are often done using maximum composite likelihoods (Guan, 2006), estimating equations (Guan et al., 2015) or Bayesian inference methods (Leininger et al., 2017; Gonçalves and Gamerman, 2018; Shirota and Banerjee, 2019). Nonparametric methods have also been widely used for estimating the spatially varying intensity functions, including the edge-corrected kernel smoothing estimators by Diggle (1985); Jones et al. (1996), the Voronoi estimator by Barr and Schoenberg (2010) using the inverse of the area of the Voronoi cell for each observed location, and a local likelihood estimation procedure in analogy to geographically weighted regression by Fotheringham et al. (2003).

Yet, statistical analysis of point patterns on complex domains presents severe challenges to many of the classical point process models reviewed above. Mainly, the commonly adopted Euclidean assumption underpinning some of these methods no longer holds for point patterns on complex domains. For example, two locations on a road network that are close by Euclidean distance may actually lie on two separate roads. Moreover, the large data size will aggravate the challenges in modeling point patterns on complex domains. There is a great need to develop spatial point pattern analysis tools that are computationally efficient to solve the so called “leakage" problem encountered on complex domains.

For some particular types of complex domain such as line networks, a number of intensity estimation methods have been developed recently. Kernel estimators of the intensity function on a line network were investigated in McSwiggan et al. (2017); Moradi et al. (2018); Rakshit et al. (2019), adapting the idea of edge-correction using path lengths. Other variations of kernel density estimation methods are reviewed in Baddeley et al. (2020). It is known that kernel estimators are, by nature, more suitable for estimating relatively smooth intensity functions because of the use of smoothing kernel functions. When intensity function exhibits discontinuities and abrupt changes in space, as discussed in Baddeley et al. (2020), piece-wise constant estimators become an appealing alternative as they have a strong adaptivity to changes. One research in this direction is the aforementioned Voronoi estimator by Barr and Schoenberg (2010). However, the method suffers from the high variance in the estimator. To reduce the variance, Moradi et al. (2019) extended it by a bootstrap resample smoothing procedure. Recently, Bassett and Sharpnack (2019) proposed to estimate the density of points on a network as opposed to the intensity function based on a total variation regularization method. While each represents advancements in estimating intensity or density of points, none has incorporated spatial covariates in estimation.

When spatial covariates are available, various methods (Baddeley et al., 2012; McSwiggan, 2019) have been developed to incorporate covariate information with the goal to investigate the effect of spatial covariates on point patterns. However, to the best of our knowledge, there has been very limited work for dealing with varying regression coefficients for spatial point patterns, even in the simpler case where point patterns are observed in the Euclidean space. One notable exception is the work by Pinto Junior et al. (2015), which modeled the regression coefficients as a multivariate Gaussian process in a similar fashion as the spatially varying coefficients (SVC) linear regression model proposed by Gelfand et al. (2003). Despite the model richness and flexibility, the SVC model is known to involve heavy computation in the presence of large spatial data due to the requirement of Metropolis MCMC and the need to invert a large covariance matrix. The intractability of the likelihood function of the spatial Poisson process further aggravates the issue. To address the computation issue, Pinto Junior et al. (2015) partitioned the study region into a small number of subregions according to administrative areas and assumed that latent spatial random effects take constant values within each subregion. However, in some applications, such a pre-determined partition may be unavailable or fail to accurately reflect the complex underlying environmental and geological conditions.

In light of these limitations in the current literature, we develop a simple yet effective approach based on a fused lasso regularization method on a graph for the estimation of piece-wise constant spatial intensity functions. We propose penalties on regression coefficients to encourage sparsity on the differences among regression coefficients that are close in space. The fused lasso methods have gained increasing popularity owning to its flexibility of learning clustered structures. However, to our knowledge, there is limited work that has investigated its performance for point pattern data analysis. In addition, we extend the approach to a piece-wise constant coefficient spatial point process model when explanatory variables are available, which models the varying relationships between point patterns and covariates. We formulate the estimation problem into penalized Poisson-based and Logistic based composite likelihoods optimizations, for which we solve by an efficient proximal gradient algorithm. We tailor the algorithm to utilize spatial graph structures to speed up computations. The choices of graphs play important roles in the modeling and computation of fused lasso problems. We consider various spatial graphs to represent spatial geometry of complex domains and compare their performance. Finally, we introduce this method to the analysis of the Western Australia accident data and the Toronto homicides data. The results of our analysis reveal several interesting clustering patterns of traffic accidents and the spatial crime distribution in relation to a number of key environmental, social, and economic variables.

The chapter is organized as follows. In Section 2.2, we review the basic mathematical formulations and definitions of spatial point processes. We then introduce our method in Section 2.3.1, followed by the computation algorithm in Section 2.3.2. Sections 2.4 and 2.5 include the simulations to illustrate the model performance and the applications to the two real data sets. We offer conclusions and discussions in Section 2.6. Additional implementation details and numerical results are included in Appendix A.

2.2 Preliminaries

2.2.1 Observation Domain

In this study, we consider spatial points on two important types of observation domains. The first type is a bounded domain $D \subset \mathbb{R}^2$ that can be fully covered by finitely many rectangles. The commonly assumed planar window $[a_1, a_2] \times [b_1, b_2]$ is a special case of this type. For any locations u_1, u_2 in a planar window, the Euclidean distance is used to measure the distance between two locations, denoted as $d(u_1, u_2)$. One example of this type is given in Figure 1.1, where the observation domain is the city of Toronto, which has irregular city limit boundaries. For any Borel subset $B \subseteq D$, the Lebesgue measure $|B|$ is the area of B.

In the second type, we assume D is a linear network. Let $[u, v] = \{tu + (1-t)v : 0 \leq t \leq 1\}$ denotes a line segment in the plane with endpoints $u, v \in \mathbb{R}^2$. A linear network is defined as the finite union $D = \cup_{i=1}^{\eta}[u_i, v_i]$ of line segments $[u_1, v_1], \ldots, [u_\eta, v_\eta]$ embedded in the same plane. One commonly used distance $d(u_1, u_2)$ is the shortest-path distance between u_1 and u_2 on the network. For any subset $B \subseteq D$, the measure $|B|$ represents the total length of all segments in B. An example of the line network is shown in the right panel of Figure 1.1, where the road network in the state of Western Australia is drawn in grey lines, and red points mark the traffic accident locations in 2011.

2.2.2 Spatial Point Processes

Let $\mathbf{X}$ be a spatial point process on D with the locally finite property, i.e., the random cardinality $N_{\mathbf{X}}(B) = \#\{u : u \in \mathbf{X} \cap B\}$ is almost surely finite for any $B \subset D$. Assume that, for any bounded $B \subset D$, if there exits a non-negative and locally integrable function $\rho(\cdot) : B \mapsto \mathbb{R}$ such that,

$$E\{N_{\mathbf{X}}(B)\} = \int_B \rho(u)du$$

then $\rho(\cdot)$ is called the intensity function of $\mathbf{X}$. The intensity function is of key interest in point pattern analysis as $\rho(u)|du|$ is interpreted as the approximate probability that an event occurs in the infinitesimal set du.

Poisson point processes are one of the most fundamental and tractable spatial point process models. In practice, $\rho(\cdot)$ is often varying over D, i.e. $\mathbf{X}$ is inhomogeneous and can also depend on some spatial covariates $\mathbf{z}(u)$. In our study, we model the intensity function with a general log-linear form,

$$\rho(u; \boldsymbol{\beta}) = \exp\{\mathbf{z}^T(u)\boldsymbol{\beta}\}, u \in D \tag{2.1}$$

where $\mathbf{z}(u) = \big(z_1(u), \ldots, z_p(u)\big)^T$ is a p-dimensional vector of spatial covariates associated with the spatial location u, and $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of regression parameters.

There are several other popular parametric point process models whose marginal intensity functions take the same log-linear form as in (2.1). The class of Cox process models is one such example. Let $\mathbf{\Lambda} = \{\Lambda(u) : u \in D\}$ denote a real, nonnegative valued random field. If the conditional distribution of $\mathbf{X}$ given $\mathbf{\Lambda}$ is a Poisson process on D with intensity function $\mathbf{\Lambda}$, then X is said to be a Cox process driven by $\mathbf{\Lambda}$. Popular examples of Cox processes models include the Neyman-Scott process and the log Gaussian Cox process. See a review in Chapter 17 of Gelfand et al. (2010).

2.2.3 Composite Likelihoods

To estimate $\boldsymbol{\beta}$ in (2.1), one commonly used method is to construct unbiased estimating equations and obtain estimators by maximizing the corresponding composite likelihoods. The Poisson based composite log-likelihood function (Waagepetersen, 2007) and the logistic based composite log-likelihood function (Baddeley et al., 2014) have been used widely in the literature, which are respectively given by:

$$\ell_{\text{PL}}(\boldsymbol{\beta}) = \sum_{i=1}^{m} \log \rho(u_i; \boldsymbol{\beta}) - \int_D \rho(u; \boldsymbol{\beta}) du \tag{2.2}$$

$$\ell_{\text{LRL}}(\boldsymbol{\beta}) = \sum_{i=1}^{m} \log(\frac{\rho(u_i; \boldsymbol{\beta})}{\delta(u_i) + \rho(u_i; \boldsymbol{\beta})}) - \int_D \delta(u) \log(\frac{\rho(u; \boldsymbol{\beta}) + \delta(u)}{\delta(u)}) du, \tag{2.3}$$

where $\{u_1, \dots, u_m\}$ denotes a set of observed points from a point process, and $\delta(u)$ is a nonnegative real-valued function. When point process is a Poisson process, the Poisson based composite log-likelihood function in (2.2) is identical to the full log-likelihood function. For other point processes models, the use of composite likelihood can be justified by the theory of estimating functions (Guan, 2006). It can be shown (see, e.g., Guan, 2006; Choiruddin et al., 2018) that the estimators obtained by maximizing both Poisson based and logistic based composite log-likelihood are the solution to the two corresponding unbiased estimating equations for $\boldsymbol{\beta}$.

Nevertheless, the composite likelihood based inference produces a less efficient estimator compared with the full likelihood based estimator, due to the loss of information incurred when only using the first-order moment information of the point process. To improve its efficiency, several methods have been developed to carefully select the weights when combining composite likelihood terms (Guan and Shen, 2010). For simplicity, we only consider the unweighted composite likelihoods in the chapter, but remark that the methods can be potentially generalized to the use of weighted composite likelihoods.

In practice, numerical approximations are needed for the composite likelihood inference because both the evaluations of (2.2) and (2.3) involve integral terms. For equation (2.2), Berman and Turner (1992) developed a numerical quadrature method that employs Riemann sum approximation to the integral part. To implement this, the domain D is partitioned into $M - m$ quadrats. More details on how we divide a 2-D bounded domain and a line network can be found in Appendix A. The $M - m$ dummy points, denoted by $\{u_i, i = m + 1, \ldots, M\}$, are then placed at the centroid of each quadrat. The Poisson based composite log likelihood is approximated by

$$\ell_{PL}(\boldsymbol{\beta}) \approx \sum_{i=1}^{M} v_i\{y_i \log \rho(u_i; \boldsymbol{\beta}) - \rho(u_i; \boldsymbol{\beta})\}, \tag{2.4}$$

where $\{u_i \in D, i = 1, \cdots, M\}$ consists of the m observed points and $M - m$ dummy points. v_i is the quadrature weight corresponding to each u_i. We set $v_i = a_i/n_i$, where n_i denotes the total number of observed points and dummy points in the quadrat that u_i resides, and a_i denotes the Lebesgue measure of the quadrat of u_i such that $\sum_{i=m+1}^{M} a_i = |D|$. The working response data becomes $y_i = v_i^{-1}\Delta_i$, where Δ_i is an indicator of whether point i is an observation ($\Delta_i = 1$) or a dummy point ($\Delta_i = 0$).

The Berman-Turner approximation in (2.4) often requires a great amount of dummy points, consequently incurring extra computational cost. Baddeley et al. (2014) showed that the logistic likelihood in (2.3) requires a smaller number of dummy points to perform competitively with the Berman-Turner approximation. The method approximates (2.3) by

$$\ell_{LRL}(\boldsymbol{\beta}) \approx \sum_{i=1}^{m} \log \frac{\rho(u_i; \boldsymbol{\beta})}{\delta(u_i) + \rho(u_i; \boldsymbol{\beta})} + \sum_{i=m+1}^{M} \log \frac{\delta(u_i)}{\delta(u_i) + \rho(u_i; \boldsymbol{\beta})} \tag{2.5}$$

where the integration term is calculated by Monte Carlo integration, and the dummy points are drawn from a Poisson point process over D with an intensity function $\delta(u)$ that is independent from $\mathbf{X}$. Applying the Campbell's formula (Moller and Waagepetersen, 2003), it is straightforward to show that the expectation of the second term in (2.5) equals to the integral part in (2.3). We follow the suggestion of Baddeley et al. (2014) and choose $\delta(u) = (M - m)/|D|$ in our numerical studies.

2.3 Methodology

2.3.1 Spatially Varying Coefficient Models

A traditional way to model the log-linear term of the intensity function is to treat regression coefficients as constants in space as in (2.1). In the proposed model, we are interested in estimating a piece-wise constant intensity function in an intercept-only log-linear model or detecting clustering patterns in $\boldsymbol{\beta}$ when covariates are available. Below, we introduce a varying coefficient log-linear intensity model (SVCI) for spatial point processes via a graph regularization method.

To elaborate, suppose a set of spatial points is observed at locations $u_1, \ldots, u_m \in D$. We assume that these spatial points are a realization from a point process $\mathbf{X}$ with an intensity function $\rho(u)$ that depends on the p-dimensional spatial explanatory variables $\mathbf{z}(u) = \{z_1(u), \ldots, z_p(u)\}$. As an extension of the constant coefficients regression model, we assume that the regression coefficients are spatially varying across D, denoted as $\boldsymbol{\beta}(u) = \{\beta_1(u), \ldots, \beta_p(u)\}^T$. The spatially varying coefficient models inherit the simplicity and easy interpretation of the traditional log-linear model in (2.1), yet they still enjoy great flexibility that allows practitioners to investigate locally varying relationships among variables.

Let $\boldsymbol{\beta}_k = \big(\beta_k(u_1), \ldots, \beta_k(u_m)\big)^T$ denote the vector of regression coefficients associated with the k-th covariate, for $k = 1, \ldots, p$. We assume that each $\boldsymbol{\beta}_k$ has its own spatially clustered pattern and is a piece-wise constant function on D; the coefficients are homogeneous in the same spatial cluster and varying across different clusters. In many spatial applications involving point patterns such as traffic accidents, crime locations and pick-up/drop-off locations of Taxi trips, it is desirable to consider spatially contiguous clustering configurations such that only adjacent locations are clustered together. This way, the practitioners can detect discontinuities across boundaries and easily interpret the detected clusters as local regions to facilitate subsequent regional analysis.

Before introducing our regularization method, we formally define spatially contiguous cluster of points using the notion of connected components in graph theory. Consider an undirected graph denoted as $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V} = \{u_i \in D, i = 1, \ldots, M\}$ is the set of vertices which in our case consists of both observed points and dummy points, and $\mathbb{E}$ is the edge set consisting of a subset of $\{(u_i, u_j) : u_i, u_j \in \mathbb{V}\}$. In graph theory, a graph $\mathbb{G}$ is said to be connected if for any two vertices there exists a path between them. A subgraph $\mathbb{G}_s$ is called a connected component of $\mathbb{G}$ if it is connected and there is no path between any vertex in $\mathbb{G}_s$ and any vertex in $\mathbb{G} \setminus \mathbb{G}_s$, where $\mathbb{G} \setminus \mathbb{G}_s$ denotes the subgraph on the set $\mathbb{V} \setminus \mathbb{V}_s$. Now we can define spatially contiguous clusters as the connected components of a graph $\mathbb{G}$. As a result, a spatially contiguous partition of $\mathbb{V}$ is defined as a collection of disjoint connect components such that the union of vertices is $\mathbb{V}$.

This motivates us to construct a graph based regularization model, which permits contiguous cluster identifications of regression coefficients for each covariate in the log-linear point process model. Let $\boldsymbol{\beta}_k^* = \big(\beta_k(u_{m+1}), \ldots, \beta_k(u_M)\big)^T$, for $k = 1, \ldots, p$, denote the vector of regression coefficients at the dummy points associated with the k-th covariate. Denote the vector of the stacked regression coefficients at both the observed and dummy points by $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^T, \boldsymbol{\beta}_1^{*,T}, \ldots, \boldsymbol{\beta}_p^T, \boldsymbol{\beta}_p^{*,T})^T \in \mathbb{R}^{pM}$. We estimate $\boldsymbol{\beta}$ by minimizing the penalized negative composite log likelihood objective function:

$$Q(\boldsymbol{\beta}) = -\frac{1}{|D|}\tilde{\ell}_c(\boldsymbol{\beta}) + \sum_{k=1}^{p} \sum_{(i,j)\in\mathbb{E}} P_\lambda(\beta_k(u_i) - \beta_k(u_j)), \tag{2.6}$$

where $\tilde{\ell}_c(\beta)$ is either the approximation of the Poisson based composite log-likelihood or the logistic regression-based composite log-likelihood function with the following expressions:

$$\tilde{\ell}_{PL}(\boldsymbol{\beta}) = \sum_{i=1}^{M} v_i\{y_i \log \rho(u_i; \boldsymbol{\beta}(u_i)) - \rho(u_i; \boldsymbol{\beta}(u_i))\}, \tag{2.7}$$

$$\tilde{\ell}_{LRL}(\boldsymbol{\beta}) = \sum_{i=1}^{m} \log \frac{\rho\big(u_i; \boldsymbol{\beta}(u_i)\big)}{\delta(u_i) + \rho\big(u_i; \boldsymbol{\beta}(u_i)\big)} + \sum_{i=m+1}^{M} \log \frac{\delta(u_i)}{\delta(u_i) + \rho\big(u_i; \boldsymbol{\beta}(u_i)\big)}. \tag{2.8}$$

The second term in the objective function (2.6) adds a graph pairwise fused regularization to the negative composite log-likelihood function. $\mathbb{E}$ is the edge set of a graph $\mathbb{G}$, and $(i,j) \in \mathbb{E}$ implies that there is an edge in $\mathbb{E}$ connecting the points at u_i and u_j. $P_\lambda(\cdot)$, a non-negative function tuned by parameter λ, penalizes the pairwise difference of regression coefficients whose corresponding locations are connected by an edge in $\mathbb{E}$. One popular choice is the L_1-penalty,

$$P_\lambda(t) = \lambda \|t\|_1$$

which is often referred to as the graph fused lasso penalty in the literature (Tibshirani et al., 2011; Arnold and Tibshirani, 2016; Li and Sang, 2019). The L_1 penalty encourages sparsity in the pairwise differences between the coefficients of edge-connected locations. As a result, the edges in the graph can be classified into a set that corresponds to the non-zero elements of $|\beta_k(u_i) - \beta_k(u_j)|$, and another set that corresponds to the zero elements of $|\beta_k(u_i) - \beta_k(u_j)|$. The solution of L_1 fused lasso penalty naturally leads to a piece-wise constant estimate of $\boldsymbol{\beta}_k$ for each covariate and hence a well defined spatially contiguous partition of the vertices for each regression coefficient function. λ is a non-negative tuning parameter that determines the strength of penalization and ultimately influences the estimated number of clusters. We use the Bayes information criterion (BIC) to select an optimal value of λ (Choiruddin et al., 2021). Specifically, $BIC = -2\tilde{\ell}_c + df \log m$, where $\tilde{\ell}_c$ is the approximated composite log likelihood as in (2.7) and (2.8), m is the number of observations, and df is the degree of freedom of $\hat{\boldsymbol{\beta}}$. Following Tibshirani et al. (2011), df is estimated by the summation of the number of clusters for each regression coefficient $\boldsymbol{\beta}_k$.

We remark that there are other choices of sparsity inducing penalty functions, including adaptive lasso (Zou, 2006), smoothly clipped absolute deviation (SCAD, Fan and Li, 2001), and minimax concave penalty (MCP, Zhang, 2010). There are also other criteria for tuning parameter selection, including Akaike information criterion (AIC), generalized cross-validation (GCV, Golub et al., 1979), and extended Bayesian information criterion (EBIC, Chen and Chen, 2012). In this chapter, we choose to use L_1 penalty together with BIC to demonstrate the utility of our method for its computational simplicity. The method itself can adopt other forms of penalty functions and model-selection criteria which may further improve its performance.

The selection of edge set $\mathbb{E}$ is a key ingredient in our SVCI model by playing two important roles. First, the corresponding graph $\mathbb{G}$ reflects the prior assumptions about the spatial structure and the contiguous constraint of the regression coefficients. In particular, we rely on $\mathbb{G}$ to incorporate the relational information among points on complex constrained domains so that we can relax the Euclidean assumption. Second, as we will explain in Section 2.3.2, the computation speed and storage complexity of the optimization algorithm are largely determined by the structure of $\mathbb{G}$. We seek to construct a graph fused lasso regularization to achieve a good balance between model accuracy and computational efficiency.

For point patterns on a bounded observation domain, one natural choice is to construct a nearest neighbor graph that connects each vertex with its k nearest neighbors (K-NN) or neighbors within a certain radius (r-NN). In practice, the number of neighbors in K-NN or the radius in r-NN needs to be chosen with care to guarantee that $\mathbb{G}$ is a connected graph. It is known in machine learning literature (see, e.g., Shaw and Jebara, 2009) that K-NN graphs can effectively preserve the intrinsic manifold structure of the data. Another approach is the Delaunay triangulation (Lee, 1980), which constructs triangles with a vertex set such that no vertex is inside the circumcircle of any triangle. In practice, edges longer than a certain threshold are removed to ensure the spatial proximity of neighboring vertices. Triangular graphs have also shown their capabilities in preserving complex topological structures of the data. See Lindgren et al. (2011); Mu et al. (2018) for examples. Moreover, when a graph has certain simple structures such as a chain or a tree graph, several recent work (Padilla et al., 2018; Li and Sang, 2019) showed that these simple graph structures enable simplified algorithms to solve the graph fused lasso problem. This motivates us to adopt a similar strategy to replace the original graph by a minimum spanning tree graph, defined as the subgraph that connects all vertices with no cycles and with minimum total edge weights. We will investigate and compare the performance of the proposed SVCI model with different types of graphs in the numerical studies in Section 2.4 and Appendix Section A3.

For point patterns on a linear network, we use an edge set that only connects pairs which are

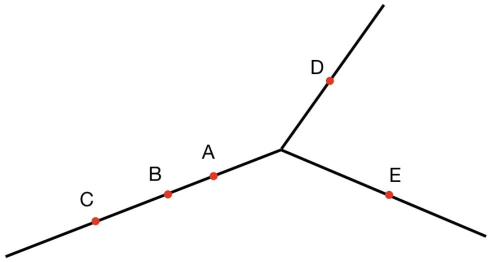

Figure 2.1: A simple illustration of connections in linear networks

natural neighbors. To illustrate how we define natural neighbors, we provide a simple example of a linear network (black segments) and 5 spatial points (red nodes) near an intersection in Figure 2.1. For any interior point such as point B, defined as a point where there exists one other point on each side of the same line, we connect it with its two adjacent points $\{\mathtt{A}, \mathtt{C}\}$. For any boundary point such as point A, defined as a point where there is no other point on the path between it and the intersection point, we connect it with $\{\mathtt{B}, \mathtt{D}, \mathtt{E}\}$, i.e., its adjacent interior point on the same line and its adjacent boundary points on the neighboring lines that share the same intersection.

2.3.2 Computation

Once we construct the edge set $\mathbb{E}$, the objective function in (2.6) can be written in a matrix form, and the estimate of $\boldsymbol{\beta}$ is obtained by solving the following fused lasso optimization problem:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \in \mathbb{R}^{pM}}{\arg\min} \{ -\frac{1}{|D|} \tilde{\ell}_c(\boldsymbol{\beta}) + \lambda \sum_{k=1}^{p} \|\mathbf{H}\boldsymbol{\beta}_k\|_1 \}, \tag{2.9}$$

where $\mathbf{H}$ is an $m'' \times M$ incidence matrix corresponding to the edge set $\mathbb{E}$ with m'' edges. Specifically, for the l-th edge of $\mathbb{E}$ connecting vertices u_i and u_j, the penalty term $|\beta_k(u_i) - \beta_k(u_j)|$ is represented as $|\mathbf{H}_l \boldsymbol{\beta}_k|$, where $\mathbf{H}_l$ is the l-th row of $\mathbf{H}$ and contains only two nonzero elements; 1 at the i-th column index and -1 at the j-th.

The path following type of algorithms (Arnold and Tibshirani, 2016) and alternating direction methods of multipliers (ADMM, Boyd et al., 2011) have been developed to solve graph fused lasso problems. However, the computation of these algorithms can be expensive for a general graph with a large number of nodes and edges. Note the number of nodes in our graph is typically a large number in practice because both the numbers of observations and dummy points are included. It is, therefore, computationally challenging to directly apply these conventional algorithms for the implementation of our model.

We note that the two approximated log composite likelihood functions in (2.7) and (2.8) coincide with the forms of the log likelihood function of a weighted Poisson linear regression and a logistic linear regression, respectively, both of which are concave functions of $\boldsymbol{\beta}$. Below, we propose to combine the proximal gradient method and the alternating direction method of multipliers to solve the convex optimization problem in (2.9). In particular, we take advantage of specific structures of our selected spatial graphs to speed up computation.

Specifically, with the current estimate of the parameters being $\boldsymbol{\beta}^{(t)}$, we follow the proximal gradient method (Beck and Teboulle, 2009) to update the value of $\boldsymbol{\beta}$ iteratively by solving:

$$\boldsymbol{\beta}^{(t+1)} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2} \left\|\boldsymbol{\beta} - \mathbf{R}^{(t)}\right\|_2^2 + \frac{\lambda}{L} \sum_{k=1}^{p} \|\mathbf{H}\boldsymbol{\beta}_k\|_1, \tag{2.10}$$

where L is the local Lipschitz constant of $-\frac{1}{|D|}\tilde{\ell}_c(\boldsymbol{\beta}^{(t)})$, $\nabla\tilde{\ell}_c(\boldsymbol{\beta}^{(t)})$ is the first derivative of $\tilde{\ell}_{PL}(\boldsymbol{\beta})$ or $\tilde{\ell}_{LRL}(\boldsymbol{\beta})$ evaluated at $\boldsymbol{\beta}^{(t)}$, and $\mathbf{R}^{(t)} = \boldsymbol{\beta}^{(t)} + (1/L)\frac{1}{|D|}\nabla\tilde{\ell}_c(\boldsymbol{\beta}^{(t)})$. We can choose L to be the maximum eigenvalue of the Hessian matrix of $-\frac{1}{|D|}\tilde{\ell}_c(\boldsymbol{\beta})$ evaluated at $\boldsymbol{\beta}^{(t)}$.

Now the optimization at each iteration boils down to solving (2.10), for which we propose to use the ADMM algorithm (Wahlberg et al., 2012). By introducing auxiliary variables $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_p\}$, Equation (2.10) is equivalent to:

$$s.t. \quad \boldsymbol{\theta}_k = \mathbf{H}\boldsymbol{\beta}_k, \quad \forall k = 1, \ldots, p$$

Its augmented Lagrangian function is:

$$\frac{1}{2}\left\|\boldsymbol{\beta}-\mathbf{R}^{(t)}\right\|_2^2+\frac{\lambda}{L}\sum_{k=1}^{p}\|\boldsymbol{\theta}_k\|_1+\frac{\gamma}{2}\sum_{k=1}^{p}\|\mathbf{H}\boldsymbol{\beta}_k-\boldsymbol{\theta}_k\|_2^2+\gamma\sum_{k=1}^{p}\mathbf{u}_k^T(\mathbf{H}\boldsymbol{\beta}_k-\boldsymbol{\theta}_k),$$

where $\mathbf{u}=\{\mathbf{u}_1,\ldots,\mathbf{u}_p\}$ are Lagrangian multipliers, and γ is a penalty parameter. ADMM alternately optimizes $\{\boldsymbol{\beta},\boldsymbol{\theta},\mathbf{u}\}$ by solving the following three subproblems:

$$\begin{cases}\boldsymbol{\beta}^{(t+1)}=\arg\min\limits_{\boldsymbol{\beta}}\{\left\|\boldsymbol{\beta}-\mathbf{R}^{(t)}\right\|_2^2+\gamma\sum\limits_{k=1}^{p}\|\mathbf{H}\boldsymbol{\beta}_k-\boldsymbol{\theta}_k^{(t)}+\mathbf{u}_k^{(t)}\|_2^2\},\\ \boldsymbol{\theta}_k^{(t+1)}=\arg\min\limits_{\boldsymbol{\theta}_k}\{\frac{\lambda}{L}\|\boldsymbol{\theta}_k\|_1+\frac{\gamma}{2}\|\mathbf{H}\boldsymbol{\beta}_k^{(t+1)}-\boldsymbol{\theta}_k+\mathbf{u}_k^{(t)}\|_2^2\},\quad k=1,\ldots,p\\ \mathbf{u}_k^{(t+1)}=\mathbf{u}_k^{(t)}+\mathbf{H}\boldsymbol{\beta}_k^{(t+1)}-\boldsymbol{\theta}_k^{(t+1)},\quad k=1,\cdots,p\end{cases}$$

where t denotes the t-th iteration.

The above sub optimization problems have the following analytical results:

$$\begin{cases}\boldsymbol{\beta}_k^{(t+1)}:(\mathbf{I}_n+\gamma\mathbf{H}^T\mathbf{H})^{-1}[\mathbf{R}_k^{(t)}+\gamma\mathbf{H}^T(\boldsymbol{\theta}_k^{(t)}-\mathbf{u}_k^{(t)})]\\ \theta_k^{(t+1)}:\mathcal{S}(\mathbf{H}\boldsymbol{\beta}_k^{(t+1)}+\mathbf{u}_k^{(t)};\frac{\lambda}{L\gamma}),\\ \mathbf{u}_k^{(t+1)}=\mathbf{u}_k^{(t)}+\mathbf{H}\boldsymbol{\beta}_k^{t+1}-\boldsymbol{\theta}_k^{t+1},\end{cases}$$

for each $k=1,\ldots,p$, where $\mathcal{S}(z,\lambda)$ is the soft-thresholding operator, and

$$\mathbf{R}_k^{(t)}=\boldsymbol{\beta}_k^{(t)}+(1/L)\frac{1}{|D|}\nabla\tilde{\ell}_c(\boldsymbol{\beta}_k^{(t)})$$

It is noted that the above optimization steps are separable for the parameters associated with each k, and hence can be conveniently solved in a parallel fashion. In addition, under our choice of the spatial graphs, the graph Laplacian matrix $\mathbf{H}^T\mathbf{H}$ is a sparse matrix. As a result, the update of $\boldsymbol{\beta}^{(t)}$ only involves the linear solver of the sparse matrix $(\mathbf{I}_n+\gamma\mathbf{H}^T\mathbf{H})^{-1}$, whose sparse Cholesky factorization can be pre-computed efficiently using the `R` package `Matrix`. We iterate the above ADMM steps until convergence.

2.4 Simulation Studies

In this section, we conduct simulation studies to investigate the performance of the SVCI model. We design two different data generation scenarios:

- Scenario 1: Point patterns are generated from a Poisson point process on a planar window, where the log intensity is a linear function of an intercept and two covariates with clustered regression coefficients, i.e., $\rho(u; \boldsymbol{\beta}(u)) = \exp\{\beta_0(u) + z_1(u)\beta_1(u) + z_2(u)\beta_2(u)\}$.
- Scenario 2: Point patterns are generated from a Poisson point process on a linear network. We consider two sub-scenarios: (a) The log intensity is a piece-wise constant function, i.e., $\rho(u; \boldsymbol{\beta}(u)) = \exp\{\beta_0(u)\}$; (b) The log intensity is a linear function of an intercept and two covariates with clustered regression coefficients as in Scenario 1.

In Scenario 1, we focus on examining the performance of our method under different model choices, including the choice of graphs used in the graph fused lasso penalty and the choice between the Poisson likelihood based SVCI (SVCI-PL) and the logistic likelihood based SVCI (SVCI-LRL). For comparison studies, to the best of our knowledge, there are very limited existing methods available for spatially clustered coefficient log linear point process models on complex domains as reviewed in the Introduction, except for the simple case of an intercept-only log-linear model. As such, Scenario 2(a) is included so that we can compare SVCI with the nonparametric kernel density estimation method on a linear network (KDE.lpp) proposed in McSwiggan et al. (2017), the fast KDE method (KDEQuick.lpp) in Rakshit et al. (2019), and the resample-smoothed Voronoi intensity estimation method (Voronoi.lpp) in Moradi et al. (2019). For the case that has spatial covariates as in Scenario 2.(b), the comparison is made with the LGCP model (Møller et al., 1998), in which the inhomogeneity of the intensity function is modeled by a latent spatial Gaussian process random effects model.

Given the estimator $\hat{\boldsymbol{\beta}}$ defined in (2.9), we predict the coefficients at any given new location $u \in D \backslash \{u_1, \dots, u_M\}$ according to $\hat{\boldsymbol{\beta}}(u) = \sum_{i=1}^{M} \mathbf{1}_{\{u_i \in \mathcal{N}_K(u)\}} \hat{\boldsymbol{\beta}}(u_i)$/K, where $\mathcal{N}_K(u)$ denotes the K nearest neighbors of u. To quantify the performance of parameter estimation, we evaluate the estimation accuracy of $\beta_k(u)$ by the mean integrated squared error (MISE$_\beta$, Davis (1977)), defined as:

$$\text{MISE}_\beta = \frac{1}{p|D|}\sum_{k=1}^{p}\int_D (\beta_k(u) - \hat{\beta}_k(u))^2 du.$$

We implement our methods in `R` and provide the codes in https://github.com/LihaoYin/SVCI. The data generations are done using the `R` package `spatstat` (Baddeley and Turner, 2005). The competing KDE.lpp and KDEQuick.lpp methods are implemented using `density.lpp` and `densityQuick.lpp` in the `R` package `spatstat`, respectively. Voronoi.lpp is implemented using `densityVoronoi.lpp` also in the `R` package `spatstat`. The competing LGCP method is implemented in `R` using the `lgcp` function provided in `geostatsp` (Brown, 2015). In KDEQuick and KDE.lpp, bandwidth was selected by maximizing the approximated log-likelihood from a candidate set of bandwidths in a neighborhood of the optimal tuning parameter that minimizes MISE. In Voronoi.lpp, we set nrep $= 100$ and select the probability f by maximizing the approximated log-likelihood from a candidate set in a neighborhood of the optimal f that minimizes MISE.

All computations were performed on a Mac Pro with 2.4 GHz Intel Core i7 laptop with 8GB of memory.

2.4.1 Simulation Scenario 1

In Simulation Scenario 1, we consider a spatial 2D window $D = [0, R]^2 \subset \mathbb{R}^2$, where the true regression coefficients in the log-intensity function are assumed to have clustering patterns as shown in the top panel of Figure 2.2. We simulate the two covariates $\{z_1(u)\}$ and $\{z_2(u)\}$ from two independent realizations of a spatial GP with mean zero and an isotropic exponential covariance function taking the form of Cov$\{z_k(u), z_k(v)\} = \sigma^2 \exp(-\|u - v\|/\phi)$, $k = 1, 2$, $u, v \in [0, R]^2$, where the range parameter $\phi = 0.3R$ corresponding to a moderate spatial correlation setting, and $\sigma^2 = 1$.

Under one chosen fixed coefficient pattern, we experiment with a range of R values to simulate one realization from the Poisson point process model described in Scenario 1 such that the number of simulated points ranges from 800 to 6000 on average, in order to examine the performance of SVCI as the sample size increases with the expanding domain. Furthermore, we report the model performance under three different choices of the number of dummy points, denoted as $\texttt{nd}^2$: (a) $\texttt{nd}^2 < m$; (b) $\texttt{nd}^2 = m$; (c) $\texttt{nd}^2 > m$, where m is the number of the observed points. We also compare with an LGCP model with intensity function $\log \rho(u) = z(u)^T \boldsymbol{\beta} + \phi(u)$, where $\boldsymbol{\beta}$ are the constant-coefficients across the domain, and $\phi(u)$ is a spatial Gaussian process with a zero mean and a Mátern correlation function.

As discussed in Section 2.3, the selection of connection graphs for the fused lasso penalty plays critical roles on estimation accuracy and computation speed. In this study, we compare the performance of SVCI using three types of connection graphs, including the minimum spanning tree graph (MST), the Delaunay triangulation (DTs) and the K-nearest neighbor graph (K-NNs, K is set to be $3, 4, 5$). Also see a comparison study between K-NNs and r-NNs in Appendix Section A3. We run 100 repeated experiments of the SVCI model using each connection graph for both SVCI-PL and SVCI-LRL with a fix number of dummy points $\texttt{nd}^2 = m$.

In Table 2.1, we report the averaged MISE of the estimates $\hat{\boldsymbol{\beta}}$ (MISE$_\beta$). There are several noticeable observations. First, a denser graph such as the 5-NN graph produces a more accurate estimation result compared with that of a sparser graph such as the MST or 3-NN graph. The bottom panel of Figure 2.2 illustrates an example of the estimated coefficients using the 5-NN graph when $m = 2000$ and $\texttt{nd}^2 = m$, which demonstrates the capability of SVCI in capturing the cluster structure in the regression coefficients and detecting the abrupt changes across the boundaries of adjacent clusters. However, there is clearly a trade off between the estimation accuracy and computation efficiency when using different graphs; as reported in the left panel of Figure 2.3, the computation time (in seconds) using the 5-NN or Delaunay triangulation graph is roughly 1.5 times of the computation time using the MST. Second, the parameter estimation is more accurate when m grows larger, as evidenced by the decreasing value of MISE$_\beta$. Finally, SVCI-LRL produces comparable results with those from SVCI-PL when $m = 800$ or using the MST graph, but it notably outperforms SVCI-PL when $m \geq 1600$. This is consistent with the findings in

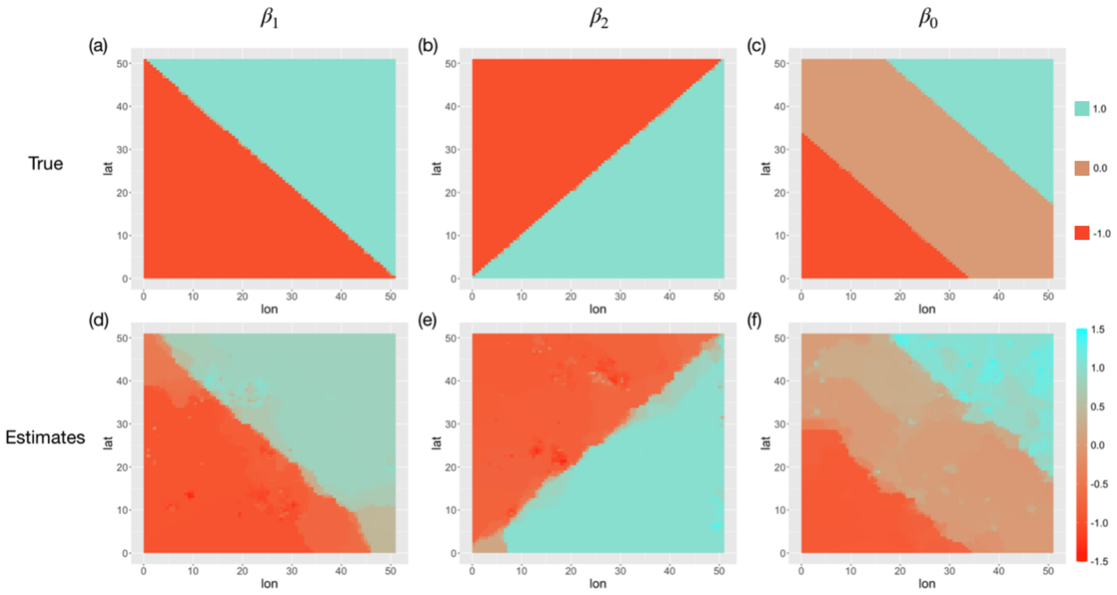

Figure 2.2: Upper panel: the true regression coefficients, $\beta_1(u)$, $\beta_2(u)$ and $\beta_0(u)$, in Scenario 1; Lower panel: the estimated coefficients from one simulation using the 5NN graph when $m = 2400$, $\texttt{nd}^2 = m$

Baddeley et al. (2014), which showed that for datasets with a large number of points or a highly structured point pattern, the logistic likelihood method produces a less biased estimator than its Poisson counterpart.

We further examine the performance of the SVCI model in terms of recovering the true intensity function. Given $\hat{\boldsymbol{\beta}}(u)$, we obtain the estimate of the log intensity function by $\log \hat{\rho}(u) = \mathbf{z}^T(u)\hat{\boldsymbol{\beta}}(u)$ for $u \in D$. The right panel of Figure 2.3 compares the MISE of $\log \hat{\rho}(u)$ from SVCI-PL, SVCI-LRL and LGCP, respectively. It is noted that SVCI-LRL maintains its superior performance when predicting the intensity function in comparison with SVCI-PL. Besides, both versions of SVCI produce more accurate estimates than LGCP when estimating the intensity function with clustered regression coefficients.

Next we examine the performance of SVCI in recovering the clusters of coefficients. Table 2.2 reports the Rand index for each of the SVCI estimates $\hat{\boldsymbol{\beta}}_1$, $\hat{\boldsymbol{\beta}}_2$ and $\hat{\boldsymbol{\beta}}_0$ averaged over 100 simulations, using the 5-NN graph and setting $\texttt{nd}^2 = m$. Rand index measures the proportion of pairs consisting of a true parameter and the corresponding estimated parameter that agree by virtue of belonging either to the same cluster or to different clusters. Overall, SVCI achieves an accurate cluster recovery result, evidenced by the relatively high Rand index value ranging from 0.73 to 0.93 in all settings. We also find that SVCI-LPL surpasses SVCI-PL in detecting spatial clusters. Finally, an interesting observation is that $\hat{\boldsymbol{\beta}}_0$ has a lower Rand index value than that of $\hat{\boldsymbol{\beta}}_1$ and $\hat{\boldsymbol{\beta}}_2$, which might be the consequence of having more clusters in the true function of $\beta_0(u)$.

Finally we check the sensitivity of the model performance to the number of dummy points. We fix $m = 1600$ and consider three different numbers of dummy points denoted by $\texttt{nd}^2$. Table 2.3 presents the averaged MISE$_\beta$ and the associated computation time over 100 simulations. For the Poisson likelihood, the default choice suggested in the `R` package `spatstat` is $\texttt{nd}^2 \approx 4m$. In our experiments, however, as presented in Table 2.3, both SVCI-PL and SVCI-LRL achieve the minimal MISE$_\beta$ when $\texttt{nd}^2 = 60^2$, i.e. when the number of dummy points roughly equals the number of points. Moreover, based on the results in Table 2.3, we observe that when $\texttt{nd}^2$ is not too large, both SVCI-PL and SVCI-LRL seem to achieve a smaller MISE$_\beta$ but at a higher computation cost as $\texttt{nd}^2$ increases. Weighing the trade-off between computation efficiency and estimation accuracy, we recommend to use $\texttt{nd}^2 \approx m$ in practice.

Table 2.1: Scenario 1: mean integrated squared error of $\boldsymbol{\beta}$ (MISE$_\beta$) averaged over 100 simulations for different values of m with $\texttt{nd}^2 = m$, different connection graphs, and the Poisson-based SVCI-PL method and the logistic regression based SVCI-LRL method.

Method	$m = 800$		$m = 1600$		$m = 2400$		$m = 3600$		$m = 6000$	
	PL	LRL	PL	LRL	PL	LRL	PL	LRL	PL	LRL
	MISE$_\beta$									
MST	0.234	0.243	0.222	0.224	0.189	0.191	0.152	0.146	0.130	0.125
3-NN	0.223	0.230	0.204	0.182	0.189	0.184	0.155	0.142	0.127	0.115
4-NN	0.214	0.209	0.200	0.181	0.157	0.132	0.133	0.116	0.115	0.098
5-NN	0.209	0.195	0.184	0.153	0.141	0.119	0.122	0.105	0.095	0.078
DT	0.215	0.211	0.174	0.141	0.128	0.111	0.113	0.097	0.080	0.072

Table 2.2: Scenario 1: Rand index of the estimates $\hat{\boldsymbol{\beta}}_1$, $\hat{\boldsymbol{\beta}}_2$ and $\hat{\boldsymbol{\beta}}_0$ (averaged over 100 Monte Carlo simulations) of SVCI-PL and SVCI-LRL for different values of m, using $\texttt{nd}^2 = m$ and 5-NN connection graphs.

	$m = 800$		$m = 1600$		$m = 2400$		$m = 3600$		$m = 6000$	
	PL	LRL	PL	LRL	PL	LRL	PL	LRL	PL	LRL
	Rand Index									
$\hat{\boldsymbol{\beta}}_1$	0.817	0.839	0.857	0.874	0.883	0.903	0.897	0.915	0.917	0.930
$\hat{\boldsymbol{\beta}}_2$	0.817	0.836	0.851	0.867	0.879	0.906	0.893	0.917	0.914	0.925
$\hat{\boldsymbol{\beta}}_0$	0.729	0.760	0.742	0.773	0.767	0.786	0.787	0.806	0.803	0.825

Table 2.3: Scenario 1: comparing the MISE_β (averaged over 100 simulations) and computation time (in seconds) between SVCI-PL and SVCI-LRL for different numbers of dummy points, using $m = 1600$ and 5-NN connection graphs.

dummy points	SVCI-PL		SVCI-LRL	
	MISE_β	time(s)	MISE_β	time(s)
$\texttt{nd}^2 = 30^2$	0.195	1.54	0.164	1.61
$\texttt{nd}^2 = 40^2$	0.184	1.86	0.153	1.81
$\texttt{nd}^2 = 60^2$	0.182	2.38	0.151	2.30
$\texttt{nd}^2 = 80^2$	0.190	3.35	0.167	3.24

2.4.2 Simulation Scenario 2

In Scenario 2, we generate spatial points on the `chicago` linear network from the `R` package `spatstat`. The network shown in the left panel of Figure 2.4 depicts the road network in an area of Chicago, USA near the University of Chicago (Baddeley and Turner, 2005). We bound the linear network in a window $D = R \times [0, 1]^2$ and increase R to expand D, so that the linear network that resides in D grows with D at the same rate to obtain an increasing number of realizations on the network.

We first consider a simplified case where there is no covariate available. We focus on the estimation of intensity function whose true value is a piece-wise constant function, that is, the intensity function $\rho(u) = \exp\{\beta_0(u)\}$, and the true value of $\beta_0(u)$ has a clustered pattern as shown in the left panel of Figure 2.4. The original graph is constructed following the method described

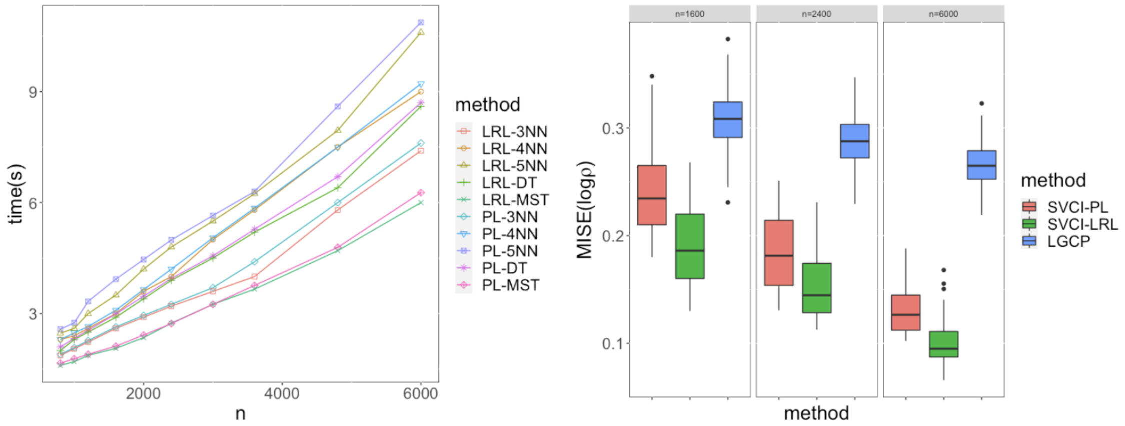

Figure 2.3: Left: computation time to solve the optimization for one tuning parameter using different connection graphs; Right: the boxplots of MISE$_\beta$ for SVCI-PL, SVCI-LRL with 5-NN graphs and LGCP. Reported results are averaged over 100 Monte Carlo simulations under Scenario 1.

in the last paragraph of Section 3.1. The upper part of Table 2.4 presents the MISE of $\log\rho$, i.e., the log intensity function for each value of m. In general, we obtain similar findings on the linear network as on the planar window presented in Scenario 1; MISE from both the Poisson based and logistic based SVCI models show a convergence tendency as the domain expands and m goes up, and the logistic likelihood based method achieves a slightly more accurate estimation than the Poisson based method with a large number of points. It is clear from Table 2.4 that both SVCI-PL and SVCI-LRL outperform the KDE based and resample-smoothed Voronoi based intensity estimation methods in almost all settings. Previous studies (Barr and Schoenberg, 2010) show that KDE estimators may suffer from the problem of having substantial bias and high variance when there are abrupt changes in the intensity. Both SVCI and Voronoi.lpp are designed to alleviate this problem, as evidenced by their improved performance over the two KDE methods in Table 4. Nevertheless, SVCI seems to be more effective than Voronoi.lpp to capture abrupt changes or clustering patterns. We illustrate an example in the right panel of Figure 2.4, which plots the true and the estimated log intensity along a selected road segment from one simulation. It clearly shows that SVCI captures the intensity with discontinuities more efficiently than KDE.lpp.

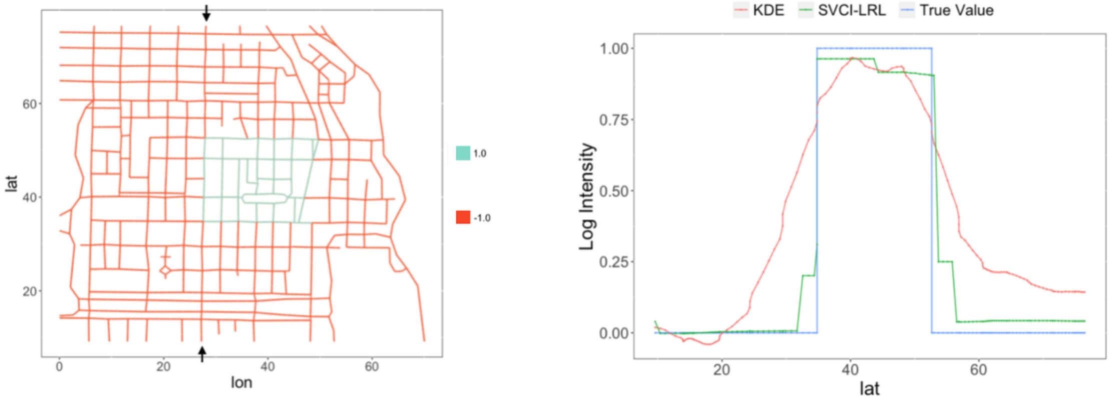

Figure 2.4: Left: the true log intensity $\beta_0(u)$ on the `chicago` network in Scenario 2(a); Right: the true and the estimated log intensity functions along one road segment corresponding to the line between the two black arrows in the left panel of Figure 2.4.

We also compare the computation time of each method and report the detailed results in Appendix Table A2 for various values of m. Taking $m = 2400$ as an example, to get one estimate, KDE.lpp requires 4.95 seconds, KDEQuick.lpp requires 0.084 seconds, and Voronoi.lpp requires 4.95 seconds. In contrast, SVCI-LRL needs 0.93 seconds to construct the connection graph and 1.11 seconds to get an estimate. Although SVCI is not the fastest among the compared methods, overall, its computation is still reasonable and competitive, especially considering its superior performance in intensity estimations.

We then consider the case with an intercept and two covariates as described in Scenario 2(b). The true regression coefficients are plotted in the subfigures (a-c) of Figure 2.5. The subfigures (d-f) of Figure 2.5 give the estimated coefficients from SVCI on the `chicago` network. The results demonstrate that our method is capable of capturing clustered coefficient patterns on a linear network. In addition, the log intensity estimation results presented in the lower part of Table 2.4 are in general consistent with the findings presented in Scenario 1 and Scenario 2(a); the performance of the SVCI model with the logistic regression likelihood or with a larger m is more preferable. Table 2.4 displays the results from LGCP as a comparison, which indicate a clear improvement in estimation accuracy when using SVCI over LGCP.

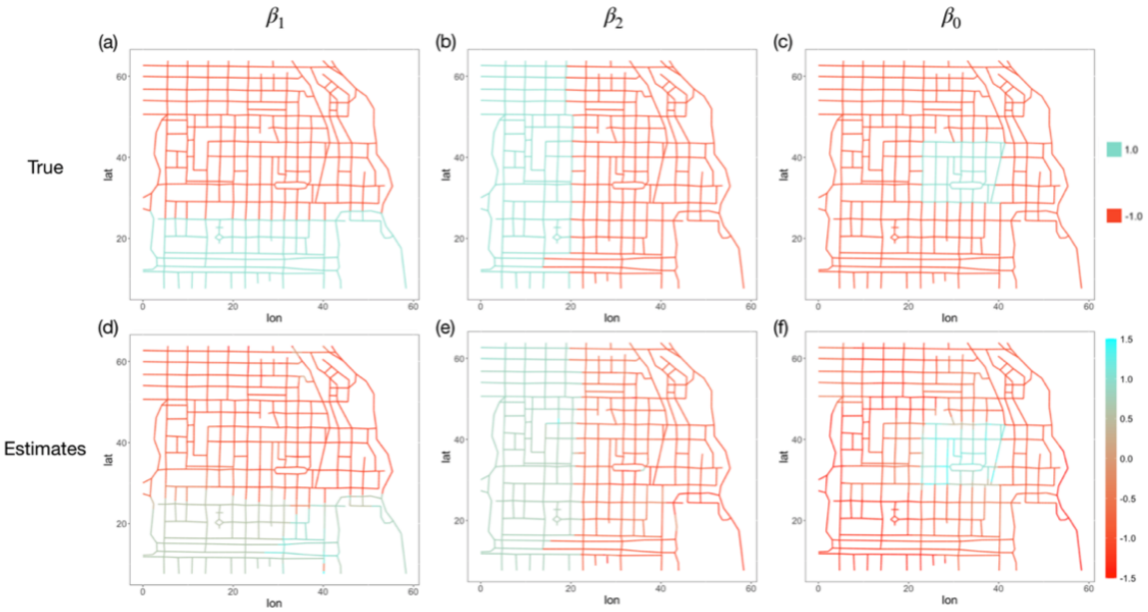

Figure 2.5: Upper panel (a-c) : the spatial structures of true coefficients $\beta_1(u)$, $\beta_2(u)$ and $\beta_0(u)$ in Scenario 2(b); Lower panel (d-f): the estimated coefficient surfaces from in one simulation using SVCI-LRL with $m = 2400$.

Table 2.4: Scenario 2: mean integrated squared error of log intensity ($\text{MISE}_{\log\rho}$) averaged over 100 simulations for different values of m. We compare the two estimating equations, the Poisson likelihood (PL) and the logistic regression likelihood (LRL), with their competitors.

Method	$\text{MISE}_{\log\rho}$				
	$m = 800$	$m = 1600$	$m = 2400$	$m = 3600$	$m = 6000$
(a): $\rho(u) = \exp\{\beta_0(u)\}$					
SVCI-PL	0.128	0.101	0.084	0.057	0.041
SVCI-LRL	0.117	0.095	0.074	0.042	0.030
KDE.lpp	0.157	0.140	0.112	0.084	0.061
KDEQuick.lpp	0.133	0.127	0.109	0.075	0.054
Voronoi.lpp	0.128	0.120	0.094	0.067	0.048
(b): $\rho(u) = \exp\{z_1(u)\beta_1(u) + z_2(u)\beta_2(u) + \beta_0(u)\}$					
SVCI-PL	0.177	0.152	0.135	0.114	0.085
SVCI-LRL	0.165	0.141	0.122	0.097	0.072
LGCP	0.227	0.202	0.188	0.154	0.137

2.5 Real Data Analysis

We consider two real data examples to illustrate the performance of the proposed method. The first Toronto Homicide data example has a moderate data size with 1398 points and three explanatory variables on a domain with irregular boundaries. And the second Western Australia Traffic Accidents data has a larger data size with $14,562$ points on a linear road network. In both studies, we use SVCI-LRL and $\texttt{nd}^2 \approx m$, due to their favorable performance in our simulation studies.

2.5.1 Toronto Homicide Dataset

We apply the proposed SVCI model to the analysis of the Toronto Homicide dataset. The raw dataset contains the information of 1398 homicides occurred in Toronto, Canada during 1990 to 2014, recording the locations of murder scenes, homocide types and information of victims obtained from the Toronto Star Newspaper (`http://www.thestar.com/news/crime/torontohomicidemap.html`). The data can be accessed from the `R` package `geostatsp` (Brown et al., 2015). We select the more recent data since 2000 and delete the data which have duplicated locations. There remain 764 homicide cases for the final analysis. Figure 1.1 shows the entire Toronto city and the locations of the selected cases within a 42×31 km rectangle window. Notably, the old Toronto region in the middle of the coast has more frequent occurrences of homicides.

The data also contains the records of average income, night lights and population density of Toronto city in 2006, and we use them as explanatory variables. Figures 2.6 (a-c) show the observations of the three variables. As can be seen, there is a large spatial variation of these variables across the city. We scale and center each spatial covariates before running our point process models.

We focus on the investigation of the relationship between the distribution of homicides and the three explanatory variables. We first fit a log Gaussian cox process model with constant regression

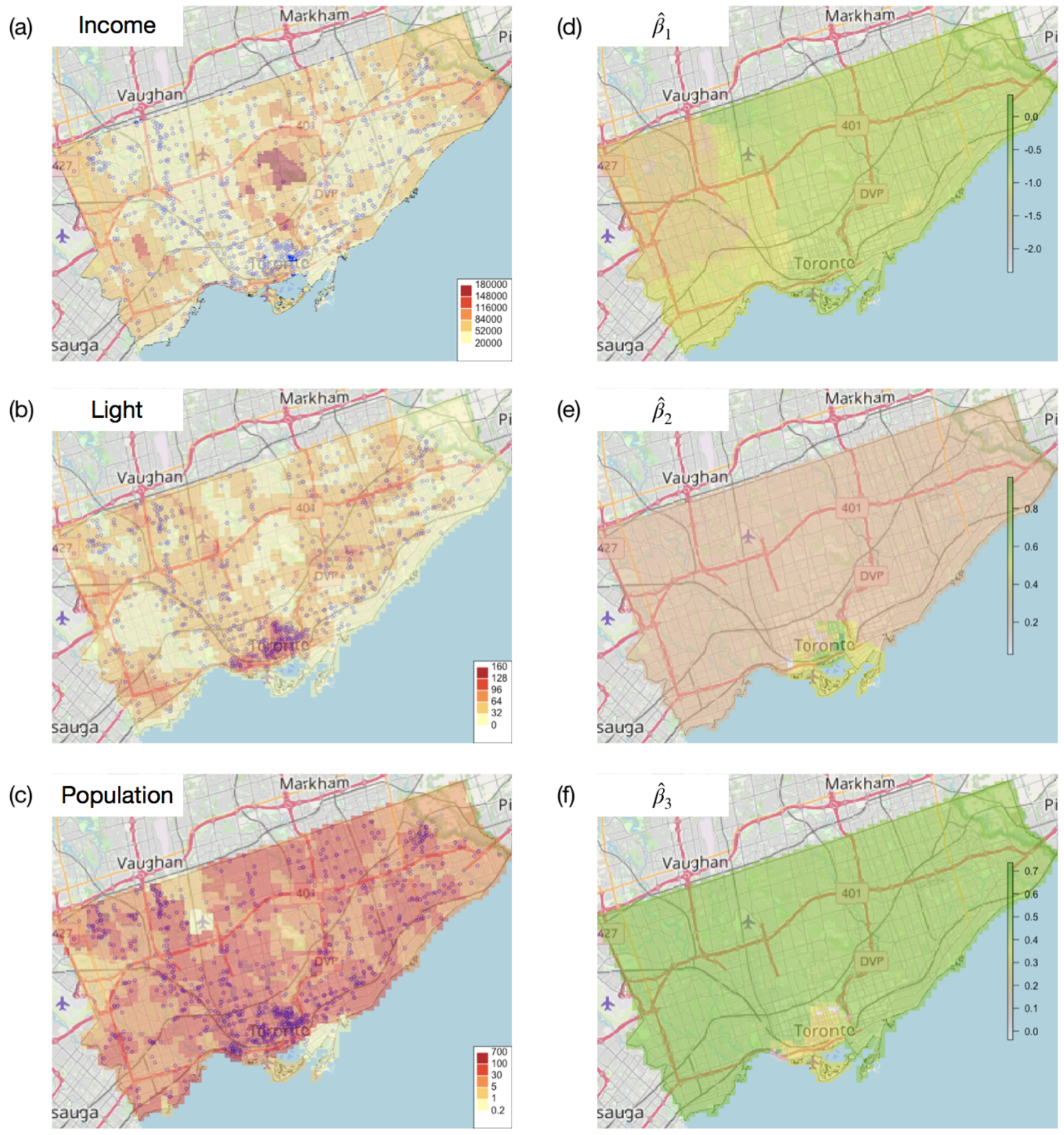

Figure 2.6: Left Panel: patterns of covariates; Right Panel: patterns of estimates; (a,d) Average income of the residents in Toronto; (b,e) Light intensity in Toronto night; (c,f) Population density in Toronto;

coefficients as a benchmark for comparisons, whose intensity function takes the form

$$\log\{\rho(u)\} = \beta_0 + \text{Income}(u)\beta_1 + \text{Night}(u)\beta_2 + \text{Pdens}(u)\beta_3 + \phi(u)$$

where β_1, β_2 and β_3 are the constant regression coefficients and $\phi(u)$ is a spatial Gaussian process with a zero mean and a Matern correlation function. The estimated parameter estimates from LGCP are $\beta_1 = -0.918$, $\beta_2 = 0.378$, $\beta_3 = 0.207$ and $\beta_0 = 1.096$, respectively. These estimates imply that the homicides are more likely to occur in the area with a lower average income, a better lights condition and a denser residential population.

We then fit the SVCI model with a 5-NN graph assuming that homicide locations follow a spatial point process with the following intensity function,

$$\log\{\rho(u)\} = \beta_0(u) + \text{Income}(u)\beta_1(u) + \text{Night}(u)\beta_2(u) + \text{Pdens}(u)\beta_3(u).$$

Here $\beta_k(u)$, $k = 0, 1, 2, 3$ are spatially varying coefficients, whose estimates are shown in Figure 2.6(d-f). It takes about 0.072 seconds to construct the 5-NN graph and 0.69 seconds to get an estimate of $\boldsymbol{\beta}$ for each tuning parameter. Clearly, the results of SVCI reveal more details about the effects of covariates than those from LGCP. 3, 4 and 3 major clusters are detected for β_1, β_2 and β_3 respectively. Overall, the signs of $\beta_k(u)$ are the same as the results of LGCP. For example, the estimates of $\beta_1(u)$ range from -2 to 0, indicating a negative relationship between income and homicide occurrence as is expected. Such a negative relationship is most prominent in the western region of Toronto whereas a weaker relationship is observed near the upper east corner of Toronto City. For both $\beta_2(u)$ and $\beta_3(u)$, we observe a small cluster at the Old Toronto region, which has the most concentrated homicide cases. It is notable that the relationships between light intensity/population density and homicides occurrence in the Old Toronto region differ significantly from the rest of Toronto city; a stronger positive relationship is observed for both variables.

2.5.2 Western Australia Traffic Accidents

In this section, we study the traffic accidents data in the state of Western Australia for the year of 2011, as shown in Figure 1.1. The data were originally provided by the Western Australian State Government Department of Main Roads and are made publicly available as part of the Western Australian Whole of Government Open Data Policy. The data can also be accessed from the `R` package `spatstat.Knet`. It consists of $14,562$ locations of accidents on a road network with $115,169$ road segments constrained in a $[217.4, 1679.1] \times [6114.9, 7320.6]$ km rectangle window.

The grey lines in Figure 1.1 represent the traffic network of Western Australia, and each red point marks an accident spot. It is clear from this Figure that accidents are highly concentrated around the Perth metropolitan area located in the western coastal region. This region contains nearly 75% of the population in Western Australia. By contrast, the remote eastern region has a much sparser road network and a smaller number of traffic accidents. Our goal is to estimate the intensity function over this network to investigate the spatial variation of accident occurrences.

We build the SVCI model of the intensity function with a spatially varying intercept, $\rho(u;\beta) = \exp\{\beta_0(u)\}$. In this study, we don't have any spatial covariates available and hence we focus on detecting the clustered patterns of the intensity function $\rho(u)$. SVCI takes about 1.75 minutes to construct a connection graph using the graph construction method in Section 3.1 and takes on average 5.01 seconds to get an estimate of $\boldsymbol{\beta}$ for each tuning parameter. 23 clusters are detected, and Figure 2.7 plots the estimated log intensity $\log \hat{\rho}(u)$ on the road network. In the western part of the city near downtown, there are many small clusters, indicating more local variations in traffic accidents intensities in these regions. In contrast, the eastern and northern part of the city has a fewer number of clusters. We also notice that $\hat{\rho}(u)$ has a large spatial variation, ranging from 0 per kilometer in some remote eastern areas to nearly 50 accidents per kilometer in some busy roads in the Perth metropolitan area.

We zoom into the sub-region of $[372, 431] \times [6434, 6501]$ km displayed in Figure 2.8 (a) to have a detailed investigation of the traffic accident rates in the densely populated Perth metropolitan area. It is clear that several roads are having substantially higher intensities than the rest of the

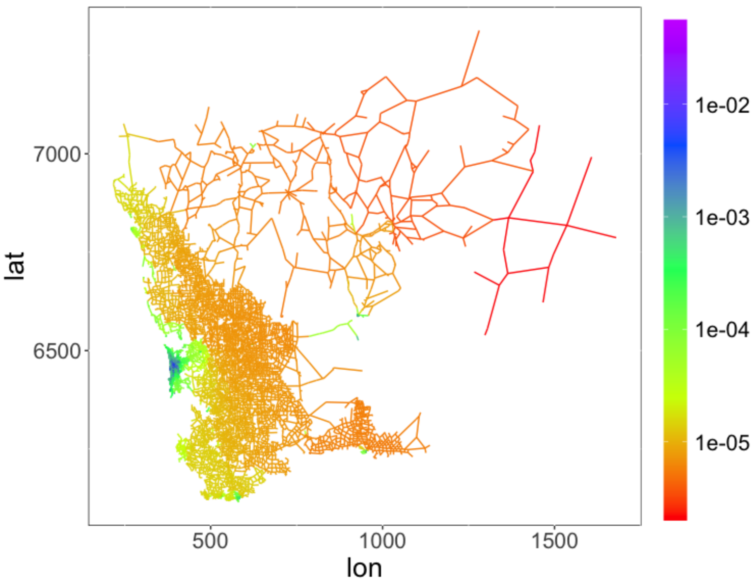

Figure 2.7: Intensity estimates for the accidents on the Western Australian road network.

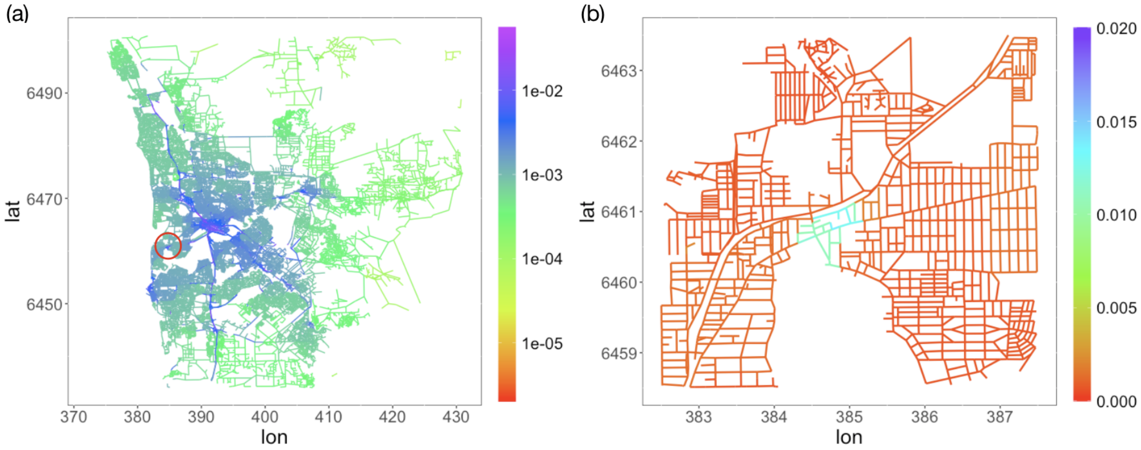

Figure 2.8: (a): Intensity estimates map for the accidents in the metropolitan Perth Area; (b): Intensity estimates map by zooming into the red circle area in the left panel.

roads, many of which are along the major freeways of the city. In particular, we observe very high intensity values at or near the center of the city marked by the purple color. Indeed, these roads and intersection are located at the Perth Central Business District. In contrast, although having dense local road networks, many residential areas away from highways have relatively lower intensity values. One advantage of SVCI lies in its capability of capturing intensity functions with abrupt changes. To give an example, we highlight a road segment on Highway 5 in Perth by a red circle in Figure 2.8 (a), and show the zoomed map in Figure 2.8 (b). It is noticeable that a sudden jump in the estimated intensity function appears near the southwest end of the road. After verifying with the Google satellite image, we confirm that the northeast part of the road passes through a large residential area, whereas the southwest part is a commercial and public service area (restaurants/shops/school/church/hospital) that is expected to have a higher rate of accidents.

2.6 Conclusions

In this study, we propose a varying coefficient log-linear intensity model, referred to as the SVCI model, for the visualization and analysis of spatial point processes. We utilize a graph fused lasso regularization to estimate the clustering patterns of the regression coefficients. The method guarantees spatially contiguous clustering configurations with highly flexible cluster shapes and data-driven cluster sizes. It supplements the current research on intensity estimation, which primarily focuses on relatively smooth intensity functions without covariates or spatially constant regression coefficients. The method also has the advantage of being applicable to a broad range of complex domains such as line networks and spatial domains with irregular boundaries. The computation of the model is made highly efficient by using a proximal gradient optimization algorithm. Numerical studies show that our method produces more accurate intensity estimations than several competing methods such as the KDE-based methods (McSwiggan et al., 2017; Rakshit et al., 2019) and the resample-smoothing Voronoi intensity estimation method (Moradi et al., 2019), when intensity functions exhibit discontinuous changes on linear networks. The computation of SVCI is in general reasonable compared to its competitors considered in this chapter. The method is applied to identify spatially heterogeneous patterns in the determinants of Toronto crime events and the intensity of traffic accidents in Western Australia.

Moving forward, this work could be further refined in several ways. First, SVCI only considers a small fixed number of covariates. However, in practice, practitioners may face a large number of available covariates but lack a strong theory to inform variable selection. There is a need of a more general model that allows researchers to undergo variable selection and spatial cluster identification simultaneously for point processes. Second, the SVCI estimator does not come with an uncertainty measure that makes it hard for statistical inference, a common issue shared by many regularization based approaches. We may consider a Bayesian version of the method or a bootstrapping based approach to address the inference problem. Third, an interesting research direction is to extend the finite dimensional graph regularization based method to an infinite dimensional process-based clustered coefficient model such that spatial predictions can be done in a more rigorous way. Finally, the method can be extended by considering a weighted composite log-likelihood to further improve statistical efficiency for non-Poisson point processes (Guan and Shen, 2010).

3. RISK BASED ARSENIC RATIONAL SAMPLING DESIGN FOR PUBLIC AND ENVIRONMENTAL HEALTH MANAGEMENT

3.1 Introduction

Arsenic (As) is ranked as the 20th most abundant element in the Earth's crust and has been studied internationally. Groundwater contaminated with arsenic has been recognized as a global threat, negatively impacting human health (Podgorski and Berg, 2020; DeSimone and Hamilton, 2009). The primary human exposure to arsenic is drinking water with additional contributors such as food and air (Almberg et al., 2017; Vahter, 2009; Sohel et al., 2009). Arsenic is a potent human carcinogen, which can cause bladder, lung, and skin cancers (Argos et al., 2012). Furthermore, arsenic and its metabolites can cross the placental barrier and create risk for adverse maternal and fetal health, leading to adverse birth outcomes (Bloom et al., 2014). The Environmental Protection Agency (EPA) federal drinking water standard established 0.01 mg/L as the arsenic maximum contaminant levels (MCLs) in drinking water. In the USA, approximately 41.8 million (13% of the total US population) people obtain drinking water from private wells, and the private wells are not regulated under the current EPA regulation (Association, 2020). The recent national Water-Quality Assessment Program from the United States Geological Survey (USGS) reports that more than one out of five wells contain contaminants at concentrations exceeding the EPA MCLs or USGS health-based screening levels. Among the various pollutants that exceed the EPA maximum contaminant levels, arsenic contamination is a common finding. Because private wells are not regulated in the US, in the Midwest region, a significant percentage of the population depending on private wells for drinking water is at risk due to drinking water arsenic contamination (Schnoebelen et al., 2017). Arsenic testing in private well water represents a fundamental mean that helps mitigate the arsenic risk in the rural population for public and environmental health. In reality, many of the private wells are not tested, which presents a significant challenge for health risk mitigation. From the management perspective, a scientifically sound sampling plan to test a representative sample size is needed to characterize the environmental arsenic hazard with limited resources.

Sampling theory can be used to guide a large number of chemical and biological analyses for environmental control and consumer safety (Minkkinen, 2004). As for arsenic testing, a systematic sampling plan is critical for risk assessment to draw science and data-based conclusions and make the best usage of limited resources. The EPA has published guidance for data quality objectives with regard to sampling design (USEPA). One of the key preparations for a sampling design is to determine the sample size and sampling error for representative sample collection.

Understanding sample statistical distributions is critical when selecting a sampling method, sampling strategy, and sample size. Application of probability distribution can help develop a science-based sampling plan and estimate the chemical and biological hazards in the environment.

Previously, binomial probability theory has been well studied for sample size determination for estimating a binomial proportion (Gonçalves et al., 2012). Application examples include the sampling plan in product inspection and surveillance (Lee et al., 2016), epidemiology (Sepúlveda and Drakeley, 2015), and medical diagnostics (Joseph and Reinhold, 2005). In many of these applications, a univariate binomial distribution is considered, that is, the underlying binomial proportion parameter is assumed to be a constant in the study. However, due to the spatial heterogeneity nature of arsenic distribution in the earth's crust and groundwater, the traditional binomial sampling scheme based on a univariate binomial distribution may not be suitable to survey the target private well population. There is a great need to develop new sampling schemes capable of accounting for the spatially heterogeneity nature of the arsenic distribution.

In terms of arsenic contamination, quite a few statistical and mathematical models have been used to estimate and predict arsenic concentrations in groundwater and private wells. Logistic models for binomial distributions are widely adopted to estimate the spatial distribution of As contamination probability at both global and regional levels (Amini et al., 2008; Ayotte et al., 2006; Winkel et al., 2008; Podgorski et al., 2017; Winkel et al., 2011; Rodríguez-Lado et al., 2013; Yang et al., 2012). For instance, a logistic linear regression model has been used to predict the high arsenic domestic well population in the US (Ayotte et al., 2017). Furthermore, boosted regression tree models (weak-learner ensemble models) and traditional logistic linear models have been compared to estimate and predict arsenic contamination probabilities in drinking water wells in the Central Valley, California (Ayotte et al., 2016). Similar to those statistical models, predictive variables are used to predict geogenic arsenic in drinking water wells in glacial aquifers, north-central USA (Erickson et al., 2018). Machine learning models have also been used to predict arsenic concentrations in groundwater in Asia (Tan et al., 2020). Nevertheless, the aforementioned models primarily focus on the estimation and prediction of arsenic distributions rather than the sampling design. Moreover, most methods often rely on a rich set of predictors and training data set to guarantee model accuracy. To the best of our knowledge, there is very limited work that combines the model-based estimation of varying arsenic distributions with the binomial sampling design method.

To close this gap in the current literature for spatial binomial distribution sampling design, the current study proposes a spatially adaptive sampling design approach, by estimating a spatially clustered underlying contamination distribution. We apply this method to determine the data locations to understand arsenic contamination in private wells in Iowa. The method is different from traditional spatial sampling design methods (Zhu and Stein, 2006; Diggle et al., 2010) that often assume continuous process-based spatial models for relatively smooth spatial fields. In contrast, we model the underlying contamination risk as a spatially clustered function for a straightforward interpretation of the result. It also has the advantage of detecting discontinuous spatial heterogeneity in the arsenic distribution and then borrowing information within each identified spatially homogeneous cluster for an adaptive sampling design. The method is built upon a graph fused lasso regularization method (Tibshirani et al., 2005), which automatically detects clusters of spatial units and estimates the underlying spatially varying contamination distributions simultaneously. Thanks to the flexibility of graphs, our spatial clustering model enjoys several nice properties. First, it leads to very flexible cluster shapes naturally satisfying spatial contiguity constraints. Second, the method automatically learns the number of clusters from the data, relaxing the limitation in other clustering algorithms that require to specify the number of clusters a priori.

Another unique advantage of estimating a spatially clustered contamination distribution over other contamination distribution estimation methods lies in its easy integration with the traditional binomial sampling theory. Within each identified spatial cluster, the contamination distribution can be treated as having a common binomial proportion, for which we propose and compare two different sample size determination methods at different levels of acceptance precision and confidence. Given the sample size calculations, a remaining sample design task is to determine the sampling locations. In our study, both the candidate wells and the available tested wells are distributed highly unevenly in the study region. To ensure the sampling design has a balanced spatial coverage, we propose a practical algorithm based on spatial point processes to distinguish areas that have been sufficiently-sampled and insufficiently-sampled, and determine new sampling locations accordingly. This new strategy, presumably more adaptive than traditional sampling without considering heterogeneity in sampling distributions, can potentially provide more precise tools to efficiently allocate sample collection efforts and resources.

3.2 Materials and Methods

3.2.1 Sample Collection and Analysis

For the private well samples, the data used to build the model was collected as part of the Iowa Grants-to-Counties (GTC) program. The Iowa GTC program was established in 1987 after the Iowa legislature passed the Iowa Groundwater Protection Act to protect groundwater. Arsenic testing has been included as part of the GTC program based on Iowa Administrative Code (of Public Health, 2016). A total of $14,570$ samples were collected and analyzed at the University of Iowa State Hygienic Laboratory from July 1st, 2015 to June 16, 2020. As part of the GTC program, the local health department collects the private well samples by conducting a home visit, and sending them to a laboratory for analysis. It should be noted that the selection of the laboratory is at the county's discretion.

For all the samples analyzed at the State Hygienic Laboratory, the water sample is collected either at the tap faucet or outside the house. Samples are collected in a 4 oz. HDPE plastic bottle containing 1 mL of $1 + 1$ nitric acid as a preservative. Cooling is not required for sampling. Samples are screened for turbidity following Standard Methods 2130 B using a HACH model 2100N Turbidimeter. Samples exceeding 1 nephelometric turbidity units (NTU) are digested prior to analysis. The arsenic analysis is performed based on the Iowa State Hygienic Laboratory standard operating procedure (SOP), similar to the EPA 200.2 method. Briefly, a 50-mL aliquot is transferred from a well-mixed sample to a polypropylene digestion tube (Environmental Express #UC475-GN). One mL of 1+1 nitric acid and 0.5 mL of $1 + 1$ hydrochloric acid (Fisher, Trace Metal Grade) are added to the tubes. Digestion is accomplished using a hot block (Environmental Express #SC154) at approximately $85°$C. The sample volume is reduced to 10 mL, and then the sample is covered with a watch glass (Environmental Express #SC505), and refluxed for 30 minutes. The tubes are cooled and diluted to 25 mL with reagent water. The samples are further diluted to 50 mL using a mixture of 2% nitric acid and 1% hydrochloric acid. The samples are then analyzed for arsenic using an Agilent 7500 CE inductively coupled plasma mass spectrometer following EPA method 200.8. Approximately 5 mL of sample is transferred to a polypropylene autosampler tube for analysis. The instrument is calibrated using a multi-point calibration curve (0, 1, 5, 50, 100, 500 ug/L). Standards are matrix-matched to the sample. Thus, digested samples are not analyzed in the same run with direct analysis samples. Internal standards are introduced via a mixing tee at the instrument. Yttrium is used as the internal standard for arsenic. Results are not reported unless all quality controls pass their acceptance limits per the method.

The raw data amount to $14,570$ previously collected observations of Arsenic tests in total (Figure 1.2). Based on the risk categories, we characterize the wells that contain higher than 0.01 mg/L arsenic as high risk wells, and use a binary variable to denote whether a well is at high risk. We exclude the observations whose location information is absent. We also aggregate the repeated measurements at the same locations into one single observation, by setting the binary value to be 1 if there is at least one concentration measurement exceeding MCL. A visual presentation of the private well arsenic testing is available through the Iowa Department of Public Health website[1].

[1] https://tracking.idph.iowa.gov/Environment/Private-Well-Water

3.2.2 Estimation of spatially clustered contamination probabilities

Let $y(\mathrm{s}_i)$ denote the binary variable at a well location s_i, for $i = 1, \ldots, n$, coded as being 1 if the arsenic concentration at s_i is exceeding the EPA MCL (i.e., 0.01 mg/L), and 0 otherwise. Here, n is the total number of available tested wells. We propose a spatially varying binary logistic model for $y(\mathrm{s})$. Specifically, we assume

$$P\big(y(\mathrm{s}_i) = 1\big) \sim \text{Bernoulli}\big(p(\mathrm{s}_i)\big), \quad \text{for } i = 1, \ldots, n, \tag{3.1}$$

where $p(\mathrm{s}_i)$ is the probability of the well located at s_i being contaminated. In the logistic regression model, we model the probability $p(\mathrm{s}_i)$ as

$$p(\mathrm{s}_i) = \frac{1}{1 + \exp\{-\beta(\mathrm{s}_i)\}}$$

or equivalently, $\log \frac{p(\mathrm{s}_i)}{1-p(\mathrm{s}_i)} = \beta(\mathrm{s}_i)$, where $\beta(\mathrm{s}_i)$ is interpreted as the log-odds of the arsenic contamination event that $y(\mathrm{s}_i) = 1$. Let $\boldsymbol{\beta} = \big(\beta(\mathrm{s}_1), \ldots, \beta(\mathrm{s}_n)\big)$ be the stacked regression parameters for all the observed well locations. It follows that the corresponding logistic regression log likelihood function takes the form:

$$\ell(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \log(1 + e^{\beta(\mathrm{s}_i)}) + \sum_{i=1}^{n} y(\mathrm{s}_i)\beta(\mathrm{s}_i) \tag{3.2}$$

It is noted from (3.1) that we relax the assumption of having a constant contamination probability p, or equivalently, contamination log-odds, β, over the whole study region, and instead assume it is varying over space. This assumption is reasonable for a large study region like Iowa due to the anticipated spatial heterogeneity in the arsenic concentration in groundwater and private wells. Specifically, we assume $p(\mathrm{s})$ is a spatially clustered function, that is, there exists a number of geographical clusters such that $p(\mathrm{s})$ stays relatively homogeneous within each cluster but varies across clusters. This will facilitate the easy visualization and interpretation of the varying contamination probability across different identified clusters. We will show in Section 3.2.3 that the spatially clustered contamination probability estimation also leads to an efficient spatially adaptive sampling design strategy.

We consider a flexible regularization model for pursing the clustered pattern of $\beta(\mathbf{s})$ and $p(\mathbf{s})$. Regularization methods have gained large popularity in modern high dimensional statistics and machine learning methods for various statistical learning tasks (Bühlmann and Van De Geer, 2011). They have proved to be effective in imposing structural assumptions on model parameters such as sparsity, smoothness, and clustering to avoid over-fitting problems. The regularization method for the Arsenic contamination model is performed in the following steps:

1. Construct a spatial graph, denoted as $G = (V, E)$ where $V = \{v_1, v_2, ..., v_n\}$ is the vertex set with n vertices and E is the edge set. For a spatial problem, each vertex represents a spatial location. For example, in the arsenic case study, each vertex v_i represents a well location $\mathbf{s}_i$, and the edge set E reflects the prior assumption on the neighborhood structure of well locations based on spatial proximity. The edge set selection is an important component of the method, which we will discuss later in this section.

2. Use the graph from step 1 to construct a homogeneity pursuit regularization, also called the fused lasso penalty function (Tibshirani et al., 2005, 2011) , for $\boldsymbol{\beta}$ as follows:

$$\rho \sum_{(i,j)\in E} |\beta(\mathbf{s}_i) - \beta(\mathbf{s}_j))|. \tag{3.3}$$

3. Combine the penalty function in (3.3) with the logistic log-likelihood function in (3.2) to form a penalized objective function, which we minimize to obtain an estimator of $\boldsymbol{\beta}$ as follows:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}) = \arg\min_{\boldsymbol{\beta}}\{-\frac{1}{n}\ell(\boldsymbol{\beta}) + \rho \sum_{(i,j)\in E} |\beta(\mathbf{s}_i) - \beta(\mathbf{s}_j)|\}. \tag{3.4}$$

4. After obtaining $\hat{\boldsymbol{\beta}}$, calculate the estimate of the contamination probability from $\hat{p}(\mathbf{s}_i) =$

$$\frac{1}{1+\exp(-\hat{\beta}(\mathbf{s}_i))}.$$

The fused lasso regularization in step 2 is used to impose the assumption that the arsenic contamination probabilities at two wells are more likely to take the same value if they are connected by an edge in E of the specified spatial graph. The objective function $Q(\boldsymbol{\beta})$ in (3.4) takes a similar form as the standard negative log-likelihood function from Bernoulli distributions for binary arsenic data, but with an added fused lasso regularization term to encourage spatial clustering of $\boldsymbol{\beta}$. As a result, when estimating the arsenic contamination probabilities from this penalized objective function $Q(\boldsymbol{\beta})$, we not only use the information from the binary arsenic testing data in the first likelihood term, but also take into account the spatial information from the spatial-graph based fused lasso penalty in the second term. ρ is a regularization tuning parameter determining the strength of fused lasso penalty and ultimately influencing the estimated number of clusters of wells. The solution of L_1 norm penalty results in an exact fusion or separation between $\beta(\mathbf{s}_i) - \beta(\mathbf{s}_j)$, that is, the edges in the graph are classified into two sets, one consists of all the non-zero elements of $\beta(\mathbf{s}_i) - \beta(\mathbf{s}_j)$ corresponding to pairs of neighboring wells that have different contamination probabilities, and the other set consists of all the zero elements of $\beta(\mathbf{s}_i) - \beta(\mathbf{s}_j)$ corresponding to pairs of neighboring wells that share the same contamination probability. As such, this regularization automatically leads to spatially clustered contamination probabilities.

The choice of graph plays two important roles in the method; it not only reflects the prior information about the geological topology and spatial clustering constraint of the data, but also determines the computation complexity of the algorithm. Some natural graph choices for spatial data include the k nearest neighbor graphs, graphs connecting neighbors within a certain radius, and spatial Delaunay triangulation graphs (see, e.g., Li and Sang (2019)). Alternatively, graphs can be constructed based on some preliminary estimates of parameters. For instance, the differences between the initial estimates of parameters at any two vertices can be used as the distance metric between vertices to replace the spatial Euclidean distance when constructing graphs. In this chapter, we take a hybrid approach to construct the graph; the k nearest neighbor edge set connecting counties is determined based on the sample proportion within each county, and the k nearest neighbor edge set within each county is determined based on the Euclidean distance.

There are several advantages of using the fused lasso penalty function for cluster detection. First, this penalization allows to detect clusters and estimate model parameters simultaneously. Second, this method guarantees to achieve a spatially contiguous clustering configuration such that only adjacent locations are clustered together. Another appealing property of this method is that the resulting clusters have very flexible shapes. We explain this point using the notion of connected components in graph theory; spatially contiguous clusters can be defined as the connected components of a graph G, and accordingly, a spatially contiguous partition of V can be defined as a collection of disjoint connect components such that the union of vertices is V. It is easy to show that any spatially contiguous partition with arbitrary cluster shapes can be recovered by removing a set of edges from a spatial graph (Li and Sang, 2019). In addition, the number of clusters does not need to be fixed a priori. Instead, we can determine it by a data-driven information criterion approach described later in this section. Finally, besides its capability to capture piece-wise constant coefficients, previous theoretical studies proved that this penalty has a strong local adaptivity in that it is also capable of capturing piece-wise Lipschitz continuous functions (Madrid Padilla et al., 2020), which implies that the method can also approximate a spatially smoothly varying contamination probability reasonably well.

We now discuss how to solve the optimization in (3.4) to obtain the parameter estimation results. Note that $-\frac{1}{n}\ell(\boldsymbol{\beta})$ is convex and differentiable with respect to $\boldsymbol{\beta}$, and $\sum_{(i,j)\in E}|\beta(\mathrm{s}_i) - \beta(\mathrm{s}_j)|$ is also convex. Therefore we propose an iterative algorithm combining the proximal gradient method (Beck and Teboulle, 2009) and the alternating direction method of multipliers (ADMM) (Boyd et al., 2011) for this convex optimization problem. Specifically, given the current estimate $\boldsymbol{\beta}^{(t)}$, we let $\mathrm{g}^{(t)} = \boldsymbol{\beta}^{(t)} + (1/L)\frac{1}{n}\nabla\ell(\boldsymbol{\beta}^{(t)})$, where L is the Lipschitz constant of $-\frac{1}{n}\ell(\boldsymbol{\beta})$, and $\nabla\ell(\boldsymbol{\beta}^{(t)})$ is the first derivative of $\ell(\boldsymbol{\beta})$ evaluated at $\boldsymbol{\beta}^{(t)}$. For the logistic regression model in (3.2), we can choose L to be $1/n$. Following the proximal gradient algorithm, we then update the value of $\boldsymbol{\beta}$ by solving:

$$\boldsymbol{\beta}^{t+1} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2} \left\| \boldsymbol{\beta} - \mathbf{g}^{(t)} \right\|_2^2 + \frac{\rho}{L} \sum_{(i,j)\in E} |\beta(\mathbf{s}_i) - \beta(\mathbf{s}_j)|. \tag{3.5}$$

We use the ADMM algorithm (Wahlberg et al., 2012) to solve the optimization in (3.5). We will release the R code of our algorithm as a supplementary file upon acceptance of this manuscript for publication.

Finally, the parameter estimation algorithm involves the selection of the tuning parameter ρ. In high dimensional statistics, data-dependent model selection criteria, such as generalized cross-validation (Golub et al., 1979), Bayesian information criterion (BIC) (Schwarz et al., 1978) and extended Bayesian information criterion (Chen and Chen, 2008) have been commonly used to determine the value of ρ. For the numerical studies in this chapter, we use BIC with the form, $\texttt{BIC} = -2\ell(\hat{\beta}) + k \log n$, where k is the estimated number of clusters. The “optimal" ρ is selected by minimizing BIC from a candidate set.

3.2.3 Spatially adaptive sampling design

We now turn the attention to the sampling design problem for the determination of the sample size and sample locations of wells. Recall in Section 3.2.2 we have obtained a spatially clustered contamination probability $p(\mathbf{s})$, that is, within each identified spatial cluster, each sample is assumed to have the same probability of being contaminated. This allows us to employ existing sampling design methods based on the univariate binomial distribution with a constant p within each cluster, while adapting to the value of p across clusters. The method leads to a simple but efficient sampling strategy accounting for the spatial variation in $p(\mathbf{s})$.

Sample size determination and confidence interval construction methods for a constant- proportion binomial distribution have been well studied in the statistics literature. Popular methods include the Clopper-Pearson exact method, Wilson score method, Wald test, Bayesian Jeffreys method, and Agresti–Coull method, among others. For a review and comparison of different methods, see, for example, Newcombe (1998) and Gonçalves et al. (2012). In this work, we consider two methods, the modified Jefferey and the Wilson score methods, following the recommendations by Gonçalves et al. (2012).

Consider a univariate binomial distribution where a random sample of size n is drawn from a large population, X is the number of 1's (e.g., the number of contaminated wells), and p is the probability of a randomly selected well is contaminated. We seek to find the sample size, n, such that, for a given p and acceptance precision level δ, the expected length of the confidence interval, $\mathrm{EL}(n, p) := \mathrm{E}\left[\Delta(X)\right]$ is equal to 2δ, where $\Delta(X)$ is the length of confidence interval, and the expectation is taken over the binomial distribution of X. The modified Jefferey and the Wilson score methods are described below.

1. The Wilson score test confidence interval takes the form

$$\frac{2X + z^2_{1-\alpha/2} \pm z_{1-\alpha/2}\sqrt{z^2_{1-\alpha/2} + 4X(1 - X/n)}}{2\left(n + z^2_{1-\alpha/2}\right)}$$

   This method is derived from Pearson's chi-square test, where the center of the interval is a weighted average of sample proportion and 1/2, such that it is more suitable than the commonly used Wald method for extreme probability or small sample sizes. The Wilson method also has the advantage of yielding an analytical formula for the sample size as follows

$$n_W = \frac{-z^2_{1-\alpha/2}[4\delta^2 - 2p(1-p)] + z^2_{1-\alpha/2}\sqrt{[4\delta^2 - 2p(1-p)]^2 - 4\delta^2\left(4\delta^2 - 1\right)}}{4\delta^2}, \quad (3.6)$$

   where n_W is the required sample size for a given estimate of p and an acceptance precision level δ.

2. The modified Jeffreys method is derived from a Bayesian approach, which uses the non-informative Jeffrey's prior $\mathrm{Beta}(1/2, 1/2)$ to derive the posterior credible interval for p, while modifying the formula at the boundary values. For $1 < X < n$, the credible interval is

$$\left[\mathrm{Beta}_{\alpha/2}(X + 1/2, n - X + 1/2), \mathrm{Beta}_{1-\alpha/2}(X + 1/2, n - X + 1/2)\right]$$

The expressions when X takes boundary values are provided in Table 1 of Gonçalves et al. (2012). The modified Jeffreys method enjoys similar coverage properties as those of the Wilson score method. But it has an additional advantage of yielding a credible interval that is equal-tailed. For modified Jeffreys,

$$\text{EL}(n;p) = \sum_{X=1}^{n} \Delta(X) \binom{n}{X} p^X (1-p)^{n-X},$$

which is a function of sample size n depending on a given p. The sample size can be calculated by solving $\text{EL}(n;p) = 2\delta$. It follows that the required sample size using the modified Jeffreys method, denoted as n_J, takes the form

$$n_J = \text{EL}^{-1}(2\delta;p). \tag{3.7}$$

n_J does not have a closed form and has to be solved numerically. In practice, it is often calculated by an approximated solution such that $|\text{EL}(n;p) - 2\delta|$ is less than a certain tolerance.

Spatial sampling design involves the determination of sample size, as well as the locations of sampling points. One simple and commonly used spatial sampling design is the uniform random sampling, where each location is chosen independently and uniformly within each cluster. However, two complications arise when applying this method for the Arsenic study. First, the number of all available candidate wells are not uniformly distributed in space. Second, a large number of wells have been tested where the sampling locations were arbitrarily chosen before the formal statistical sampling design, which results in a highly unbalanced sampling in space with some areas over-sampled and the other areas insufficiently-sampled. The design for the new sample well locations needs to exclude those previously tested wells. Our goal is to sample the candidate wells with the expectation that the combined new sample wells and the previously tested wells are spatially uniformly distributed in each cluster except for the over-sampled areas. To achieve this goal, we utilize the connection between the uniform distribution in space and the spatial Poisson point process model, and adopt the thinning sampling idea from the latter. As a preliminary, we introduce the intensity function of the spatial point processes (Diggle, 1985), which characterizes the probability that a point occurs in an infinitesimal ball around a given location. If there is a point process $\mathcal{X}$ on $D \subset \mathbb{R}^2$, let $N(B)$ denote the expected number of points within any subset $B \subset D$. The intensity function $\lambda(\mathrm{s})$ at location $\mathrm{s} \in D$ is defined as,

$$\lambda(\mathbf{s}) = \lim_{|b(\mathbf{s})| \to 0} \frac{N(b(\mathbf{s}))}{|b(\mathbf{s}) \cap D|}$$

where $b(\mathrm{s})$ denotes a small ball containing s, and measure $|\cdot|$ denotes the area. If $\lambda(\mathrm{s}) = \lambda$ is a constant for all $\mathrm{s} \in B$, then $\mathcal{X}$ is called a homogeneous point process on B, implying the point has the same probability to occur at each location in B. Besides, the intensity function determines the expected number of points on B by $\mathrm{E}[N(B)] = \int_B \lambda(\mathrm{s}) d\mathrm{s}$. It is known that, conditional on the number of points, the locations from a homogeneous Poisson point process are uniformly distributed on B. Therefore, the desired sample well locations have the intensity function $\hat{\lambda}(\mathrm{s}) = n_i/a_i$ for s located in cluster i, to render the sampled wells evenly-distributed. Here n_i and a_i denote the number of required samples and the area in cluster i respectively.

The detailed sampling algorithm is described below. First, we use the nonparametric intensity estimation approach via `R` function `density.ppp` in package `spatstat` to estimate the candidate well intensity function, denoted as $\hat{\lambda}^{candi}(\mathrm{s})$, and the previously tested well intensity, denoted $\hat{\lambda}^{exist}(\mathrm{s})$. To exclude the previously tested wells in Iowa from new samples, we calculate the target intensity from $\hat{\lambda}^{targ}(\mathrm{s}) = \max\{\hat{\lambda}(\mathrm{s}) - \hat{\lambda}^{exist}(\mathrm{s}), 0\}$. Locations that have negative $\hat{\lambda}(\mathrm{s}) - \hat{\lambda}^{exist}(\mathrm{s})$ values correspond to the over-sampled areas where the intensity of previously tested wells exceeds the required sampling density. We will leave them out when drawing new samples. Finally, for other areas, each candidate well will be selected with probability $\hat{\lambda}^{targ}(\mathrm{s})/\hat{\lambda}^{candi}(\mathrm{s})$, where s is the location of the candidate well. The last step is based on the assumption that $\hat{\lambda}^{candi}(\mathrm{s})$ is large enough to bound $\hat{\lambda}^{targ}(\mathrm{s})$, and indeed there are adequate wells available in Iowa to meet this assumption. As a result, the algorithm guarantees that the combined new samples and existing samples other than the over-sampled areas will be (nearly) uniformly distributed, and the expected

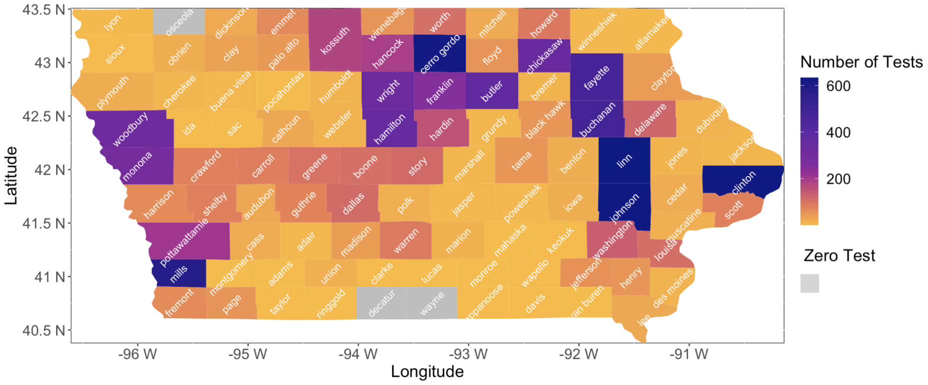

Figure 3.1: The number of tested wells in each county in Iowa.

sample size meets the requirement in Table 3.1.

3.3 Results

3.3.1 Descriptive Statistical Analysis Results

After the data pre-processing steps, there remain 9842 observations at different locations. Figure 1.2 shows the spatial distribution of the observations, and Figure 3.1 shows the spatial map of the number of observations in each county. From the existing tested data, the most tested regions include northern central Iowa, a few counties in the western central, southwestern, and eastern central Iowa regions (Figure 3.1). Less than 20% of the counties have more than 100 tests per county. There are fewer tests per county in the southern, northeastern, and northwestern regions. We show in Figure 3.2 the sample proportion of the contaminated wells among all the tested wells at each county, as a means to visualize a rough empirical estimate of the arsenic risk and its spatial pattern. Even though we see an uneven testing distribution, which means uneven sampling at the current testing scale, we observe that the arsenic risk characterization appears to be independent of the testing density (Figures 3.1 and 3.2).

3.3.2 Risk clusters and regional management

Ayotte et al. Ayotte et al. (2017) use a predictive logistic regression model to estimate arsenic presence in regions with limited arsenic data. In that study, a total of 20450 domestic well samples

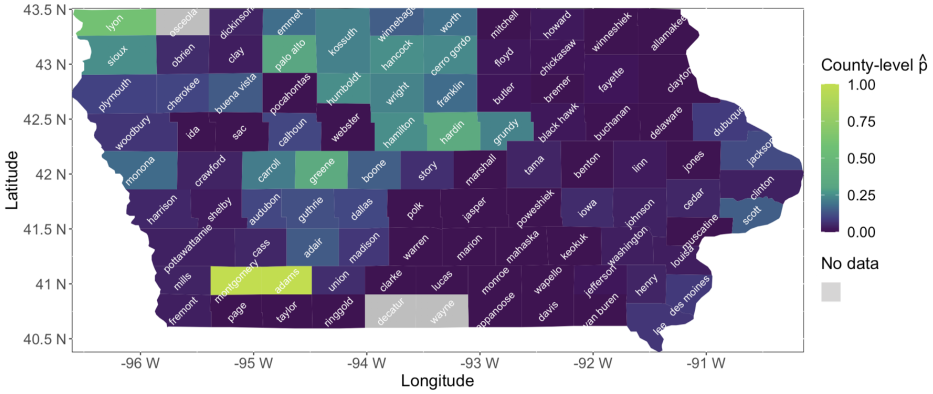

Figure 3.2: This figure illustrates the sample proportion of the contaminated wells among all the observed tested wells for each county in Iowa; Grey color indicates there is no observed data in the county.

are used to develop the model to estimate for the whole conterminous US. Unique to our research, we do not aim to establish a predictive model to accurately predict the arsenic contamination in a given region, as the risk of As has been already recognized by the state and many local health risk management agencies. We aim to utilize the locally clustered arsenic risks to estimate a sample size with minimum bias, which can be managed with appropriately allocated resources. In order to do that, we define the binary existence of arsenic in a given private well is higher than 0.01 mg/L, which is the current EPA regulation level. In other words, we regard if the private well contains less than 0.01 mg/L arsenic, then the health risks are absent in a risk-based sampling scheme. We first model and estimate the underlying contamination risk as a spatially clustered function following the method described in Section 3.2.2 for the straightforward interpretation of the result and easy implementation of the sampling design. The optimization result partitions the whole state into three risk clusters based on the estimated arsenic presence probability (Figure 3.3). The three risk probabilities (p) are 0.03, 0.21, and 0.33 for clusters 1, 2, and 3, respectively. The risk cluster assignment is consistent with some previous observations and predictions. For example, cluster 1 is largely consistent with the estimations in Ayotte et al. (2017). Cluster 2 is also highlighted with potential high As contamination in the same study. Furthermore, a targeted As study performed in Cerro Gordo County (Northern Central Iowa) has sampled 68 wells over three years

(Schnoebelen et al., 2017). The study reveals one potential mechanism of As mobilization in the shallow aquifer. The naturally occurring sulfide minerals (typically pyrite) in the bedrock aquifers could be the source of As. Under the oxidizing condition, the As mobilization could happen from rocks to the water. Significantly, the Cerro Gordo study has resulted in a policy change for arsenic testing and well completion locally. Interestingly, cluster 3 at the border of Iowa and Nebraska is identified as a new As "hotspot" in this current study. Notably, the cluster 3 region overlaps with the Missouri alluvial plain. The Missouri River valley contains up to around 150 feet of highly-permeable alluvial sediments. Alluvial sediments could be quite heterogeneous in their gravel, sand, silt, and clay compositions, dependent on the locations. At the same time, those sediments could contain a large percent of argillaceous materials composed of organics, clays, and silts. The presence of argillaceous materials could assist in disseminating arsenic pyrite from the materials themselves or from ferrous hydroxides coating the sand grains, which often contain arsenic as well. Furthermore, diverse geochemical and bacterially mitigated reactions (i.e., oxidation, reduction, adsorption, precipitation, methylation, and volatilization) can participate actively in arsenic recycling within alluvial aquifers. As the alluvial aquifers are largely unconfined, the water table's movement up and down in the aquifer can mobilize arsenic from the argillaceous material or the ferrous hydroxide coating the sand grains through oxidation reactions. The potential high arsenic concentration in the private well in the alluvial plain (i.e., cluster 3) could be attributed to the permeable alluvial sediments and their unique properties.

3.3.3 Sample design

Based on the estimated probability clusters, we further estimate the ideal sample size based on various acceptance precision and confidence levels. Based on the publicly available database (Iowa Private Well Tracking System), it is estimated there are more than 300,000 private wells in Iowa. Among them, $291,882$ wells are geo-coded. The total geo-coded well population locations are shown in Figure 3.4, clearly indicating an uneven spatial distribution in Iowa. Based on the regional cluster risk probability, we thus define three different cluster regions (clusters 1, 2, and 3) with different risk cluster ranks. For clusters with a reasonable testing coverage, we have three

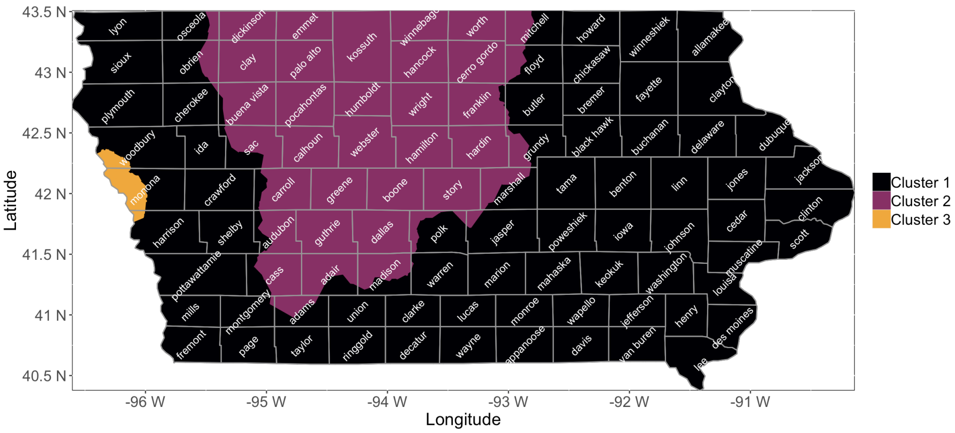

Figure 3.3: Partition of the map in terms of estimated p; In cluster 1, $\hat{p} = 0.02869485$; In cluster 2, $\hat{p} = 0.2088291$; In cluster 3, $\hat{p} = 0.3373494$; The numbers of observations in each cluster are 6482, 3194 and 166 respectively.

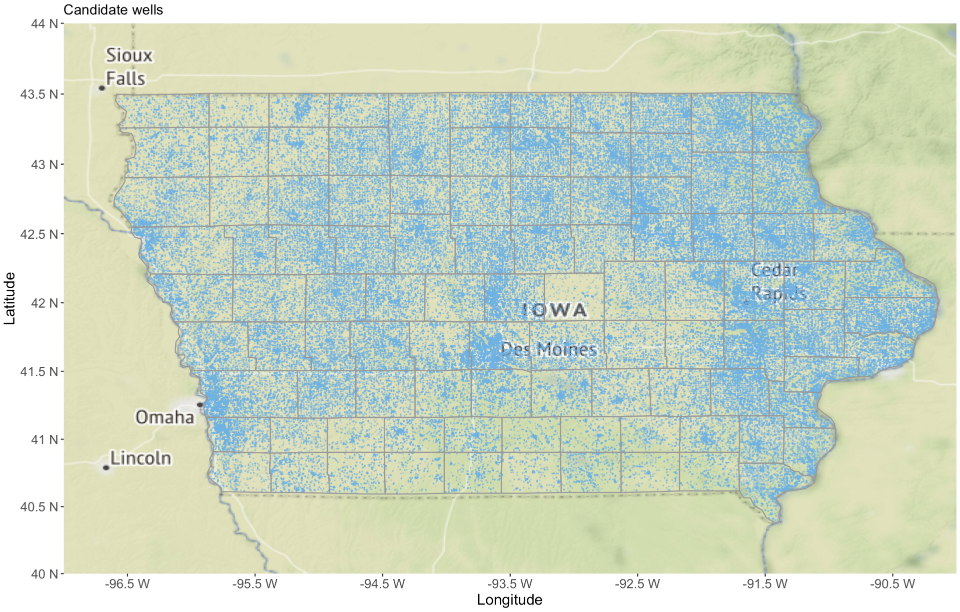

Figure 3.4: Locations of the $291,882$ candidate wells in Iowa, after discarding the wells in absence of their location information.

Table 3.1: Expected number of well sampling in each cluster;

Confidence Level	Method	Cluster 1	Cluster 2	Cluster 3
90%	Wilson	8766	1017	523
	Jeffrey	8746	1015	523
95%	Wilson	12446	1444	743
	Jeffrey	12420	1442	743
99%	Wilson	21497	2493	1282
	Jeffrey	21456	2492	1284

probabilities. For cluster 1, the estimated probability for As concentration higher than 0.01 mg/L probability is 0.03. For clusters 2 and 3, the probabilities are 0.21 and 0.34, respectively. If we define the precision acceptance as 10% of the probability, the precision acceptance is 0.003 for cluster 1 (e.g. 10% of 0.03), 0.021 for cluster 2, and 0.034 for cluster 3. Table 3.1 provides the calculated required sample size for each cluster under three different confidence levels (90%, 95%, and 99%) using both the Wilson in (3.6) and Jeffrey methods in (3.6). For example, at the 95% confidence interval, the estimated sample size based on the Jeffrey method is 12446 for cluster 1. Applying the same criteria to clusters 2 and 3, the estimated sample size would be 1442 for cluster 2 and 743 for cluster 3. The sample sizes calculated by the Wilson method only slightly differ from those of the Jeffrey method. Accordingly, at the 99% confidence interval, we estimate that 21456, 2492, and 1282 samples are needed for clusters 1, 2, and 3, respectively.

In the existing As data set, there are 6482, 3194, and 166 tested wells already collected from clusters 1, 2 , and 3, respectively. It is noted that the sample size of the tested wells within each cluster constitutes a large proportion or exceeds the required sample size calculated in Table 3.1. However, we recognize the current As data collection is operated at the county level since the local environmental health jurisdiction resides in each county. County level data generation results in an uneven spatial distribution of sampling locations for the whole state discussed in Section 3.3.2. Therefore, although some areas are over-sampled, new samples still need to be collected at those places that are only sparsely sampled previously.

We follow the method presented in Section 3.2.3 to determine the locations of new sampling

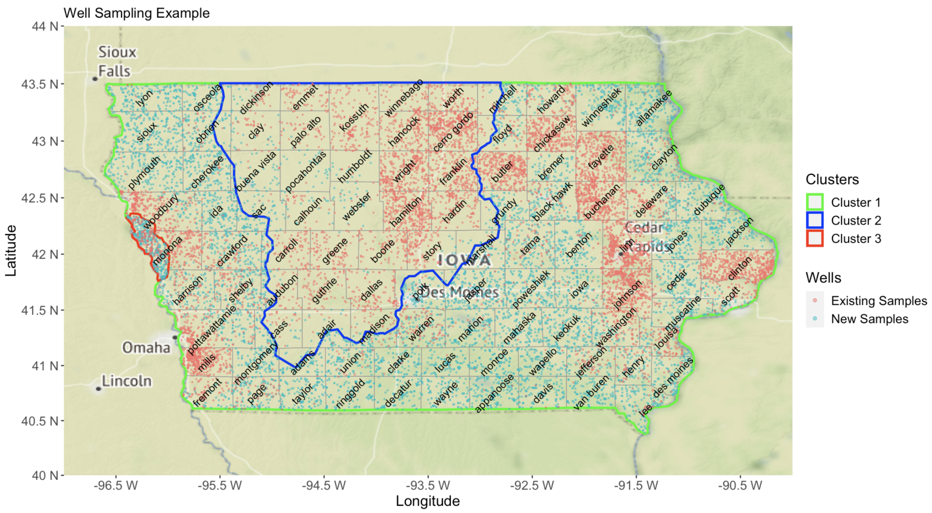

Figure 3.5: An example of sampling results; 8174, 313 and 586 additional wells are sampled in each cluster respectively in this example.

locations. We use the private wells in the current Iowa PWTS database as the target sampling population (Figure 3.4). The goal is to achieve a spatially balanced sampling design that meets the required sample size, while accounting for the fact that both the candidate wells and existing tested wells are distributed highly non-uniformly in space. To illustrate, we give an example of the sampling scenario using the sample size calculated from the Wilson method for the 95% CI in Figure 3.5. The dense red point clouds reveal the previously over sampled areas in this Figure. Cluster 2 has the largest proportion of previously over-sampled areas. Only a relatively small number of additional wells (marked by green dots) need to be sampled, mostly are located in the middle west of this region. In contrast, most areas in cluster 1 have not been sampled and tested previously, with exceptions in several counties (e.g., Buchanan, Butler, and Clinton). In cluster 3, although the spatial coverage of the existing tested samples is nearly uniform, our method suggests that an additional number of wells need to be collected to achieve the desired confidence level and precision accuracy. Overall, it is noted that the locations of samples in Figure 3.5 appear to be uniformly distributed except for the previously over-sampled areas. Looking more closely, we observe that the intensity/density of samples differs across the identified risk clusters, due to adopting a spatially adaptive sampling design according to each cluster's own contamination risk.

3.4 Conclusions

It is commonly recognized that many conditions such as geological, geochemical, and hydrologic variables, impact arsenic presence in groundwater. For example, It has been observed high arsenic concentrations are often found in more arid western US (Ayotte et al., 2017). Furthermore, precipitation and recharge show significant correlations with arsenic concentrations in domestic wells in the conterminous US. Among various conditions, glaciated terrain, bedrock geology, soil hydrology, soil tile drainage, water table depth and climate factors can also impact arsenic concentrations in groundwater. Particularly, Iowa's groundwater resources are majorly surficial aquifers and bedrock aquifers. For a long history contacting with glaciers, many parts of Iowa soil/dirt contain glacier age materials with moderate to low permeability. The water table beneath those materials occurs at relatively shallow depths and varies from 3 to 30 feet below ground (Prior et al., 2003).The micro-environment such as pH, soil, and water bacterial activity, oxidation and reduction reactions (Redox), coexistence with other elements (e.g., iron) can also play a significant role in arsenic concentration in groundwater. Taking account of all those macro and micro-environmental conditions is a shared challenge for all current available predictive models to estimate arsenic concentrations at the county, state/province, or region levels.

There are several potential benefits to adopt the proposed sampling design. First, the sample size estimate suggests future feasible random sampling targets, given the total Iowa private well population. As the sample sizes are dependent on the arsenic probabilities, we present options for the same probability with different sampling precision goals. We also recognize there are regions with too few or no data points (Figure 3.2, thus warrant further sampling for probability estimate). Second, the method developed in this study helps pinpoint future sampling locations with adequate statistical power. From the resource management perspective, future planning can prioritize the high-risk well sampling, eliminate redundant testing, and collect representative samples for risk assessment purposes. In practice, sample collections and management are often conducted at certain administration levels. It is desired to develop a sampling design method that is easy and fast to implement at each administration unit. Third, this design presents future opportunities to investigate practical solutions to coordinate joint efforts across counties for the efficient implementation of the sampling design method.

Moving forward, this work could be further refined in several ways. First, the estimator we obtained by optimizing the regularized log-likelihood function does not come with an uncertainty measure. As such, the sample size calculation is only based on a point estimate of the contamination risk. A potential solution is to consider a Bayesian version of the method. In principle, the modified Jeffrey's method for sample size calculation can be adapted to account for the uncertainty in the estimate of the contamination probability p, where the expected length of the confidence interval used in (3.7) can be taken over both the distributions of p and the binomial random variable X instead of X only. Second, we assume that the clusters of wells are spatially contiguous, and the contiguity constraint is defined with respect to the choice of a spatial graph. However, in practice, the spatial contiguity constraint may not dominate the clustering configuration globally, in the sense that two or more locally contiguous clusters that are remote in space may actually have very similar arsenic concentrations, and hence should be classified into the same cluster. The method presented in this chapter needs to be modified to handle the case with both globally dis-contiguous and locally contiguous clusters. One idea is to perform a two-step analysis, where in the first step we obtain local spatial clusters from the method presented in this chapter, and in the second step, conduct another clustering analysis without any spatial constraint based on the local clustering results from the first step. Third, the model can be further improved with more representative samples. As we noted, there are counties without testing data, which presents a gap for risk analysis. We expect collecting data in those regions helps build a more comprehensive evaluation of arsenic health risk at the state level. Overall, the current study presents a targeted approach to save cost and time for effective public health management strategy. The rational sampling design focuses on risk categories, which assures that preventive measures and mitigation practices are implemented where most needed.

4. ROW-CLUSTERING OF A POINT PROCESS-VALUED MATRIX

4.1 Introduction

Large-scale, high-resolution, and irregularly scattered event time data has attracted enormous research interest recently in many applications, including medical visiting records (Lasko, 2014), financial transaction ledgers (Xu et al., 2020) and server logs (Husin et al., 2013). Given a collection of event time sequences, one research thread is to identify groups displaying similar patterns. In practice, the significance of this task emerges in multifarious scenarios. For example, matching users with similar activity patterns on social media platforms is beneficial to ads recommendations; clustering patients by their visiting records may help predict the course of the disease progression.

Our study is motivated by a dataset we collected from Twitter, which consists of posting times of 500 university's official accounts during April 15, 2021 to May 14th, 2021. Figure 1.3 displays posting time stamps of seven selected accounts in five consecutive days. While the daily posting patterns varied across different accounts, the date on which a posting was made seemed to also play an important role. Specifically, all accounts cascaded a barrage of postings on April 16th while few postings appeared on April 18th. Lastly, each posting was associated with a specific type of activity, namely, tweet, retweet, or reply. Our main interest is to cluster these multi-category, dynamic posting patterns into subgroups.

To characterize the highly complex posting patterns, we propose a mixture model of Multi-level Marked Point Processes (MM-MPP). We assume that the event sequences from each cluster are realizations of a multi-level log-Gaussian Cox process (LGCP) (Møller et al., 1998), which has been demonstrated useful for modelling repeatedly observed event sequences (Xu et al., 2020) . We here extend their work to the case of mixture models and propose a semiparametric Expectation-Solution algorithm to learn the underlying cluster structure. The proposed learning algorithm avoids iterative numerical optimizations within each ES step and hence is computationally efficient. In addition, we design an algorithm that can take advantage of array programming and GPU acceleration to further speed up computation.

In summary, our main contributions in this chapter are two folds: (1) we propose an MM-MPP model for repeatedly observed multi-category event sequences; and (2) we develop a highly efficient semiparametric ES algorithm for event sequence clustering.

4.2 Related Work

Modelling of Event Sequences The most traditional models for event sequences can stretch back to the time series models with discretized time-lagged observations (Liao, 2005; Maharaj, 2000; Van Wijk and Van Selow, 1999). These methods rely on the mixture of time series models, such as the ARMA model (Kalpakis et al., 2001) and the Markov model (Ge and Smyth, 2000; Luo et al., 2016). This kind of model have two major issues. The first is that they always depend on certain assumptions, such as auto-regression (Kalpakis et al., 2001) or stationarity (Ge and Smyth, 2000). Point processes have been widely used to model event sequences (Daley and Vere-Jones, 2003), although most existing work rely on strong parametric assumptions. One prominent example is the Hawkes process (Hawkes and Oakes, 1974), which accounts for temporal dependence among events by a self-triggering mechanism. However, existing Hawkes processes often assume that the triggering function only relies on the distance between two time points and hence are not suitable to model the data in our case that have multi-level variations. One way to account for variations from multiple sources is to exploit Cox process models, whose intensities are modeled by latent random functions. One popular class of Cox processes is the log-Gaussian Cox process (LGCP) (Møller et al., 1998), whose latent intensity functions are assumed to be Gaussian processes. Recently, Xu et al. (2020) proposed a multi-level LGCP model to account for different sources of variations for repeatedly observed event data. However, clustering of repeatedly observed marked event time data was not considered in their work.

Clustering of Event Sequences. Extensive research has been done on this topic. To our knowledge, clustering models for point processes can be summarized into two major categories: distance-based clustering (Berndt and Clifford, 1994; Bradley and Fayyad, 1998; Pei et al., 2013) and distribution-based clustering (Xu and Zha, 2017; Luo et al., 2015). The former measures the closeness between event sequences based on some extracted features and then uses classical distanced-based clustering algorithms such as k-means (Bradley and Fayyad, 1998; Peng et al., 2008) or EM algorithms (Wu et al., 2020a). The second approach, also referred to as model-based clustering, assumes that event sequences are derived from a parametric mixture model of point processes. One notable thread is the mixture model of the Hawkes point processes. For example, Xu and Zha (2017) proposed a Dirichlet mixtures of Hawkes processes (DMHP) under the Expectation-Maximization (EM) framework to identify clusters. However, existing EM algorithms for event sequence clustering have a common issue that they typically require iterative numerical optimizations within each M-step, which would drastically overburden the computation. This computational issue will be accentuated when event data are repeatedly observed and have marks.

4.3 Model-based Row-clustering for a Matrix of Marked Point Processes

Notation. Mathematically, suppose that we observe daily event sequences from n accounts during m days. For account i on day j, let $N_{i,j}$ denote the total number of events, $t_{i,j,l} \in (0, T]$ denote the l-th event time stamp, and $r_{i,j,l} \in \{1, \cdots, R\}$ denote the corresponding event types (marks). The activities of account i on day j can be summarized by a set $S_{i,j} = \{(t_{i,j,l}, r_{i,j,l})\}_{l=1}^{N_{i,j}}$, recording the time stamps and types for all $N_{i,j}$ events. This general notation can also describe other marked event sequences which are repeatedly observed on m non-overlapping time slots. We represent the collection of all marked daily event sequences as an $n \times m$ matrix $\mathcal{S}$, whose (i, j)th entry is a marked event sequence $S_{i,j}$. We aim to cluster the rows of $\mathcal{S}$ to identify potential heterogeneity in account activity patterns, while taking into account the dependence across rows and columns to characterize the complex event patterns and interactions among accounts, days, and event types.

4.3.1 A Mixture of Multi-level Marked LGCP Model

Given a matrix of daily event sequences $\mathcal{S}$, we can separate each matrix entry $S_{i,j}$ according to their marks (event types). Let $S_{i,j}^{r} = \{t_{i,j,l} | r_{i,j,l} = r\}$ record the time stamps of event type

$r \in \{1, \cdots, R\}$. We model each $S^r_{i,j}$ by an inhomogeneous Poisson point process conditional on a latent intensity function: $\lambda^r_{i,j}(t|\Lambda^r_{i,j}) = \exp\{\Lambda^r_{i,j}(t)\}$, where $\Lambda^r_{i,j}(t) : [0, T] \mapsto \mathbb{R}$ is the random log intensity function on $[0, T]$. Following Xu et al. (2020), we assume a multi-level model for $\Lambda^r_{i,j}(t)$:

$$\Lambda^r_{i,j}(t) = X^r_i(t) + Y^r_j(t) + Z^r_{i,j}(t), \quad t \in [0, T], \tag{4.1}$$

for $i = 1, \cdots, n$, $j = 1, \cdots, m$ and $r = 1, \cdots, R$. In model (4.3.1), $X^r_i(t)$, $Y^r_j(t)$ and $Z^r_{i,j}(t)$ are random functions on $[0, T]$, characterizing the variations of account i, day j and the residual deviation, respectively. In addition, we also take into account the dependence across event types when modelling $X^r_i(t)$, $Y^r_j(t)$ and $Z^r_{i,j}(t)$, while assuming independence across accounts, that is, for any (r, r'), $X^r_i(t)$ and $X^{r'}_{i'}(t)$ are independent when $i \neq i'$, $Y^r_j(t)$ and $Y^{r'}_{j'}(t)$ are independent when $j \neq j'$, and $Z^r_{i,j}(t)$ and $Z^{r'}_{i',j'}(t)$ are independent if $(i, j) \neq (i', j')$.

We assume that $\mathbf{X}_i(t) = \{X^r_i(t)\}^R_{r=1}$ is a mixture of multivariate Gaussian processes with C components in order to detect heterogeneous clusters. We introduce a binary vector $\boldsymbol{\omega}_i = \{\omega_{1,i}, \cdots, \omega_{C,i}\}'$ to encode the cluster membership for account i, where $\omega_{c,i} = 1$ if account i belongs to the c-th cluster and 0 otherwise. In analogy to other model-based clustering approaches, the unobserved cluster membership $\boldsymbol{\omega}_i$ are treated as missing data and assumed to follow a categorical distribution with parameter $\boldsymbol{\pi} = \{\pi_1, \cdots, \pi_C\}$, where π_c indicates the probability that an account belongs to the c-th cluster. Conditional on $\boldsymbol{\pi}$, we assume that $X^r_i(t)$'s in different clusters have heterogeneous behavioral patterns, characterized by their corresponding cluster-specific multivariate Gaussian processes with mean functions $\mu^r_{x,c}(t) = \mathbb{E}[X^r_i(t)|\omega_{c,i} = 1]$ and cross covariance functions $\Gamma^{r,r'}_{x,c}(s, t) = \text{Cov}[X^r_i(s), X^{r'}_i(t)|\omega_{c,i} = 1]$, for $s, t \in [0, T]$, and $r, r' = 1, \cdots, R$. Here, $\mu^r_{x,c}(t)$ characterizes the cluster-specific first-order intensity function, and $\Gamma^{r,r'}_{x,c}(s, t)$ describes the temporal dependence patterns within and across event types.

Similarly, we assume $\mathbf{Y}_j(t) = \{Y^r_j(t)\}^R_{r=1}$ and $\mathbf{Z}_{i,j}(t) = \{Z_{i,j}(t)^r\}^R_{r=1}$ are both mean-zero multivariate Gaussian processes to account for dependence of day-level and residual random effects within and across event types, respectively. The covariance functions take the forms: $\Gamma^{r,r'}_y(t) = \text{Cov}[Y^r_j(t), Y^{r'}_j(t)]$, and $\Gamma^{r,r'}_z(t) = \text{Cov}[Z^r_{i,j}(t), Z^{r'}_{i,j}(t)]$. As the heterogeneity patterns are assumed to be mainly explained by the account-level effect $\mathbf{X}$, both $\Gamma_y^{r,r'}(t)$ and $\Gamma_z^{r,r'}(t)$ are assumed to be homogeneous across all clusters.

A Single-level Special Case. When $m = 1$, our data matrix $\mathcal{S}$ only has one column of event sequences. The multi-level model in (4.1) reduces to a single-level model:

$$\lambda_{i,1}^r(t|\Lambda_{i,1}^r) = \exp\{\Lambda_{i,1}^r(t)\}, \quad \Lambda_{i,1}^r = X_i^r(t), \quad t \in [0, T] \tag{4.2}$$

where $\mathbf{X}_i(t) = \{X_i^r(t)\}_{r=1}^R$ has the same model specification as in the multi-level case described earlier. We remark that it is still of importance to consider this special case, as even in this simpler case limited work has been done for the clustering of repeatedly observed marked point processes.

4.3.2 The Likelihood Function

We denote the parameters concerning $\mathbf{X}_i(t)$ in cluster c as $\Theta_{x,c}$ and the parameters concerning $Y_j(t)$ and $Z_{i,j}(t)$ as Θ_y and Θ_z, respectively. Therefore, the parameters in model (4.1) consist of $\Omega = \{\boldsymbol{\pi}, \Theta_y, \Theta_z, \Theta_{x,c}, c = 1, \cdots, C\}$. When $m = 1$, $\Omega = \{\boldsymbol{\pi}, \Theta_{x,c}, c = 1, \cdots, C\}$ representing the parameters involved in model (4.2). The complete data $\mathcal{D}$ consists of the observed data $\mathcal{S}$ and the unobserved latent variables $\big\{\{\boldsymbol{\omega}_i\}_{i=1}^n, \mathcal{L}\big\}$, where $\mathcal{L} = \big\{\{\mathbf{X}_i(t)\}, \{\mathbf{Y}_i(t)\}, \{\mathbf{Z}_{i,j}(t)\}\big\}$ for model (4.1) and $\mathcal{L} = \{\{\mathbf{X}_i(t)\}\}$ for model (4.2). Let S_i be the i-th row of $\mathcal{S}$ representing activities of the i-th account. In our mixture model, the probability of the observed data $\mathcal{S}$ can be written as

$$p(\mathcal{S}; \Omega) = \mathbb{E}_\omega \mathbb{E}_\mathcal{L} \left[\prod_{i=1}^n \text{PP}(S_i|\mathcal{L}) \mid \{\boldsymbol{\omega}_i\}_{i=1}^n; \Omega \right], \tag{4.3}$$

where the expectations are taken with respect to the conditional distribution of latent variables $\mathcal{L}$ and $\boldsymbol{\omega}_i$'s, and PP$(S_i|\mathcal{L})$ is the conditional probability of a Poisson point process,

$$\text{PP}(S_i \mid \mathcal{L}) = \prod_{j=1}^m \prod_{r=1}^R \left\{ \prod_{t \in S_{i,j}^r} \lambda_{i,j}^r(t \mid \Lambda_{i,j}^r) \exp\left[-\int_0^T \lambda_{i,j}^r(s \mid \Lambda_{i,j}^r) ds) \right] \right\}, \tag{4.4}$$

where, conditional on $\mathcal{L}$, $\Lambda_{i,j}^r(t)$ has the form as (4.1) for $m > 1$ and as (4.2) for $m = 1$.

4.4 Row-clustering Algorithms

Existing mixture model-based clustering methods typically rely on likelihood-based Expectation -Maximization algorithms (Aitkin and Rubin, 1985) by treating unobserved latent variables, $\{\{\boldsymbol{\omega}_i\}_{i=1}^n, \mathcal{L}\}$ in our case, as missing data. However, standard EM algorithms are computationally intractable for the models we consider here. One computation bottleneck is the numerical optimizations involved in M-steps, which require many iterations due to the lack of close-form solutions when updating parameters. Moreover, the computation burden is severely aggravated by the fact that the expectations in E-step (see (4.3) for an example) involve an intractable multivariate integration.

In Section 4.4.1, we describe a novel efficient semi-parametric Expectation-Solution algorithm for the single-level model in (4.2) to bypass the computation challenges described above. We then show in Section 4.4.2 that the learning task of multi-level models in (4.1) can be transformed and solved by utilizing an algorithm similar to that of single-level models.

4.4.1 Learning of Single-level Models

The ES algorithm (Elashoff and Ryan, 2004) is a general iterative approach to solving estimating equations involving missing data or latent variables. The algorithm proceeds by first constructing estimating equations based on a complete-data summary statistic, which may arise from a likelihood, a quasi-likelihood or other generalized estimating equations. Similar to the EM algorithm, the ES algorithm then iterates between an expectation (E)-step and a solution (S)-step until convergence to obtain parameter estimates. The detailed steps of a general ES algorithm framework are included in Supplementary S.2. The EM framework is a special case of ES when estimating equations are constructed from full likelihoods and using complete data as the summary statistic.

Due to the lack of closed-form for the likelihood function (4.3), we opt to design our algorithm under the more flexible and general ES framework for parameter estimations of the single-level models in (4.2), i.e., $m = 1$. The algorithm is summarized in Algorithm 1 and detailed below.

As a preliminary, we give the form of the expectation of the conditional intensity function given cluster memberships as follows:

$$\rho_c^r(t) = \mathbb{E}[\lambda_{i,1}^r(t) \mid \omega_{c,i} = 1] = \exp[\mu_{x,c}^r(t) + \Gamma_{x,c}^r(t,t)/2]. \tag{4.5}$$

The form of the second-order conditional intensity function is

$$\begin{aligned}\rho_{c,i}^{r,r'} &= \mathbb{E}[\lambda_i^r(s)\lambda_i^{r'}(t) \mid \omega_{c,i} = 1] \\ &= \mathbb{E}\{\exp[X_i^r(s) + X_i^{r'}(t) | \omega_{c,i} = 1]\} \\ &= \rho_c^r(s)\rho_c^{r'}(t)\exp[\Gamma_{x,c}^{r,r'}(s,t)] \end{aligned} \tag{4.6}$$

for $i = 1, \cdots, n$, $r, r' = 1, \cdots, R$, where the last equality is derived following the moment generating function of a Gaussian random variable.

Estimating Equations. We carefully construct estimating equations of unknown parameters with three considerations in mind: (1) the expectation of the estimating equations over the complete data should be zero; (2) the conditional expectation of the estimating equation can be solved efficiently in the S-step; (3) the estimating equations should be fast to calculate.

Let $K(\cdot)$ be a kernel function and $K_h(t) = h^{-1}K(t/h)$ with bandwidth h. We define

$$A_c^{r,r'}(s,t;h) = \sum_{i=1}^n \omega_{c,i} a_i^{r,r'}(s,t;h), \quad \text{where } a_i^{r,r'}(s,t;h) = \sum_{u \in S_i^r, v \in S_i^{r'}}\sum^{u \neq v} \frac{K_h(s-u)K_h(t-v)}{ng(s;h)g(t;h)};$$

$$B_c^r(t;h) = \sum_{i=1}^n \omega_{c,i} b_i^r(t;h), \qquad \text{where } b_i^r(t;h) = \sum_{u \in S_i^r} \frac{K_h(t-u)}{ng(t;h)},$$

for $c = 1, ..., C$, and $r, r' = 1, ..., R$, where $g(x;h) = \int K_h(x-t)dt$. Using the Campbell's Theorem (Moller and Waagepetersen, 2003) and the moment generating function of the normal distribution, it is straightforward to show that $\mathbb{E}\left[A_c^{r,r'}(s,t;h)|\boldsymbol{\omega}\right] \approx \pi_c\rho_c^r(s)\rho_c^{r'}(t)\exp[\Gamma_{x,c}^{r,r'}(s,t)]$ and that $\mathbb{E}\left[B_c^r(t;h)|\boldsymbol{\omega}\right] \approx \pi_c\rho_c^r(t)$, provided that h is sufficiently small. This motivates us to consider the following estimating equations:

$$\begin{cases} A_c^{r,r'}(s,t;h) - \pi_c \rho_c^r(s)\rho_c^{r'}(t)\exp[\Gamma_{x,c}^{r,r'}(s,t)] = 0 \\ B_c^r(t;h) - \pi_c \rho_c^r(t) = 0 \\ \dfrac{\sum_{i=1}^n \omega_{c,i}}{n} - \pi_c = 0. \end{cases} \tag{4.7}$$

Expectation (E-step). Given an observed data $\mathcal{S}$ and a current parameter estimate Ω^*, we calculate the conditional expectation of the estimation equations in (4.7). Note that the three estimating equations are all linear with respect to $\{\omega_{c,i}, c = 1, \cdots, C, i = 1, \cdots, n\}$. Therefore, the conditional expectations of the estimating equations are obtained by replacing $w_{c,i}$ with its conditional expectation $\mathbb{E}_{\boldsymbol{\omega}}[\omega_{c,i}|\mathcal{S};\Omega^*]$, which has the following form:

$$\mathbb{E}_{\boldsymbol{\omega}}[\omega_{c,i}|\mathcal{S};\Omega^*] = \frac{\pi_c^* f(S_i|\omega_{c,i}=1;\Omega^*)}{\sum_{c=1}^C \pi_c^* f(S_i|\omega_{c,i}=1;\Omega^*)}, \tag{4.8}$$

where $f(S_i|\omega_{c,i}=1;\Omega^*) = \mathbb{E}_{\mathcal{L}}[\text{PP}(S_i|\mathcal{L})|\omega_{c,i}=1;\Omega^*]$. Here $\text{PP}(\cdot)$ is the conditional distribution function of S_i given $\omega_{c,i}$ and Ω^* as defined in (4.4). We propose to approximate $f(S_i|\omega_{c,i}=1;\Omega^*)$ by its Monte Carlo counterpart,

$$\hat{f}(S_i \mid \omega_{c,i}=1;\Omega^*) \approx \frac{1}{Q}\sum \hat{\text{PP}}(S_i \mid \boldsymbol{X}_c^{(q)}(t)), \tag{4.9}$$

where Q is the Monte Carlo sample size, $\boldsymbol{X}_c^{(q)}(t)$'s are independent samples from the multivariate Gaussian process with parameters $\Theta_{x,c}^*$, and $\hat{\text{PP}}(\cdot)$ is a numerical quadrature approximation of $\text{PP}(\cdot)$ following Berman and Turner (1992):

$$\hat{\text{PP}}(S_i|\boldsymbol{X}(t)) = \exp\left\{\sum_{r=1}^R \sum_{u\in\tilde{S}_{i,1}^r} v_u[y_u X^r(u) - \exp X^r(u)]\right\}. \tag{4.10}$$

In the above, $\tilde{S}_{i,1}^r$ is the union of $S_{i,1}^r$ and a set of regular grid points, v_u is the quadrature weight corresponding to each u and $y_u = v_u^{-1}\Delta_u$, where Δ_u is an indicator of whether u is an observation

Algorithm 1 Learning of the Single-level model in (4.2)

```
Input: $\mathcal{S} = \{S_i\}_{i=1}^n$, the number of clusters $C$, the bandwidth $h$;
Output: Estimates of model parameters, $\hat{\boldsymbol{\pi}}$, $\hat{\mu}_{x,c}^r(t)$, $\hat{\Gamma}_{x,c}^{r,r'}(s,t)$, for $c = 1, \cdots, C, r, r' = 1, \cdots, R$;
Calculate the components $a_i^{r,r'}$'s and $c_i^r$'s;
Initialize $\Omega^* = \{\boldsymbol{\pi}^*, \Theta_{x,c}^*, c = 1, \cdots, C\}$ randomly;
Repeat:
    E-Step:
    Calculate $\mathbb{E}_{\boldsymbol{\omega}}[\omega_{c,i}|\mathcal{S}; \Omega^*]$ as (4.8);
    Calculate $\mathbb{E}_{\boldsymbol{\omega}}[A_c^{r,r'}(s,t)|\mathcal{S}; \Omega^*]$ and $\mathbb{E}_{\boldsymbol{\omega}}[B_c^r(t)|\mathcal{S}; \Omega^*]$;
    M-Step:
    Update $\boldsymbol{\pi}^*$, $\mu_{x,c}^{r*}(t)$ and $\Gamma_{x,c}^{r,r'*}(s,t)$ according to (4.11),(4.12) and (4.13);
    End;
Until: Reach the convergence criteria;
$\hat{\boldsymbol{\pi}} = \boldsymbol{\pi}^*$, $\hat{\mu}_{x,c}^r = \mu_{x,c}^{r*}(t)$ and $\hat{\Gamma}_{x,c}^{r,r'}(s,t) = \Gamma_{x,c}^{r,r'*}(s,t)$;
```

($\Delta_u = 1$) or a grid point ($\Delta_u = 0$).

Solution (S-step). In this step, we update the parameters by finding the solutions to the expected estimating equations from the E-step. For $c = 1, \cdots, C$, $r, r' = 1, \cdots, R$ and $r \neq r'$, the solutions take the following closed forms:

$$\pi_c^* = \frac{\sum_{i=1}^n \mathbb{E}[\omega_{c,i}|\mathcal{S}; \Omega^*]}{n}, \tag{4.11}$$

$$\Gamma_{x,c}^{r,r'*} = \log \frac{\hat{\pi}_c \mathbb{E}_{\boldsymbol{\omega}}[A_c^{r,r'}(s,t;h)|\mathcal{S}; \Omega^*]}{\mathbb{E}_{\boldsymbol{\omega}}[B_c^r(s;h)|\mathcal{S}; \Omega^*]\mathbb{E}_{\boldsymbol{\omega}}[B_c^{r'}(t;h)|\mathcal{S}; \Omega^*]}, \tag{4.12}$$

$$\mu_{x,c}^{r*}(t) = \log \frac{\mathbb{E}_{\boldsymbol{\omega}}[B_c^r(t;h)|\mathcal{S}; \Omega^*]}{\hat{\pi}_c \exp[\hat{\Gamma}_{x,c}^r(t,t)/2]}. \tag{4.13}$$

Sampling Strategies. The multi-dimensional functional form of $\boldsymbol{X}_c^{(g)}$ renders the sampling procedures in (4.9) intractable. Given the parameters $\Theta_{x,c}$, one solution is to find a low-rank representation of $\boldsymbol{X}_i$ with the functional principal components analysis (FPCA) (Ramsay and Silverman, 2005). Specifically, we approximate the latent Gaussian process $\boldsymbol{X}_i$ in cluster c nonparametrically using the Karhunen-Lòeve expansion (Watanabe, 1965) as: $X_i^r(t) = \boldsymbol{\mu}_c + < \boldsymbol{\xi}_i, \boldsymbol{\phi}(t) >$, for

$r = 1, \cdots, R$, where $\boldsymbol{\xi}_i$ is a vector of normal random variables, and $\boldsymbol{\phi}(t)$ is a vector of orthogonal eigenfunctions. Using FPCA, we can obtain the sampling of $\boldsymbol{X}_i$ from the sampling of $\boldsymbol{\xi}_i$ indirectly. More detailed sampling procedure via FPCA can be seen in Supplementary B.2.

Model Selection. Our clustering procedures require choosing the proper number of clusters C and bandwidth h. In model-based clustering, one popular method for choosing the number of clusters is based on the Bayes information criterion (BIC) (Schwarz et al., 1978). Since the probability $f(\mathcal{S}|\boldsymbol{\omega})$ is already calculated in each iteration, one can directly use this term for the BIC calculation. The choice of kernel bandwidth h also plays an important role for model stability. A small h may produce unstable clustering results while a large h would dampen the characteristics of each cluster. We use an adaptive h that maximizes the likelihood in each iteration. Specifically, we pre-calculate a series of $a_c^{r,r'}$ and b_c^r for different candidates of h. Then in each iteration, we select the one that gives the maximum likelihood.

Remarks. The most significant advantage of our method is that it avoids expensive iterations inside each E-step and S-step, unlike other EM algorithms for mixture point process models (Xu and Zha, 2017). The elements $a_{i,h}^{r,r'}$ and $c_{i,h}^r$ in the estimating equations can be pre-calculated before E-S iterations to save computations. Moreover, the S-step is fast to execute thanks to the closed-form solutions. We will analyze the overall computation complexity of the learning algorithm in Section 4.4.4.

4.4.2 Learning of Multi-level Models

We now consider developing the learning algorithm of multi-level models in (4.1), assuming we repeatedly observe R types of events from n accounts on m days with $m > 1$. Below, we propose a method to transform the learning task of a multi-level model into a problem that can be solved by a two-step procedure, where the second step is mathematically equivalent to a single-level model and hence can be conveniently solved by a similar algorithm as in Algorithm 1.

For a given account i, consider the aggregated event sequence $\bar{S}_{i\cdot}^r = \cup_{j=1}^m S_{i,j}^r$ for each row of $\mathcal{S}$ and event type r. If we assume a multi-level model for each $S_{i,j}^r$ as in (4.1), conditional on latent variables $\mathcal{L}$, $\bar{S}_{i\cdot}^r$ is a superposition of m independent Poisson processes and hence can be viewed as a new Poisson process with intensity functional $\lambda^r_{i\cdot}(t|\mathcal{L}) = \sum_{j=1}^m \exp \Lambda^r_{i,j}(t)$. We approximate the distribution of $\bar{S}^r_{i\cdot}$ by a Poisson process with a marginal intensity function,

$$\bar{\lambda}^r_i(t) = \mathbb{E}_{YZ}\{\lambda^r_{i\cdot}(t|\mathcal{L})|X_i(t)\} = m \exp\{\tilde{X}^r_i(t)\} \tag{4.14}$$

where $\tilde{\boldsymbol{X}}_i = \{\tilde{X}^1_i, \cdots, \tilde{X}^R_i\}$ is a new multivariate mixture Gaussian process with mean function $\tilde{\mu}^r_{x,c}(t) = \mu^r_{x,c} + \Gamma^{r,r}_y(t,t)/2 + \Gamma^{r,r}_z(t,t)/2$ and covariance function $\tilde{\Gamma}^{r,r'}_{x,c}(s,t) = \Gamma^{r,r'}_{x,c}(s,t)$, if account i belongs to cluster c. When m is large, we expect the above approximation is accurate.

Note that the model in (4.14) for the aggregated event sequence $\bar{S}^r_{i\cdot}$ is inherently reduced to a single-level model. It allows us to separate the inference of the multi-level model in (4.1) into two steps: (*Step I*) learn the parameters in Θ_y and Θ_z and denote the estimated parameters as $\hat{\Gamma}^{r,r'}_y(s,t)$ and $\hat{\Gamma}^{r,r'}_z(s,t)$; (*Step II*) learn the clusters of the single-level model in (4.14) and estimate the parameters $\boldsymbol{\pi}$, $\tilde{\mu}^r_{x,c}$ and $\tilde{\Gamma}^{r,r'}_{x,c}$; Afterwards, the parameters involved in $\Theta_{x,c}$ can be obtained by,

$$\hat{\mu}^r_{x,c}(t) = \tilde{\mu}^r_{x,c}(t) - \hat{\Gamma}^{r,r}_y(t,t)/2 - \hat{\Gamma}^{r,r}_z(t,t)/2, \quad \hat{\Gamma}^{r,r'}_{x,c}(s,t) = \tilde{\Gamma}^{r,r'}_{x,c}(s,t).$$

For the learning task in Step I, Xu et al. (2020) developed a semi-parametric algorithm to learn the repeatedly observed event sequences. In analogy to their work, we propose a similar inference framework to estimate Θ_y and Θ_x in our mixture multi-level model (4.1) and provide the details in Supplementary 4.4.3. For step II, we resort to the single-level model algorithm described in Section 4.4.1.

4.4.3 Step I of the Two-step Learning of the Multi-Level Model

We consider a multi-level model with the following latent intensity function:

$$\lambda^r_{i,j}(t) = \exp\{X^r_i(t) + Y^r_j(t) + Z^r_{i,j}(t)\}, \quad t \in [0,T] \tag{4.15}$$

for $i = 1, \cdots, n$, $j = 1, \cdots, m$ and $r = 1, \cdots, R$.

As discussed in Section 4.4.2, the learning algorithm is decomposed into two steps as in Algorithm 2. In Step I, we seek to estimate the parameters in Θ_y and Θ_z. Other cluster-specific model parameters such as cluster assignment probabilities $\boldsymbol{\pi}$ are estimated in Step II following the procedure described in Section 4.4.2. Xu et al. (2020) developed a semi-parametric algorithm to estimate the covariance functions of a multi-level log-Gaussian Cox process. We extend their estimation method to also take into account unknown clustering when estimating Θ_y and Θ_z in Step I. Interestingly, we will show that the resulting estimators of Θ_y and Θ_z do not depend on any other cluster-specific parameters and hence avoid iterations between the two steps.

Specifically, following the formula of the moment generating function of a Gaussian random variable, the marginal intensity functions can be calculated as

$$\rho^r(t) = \mathbb{E}[\lambda^r_{i,j}(t)] = \sum_{c=1}^{C} \pi_c \exp\{\mu^r_{x,c}(t) + \Gamma^{r,r}_{x,c}(t,t)/2 + \Gamma^{r,r}_{y}(t,t)/2 + \Gamma^{r,r}_{z}(t,t)/2\},$$

and derived in a similar way, the marginal second-order intensity functions are:

$$\begin{aligned}\rho^{r,r'}_{i,j,i',j'}(s,t) &= \mathbb{E}[\lambda^r_{i,j}(s)\lambda^{r'}_{i',j'}(t)]\\ &= \sum_{c}\sum_{c'} \mathbb{E}[\exp\{Y^r_j(s) + Y^{r'}_{j'}(t) + Z^r_{i,j}(s) + Z^{r'}_{i',j'}(t)\}]\\ &\quad \cdot \mathbb{E}[\omega_{c,i}\omega_{c',i'}] \cdot \mathbb{E}[\exp\{X^r_i(s) + X^{r'}_{i'}(t)\}|\omega_{c,i}=1, \omega_{c',i'}=1]\end{aligned}$$

for $i, i' = 1, \cdots, n$, $j, j' = 1, \cdots, m$ and $r, r' = 1, \cdots, R$.

We analyze the form of $\rho^{r,r'}_{i,j,i',j'}$ under four different situations and use $A^{r,r'}$, $B^{r,r'}$, $C^{r,r'}$ or $D^{r,r'}$

to represent its form under each situation respectively,

$$\rho^{r,r'}_{i,j,i',j'}(s,t) = \begin{cases} A^{r,r'}(s,t) \equiv \exp\{\Gamma^{r,r'}_y(s,t) + \Gamma^{r,r'}_z(s,t)\} \sum_c \pi_c \rho^r_c(s) \rho^{r'}_c(t) \exp\{\Gamma^{r,r'}_{x,c}(s,t)\}, & \text{if } i = i', j = j' \\ B^{r,r'}(s,t) \equiv \sum_c \pi_c \rho^r_c(s) \rho^{r'}_c(t) \exp\{\Gamma^{r,r'}_{x,c}(s,t)\}, & \text{if } i = i', j \neq j' \\ C^{r,r'}(s,t) \equiv \exp\{\Gamma^{r,r'}_y(s,t)\} \sum_{c,c'} \pi_c \pi_{c'} \rho^r_c(s) \rho^{r'}_{c'}(t), & \text{if } i \neq i', j = j' \\ D^{r,r'}(s,t) \equiv \sum_{c,c'} \pi_c \pi_{c'} \rho^r_c(s) \rho^{r'}_{c'}(t), & \text{if } i \neq i', j \neq j' \end{cases} \tag{4.16}$$

It can be seen that $A^{r,r'}(s,t)$, $B^{r,r'}(s,t)$, $C^{r,r'}(s,t)$ and $D^{r,r'}(s,t)$ captures different correlation information, namely, the correlation within same-account same-day, within same-account across different-day, within same-day across different-account, and across different-account different-day, while integrating out the unknown cluster memberships of i and i'.

Following a similar derivation as Xu et al. (2020), the corresponding empirical kernel estimate of $\rho^{r,r'}_{i,j,i',j'}$ under each situation is given by

$$\begin{cases} \hat{A}^{r,r'}(s,t;h) = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{u \in S^r_{i,j}, v \in S^{r'}_{i,j}}^{u \neq v} \sum \frac{K_h(s-u)K_h(t-v)}{nmg(s;h)g(t;h)} \\ \hat{B}^{r,r'}(s,t;h) = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{j' \neq j} \sum_{u \in S^r_{i,j}} \sum_{v \in S^{r'}_{i,j'}} \frac{K_h(s-u)K_h(t-v)}{nm(m-1)g(s;h)g(t;h)} \\ \hat{C}^{r,r'}(s,t;h) = \sum_{i=1}^{n} \sum_{i' \neq i} \sum_{j=1}^{m} \sum_{u \in S^r_{i,j}} \sum_{v \in S^{r'}_{i',j}} \frac{K_h(s-u)K_h(t-v)}{n(n-1)mg(s;h)g(t;h)} \\ \hat{D}^{r,r'}(s,t;h) = \sum_{i=1}^{n} \sum_{i' \neq i} \sum_{j=1}^{m} \sum_{j' \neq j} \sum_{u \in S^r_{i,j}} \sum_{v \in S^{r'}_{i',j'}} \frac{K_h(s-u)K_h(t-v)}{n(n-1)m(m-1)g(s;h)g(t;h)} \end{cases} \tag{4.17}$$

for $r, r' = 1, \cdots, R$, where $K_h(t) = h^{-1}K(t/h)$ is a kernel function with bandwidth h and $g(x;h) = \int K_h(x-t)dt$ is an edge correction term.

Matching (4.16) with (4.17), we propose to estimate the covariance functions using,

$$\hat{\Gamma}_y^{r,r'}(s,t;h) = \log \frac{\hat{C}^{r,r'}(s,t;h)}{\hat{D}^{r,r'}(s,t;h)}, \quad \hat{\Gamma}_z^{r,r'}(s,t;h) = \log \frac{\hat{A}^{r,r'}(s,t;h)\hat{D}^{r,r'}(s,t;h)}{\hat{B}^{r,r'}(s,t;h)\hat{C}^{r,r'}(s,t;h)} \tag{4.18}$$

Algorithm 2 Learning of the Multi-level model (4.1)

```
Input: $\mathcal{S} = \{S_{i,j}^r\}$, the number of clusters $C$, the bandwidth $h$;
Output: Estimates of model parameters, $\hat{\boldsymbol{\pi}}$, $\hat{\Theta}_y$, $\hat{\Theta}_z$, $\hat{\Theta}_{x,c}$, for $c = 1, \cdots, C$;
Step I: Given $\mathcal{S}$, obtain $\hat{\Theta}_y$ and $\hat{\Theta}_z$ using the estimation framework in Section 4.4.3;
Step II:
    a) Aggregate the event sequences by $\bar{S}_{i\cdot}^r = \cup_{j=1}^m S_{i,j}^r$;
    b) Based on $\{\bar{S}_{i\cdot}^r\}_{i=1}^n$ from a), fit the single-level model with parameters $\{\boldsymbol{\pi}, \tilde{\Theta}_{x,c}\}$ using Algorithm 1;
    c) Calculate,
        $\hat{\mu}_{x,c}^r(t) = \tilde{\mu}_{x,c}^r(t) - \hat{\Gamma}_y^{r,r}(t,t)/2 - \hat{\Gamma}_z^{r,r}(t,t)/2 - \log m, \quad \hat{\Gamma}_{x,c}^{r,r'}(s,t) = \tilde{\Gamma}_{x,c}^{r,r'}(s,t)$
```

4.4.4 Computational Complexity and Acceleration

Assume that the training event sequences belong to n accounts and C clusters and are repeatedly observed on m time slots. We also assume that the data contains R types of events and each sequence consists of I time stamps on average. Let Q be the sampling size used in the Monte Carlo integration in (4.9). In numerical implementation, we divide the interval $[0, T]$ into D equally spaced grid points $\mathcal{D} = \{0 = u_1 < \cdots < u_D = T\}$. In Step I, it requires $O(nmR^2D^2)$ computation complexity to estimate Θ_y and Θ_z, according to Xu et al. (2020). Computation complexity to pre-calculate $a_i^{r,r'}(s,t;h)$'s and $b_i^r(t;h)$'s in (4.7) for all $s,t \in \mathcal{D}$ is of the order $O(nmR^2D^2)$ if we decomposition $a_i^{r,r'}(s,t;h)$ as:

$$a_i^{r,r'}(s,t;h) = \left[\sum_{u \in S_i^r} \frac{K_h(s-u)}{g(s;h)}\right] \left[\sum_{v \in S_i^{r'}} \frac{K_h(t-v)}{g(t;h)}\right] - \sum_{u \in S_i^r \cap S_i^{r'}} \frac{K_h(s-u)K_h(t-v)}{g(s;h)g(t;h)}.$$

In Step II, for each E-S iteration, we need $O(CQR^3)$ for sampling and $O(nCIQR^2)$ for other calculations. Therefore, the overall computational complexity is $O(R^2(nmD^2 + CQR + nCIQ))$. To further reduce computation, we use array programming and GPU acceleration to calculate the high-dimensional integration in the Monte Carlo EM framework (Wu et al., 2020b) to reduce the runtime of (4.8). The details are included in Supplementary S.2, and a numerical demonstration is given in Section 4.5.1.

4.5 Numerical Examples

We examine the performance of our **MM-MPP** framework for clustering event sequences via synthetic data examples and real-world applications and compare the performances between the proposed method and two other state-of-the-art methods. One competing method is a discrete Fréchet distance-based method (**DF**) by Pei et al. (2013). Unlike other distance-based clustering methods, the DF cluster can characterize interactions among events. Another is a model-based clustering method based on the Dirichlet mixture of Hawkes processes (**DMHP**) by Xu and Zha (2017). DMHP is chosen as a competitor due to its capability of accounting for complex point patterns while performing clustering and making efficient variational Bayesian inference algorithms under a nested EM framework.

We first defined some abbreviations

- **ES**: Expectation-Solution;
- **LGCP**: log-Gaussian Cox process;
- **FPCA**: Functional principal component analysis;

- **MM-MPP**: Mixture Multi-level Marked Point Processes;
- **MS-MPP**: Mixture Single-level Marked Point Processes;
- **MC**: Monte Carlo
- **DMHP**: Dirichlet mixture of Hawkes processes;
- **DF**: discrete Fréchet;

4.5.1 Synthetic Data

Setting. We generate the synthetic data from the proposed mixture model of log-Gaussian Cox processes in (4.1) and (4.2), in which there are $R = 5$ event types and daily time stamps reside in $[0, 2]$. We set the number of clusters C from 2 to 5 and set the number of accounts in each cluster to 500. We experiment with an increasing number of replicates ($m = 1$, 20 or 100), to check the convergence of our method. When $m = 1$, we generate event sequences from the single-level model in (4.2) without day-level variations. In this case, we compare the clustering results of DF, DMHP with those of the single-level model (MS-MPP). When $m = 20$ or 100, we generate data from the multi-level model in (4.1) and use the MM-MPP method to model the scenario where event sequences are repeatedly observed. However, the two competing methods, DF and DMHP, are not directly applicable for repeated event sequences. Therefore, in this case, we concatenate $\{S^r_{i,j}\}_{j=1}^m$ sequentially into a new event sequence $S^r_{i\cdot}$ on $[0, mT]$ and then apply DF and DMGP to this new sequence. The detailed settings of $X^r_i(t)$'s, $Y^r_j(t)$'s and $Z^r_{i,j}(t)$'s and other details of synthetic data examples are elaborated in Supplementary S.3.

Results. We evaluate the clustering performance of each method over 100 repeated experiments under each setting, using *clustering purity* (Schütze et al., 2008) as a evaluation metric. Table 4.1 reports the averaged clustering purity of each method on the synthetic data. When $m = 1$, MS-MPP obtains the best clustering result in terms of purity consistently across different numbers of clusters. Especially when C increases, in which case there are more overlaps among clusters, the advantage of MS-MPP becomes more prominent. When $m = 20, 100$, MM-MPP still significantly outperforms the other two competitors. It is also noticeable that the performance of DF and DMPH, in general, deteriorates as m increases, although more repeated event sequences offer more information for clustering. One explanation is that both DF and DMHP may incur bias due to ignoring different sources of variations for repeatedly observed event times. Another reason may be that many existing Hawkes process models, such as DMHP, assume a constant triggering function over time, which may not be flexible enough to characterize the data generated from models (4.1) and (4.2).

Table 4.1: Clustering Purity on Synthetic Data.

	$m = 1$			$m = 20$			$m = 100$		
C	DF	DMPH	MS-MPP	DF	DMPH	MM-MPP	DF	DMPH	MM-MPP
2	0.597	0.537	**0.831**	0.536	0.513	**0.947**	0.532	0.522	**0.988**
3	0.514	0.466	**0.767**	0.465	0.423	**0.902**	0.477	0.394	**0.967**
4	0.443	0.421	**0.714**	0.422	0.356	**0.874**	0.436	0.285	**0.944**
5	0.379	0.354	**0.675**	0.351	0.298	**0.835**	0.333	0.276	**0.919**

Our code can be accessed via `https://github.com/LihaoYin/MMMPP`. To show the computational advantage of the proposed ES algorithm over the EM algorithm, Table 4.2 gives the computation times of CPU-based EM, CPU-based ES, and GPU-based ES algorithms for 20 iterations in the estimation of model (4.1) with $n = 500, 100$, $m = 20$, $R = 5$ and $C = 3$. For each iteration, $10,000$ MCMC samples are drawn to approximate (4.9). Table 4.2 demonstrates that with the GPU acceleration, the computation time of the proposed ES can be reduced by more than 20 folds in this case scenario compared to the EM algorithm, which is not suitable for array programming (Harris et al., 2020).

4.5.2 Real-world Data

In this section, we apply our method to the following real-world datasets.

Twitter Dataset. The Twitter dataset consists of the postings of the official accounts of America's top 500 universities from April 15, 2021, to May 14, 2021. The data set was scraped from

Table 4.2: Running Time (in seconds) on Synthetic Data

Methods and devices	$n = 500$	$n = 1000$
GPU-ES (RTX 8000 48G GPU)	**30.09**	**51.42**
CPU-**ES** (i7-7700HQ CPU)	275.87	505.07
CPU-EM (i7-7700HQ CPU)	568.36	1105.46

Twitter with the API `rtweet` (Kearney, 2019). The dataset involves three categories of postings (tweet, retweet, and reply), indicating $R = 3$ in this study. As a result, the dataset contains $n = 500$ Twitter accounts for $m = 30$ consecutive days with a total of $233,465$ time stamps.

Chicago City Taxi Dataset The City of Chicago collected the information of all taxi rides in Chicago since 2013 [1]. Each trip record in the dataset consists of drivers' encrypted IDs, pick-up/drop-off time stamps, and locations (in the form of latitude/longitude coordinates). We gathered the trips of 9,000 randomly selected taxi drivers from Jan 1 to Dec 31, 2016, and more than 19 million trip records were picked. We mapped the pick-up coordinates to their corresponding zoning types according to Chicago Zoning Map Dataset[2], which divides the city into nine basic zoning districts[3], as shown in Figure 4.1, including Residence (R), Business (B), Commercial (C), Manufacturing (M), etc. For this data set, we have $n = 9000$, $m = 366$, and $R = 9$.

Credit Card Transaction Dataset. The dataset contains $641,914$ transaction records of $5,000$ European credit card customers ($n = 5000$) during the period covering January 1 to December 31, 2016 ($m = 366$). We applied the univariate model ($R = 1$) without event marks to the dataset.

We evaluate and compare clustering stability based on a measure called *clustering consistency* via K-trial cross-validations (Tibshirani and Walther, 2005; von Luxburg, 2009), as there are no ground truth clustering labels. The detailed definition of *clustering consistency* and other real data example details are included in Supplementary S.4.

Results. We compare the performance of DF, DMHP, and MM-MPP in terms of clustering consistency for three data sets with $K = 100$ trials. The results in Table 4.3 suggest that MM-

[1] `https://data.cityofchicago.org/Transportation/Taxi-Trips/wrvz-psew`

[2] `https://data.cityofchicago.org/`

[3] `https://secondcityzoning.org/zones/`

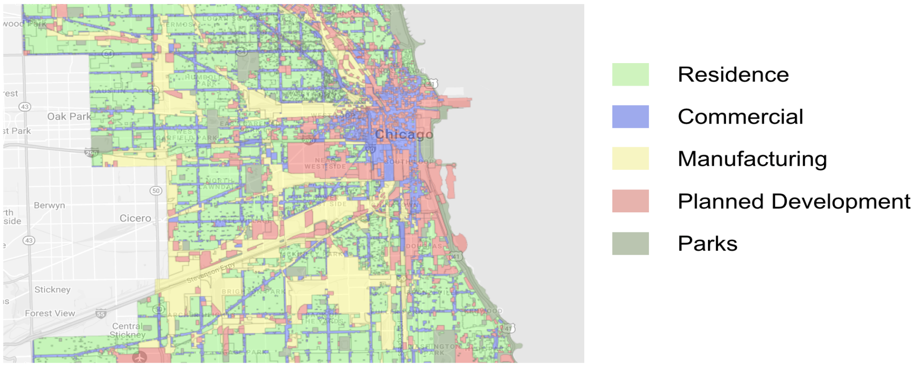

Figure 4.1: Illustration for basic zoning districts in Chicago

MPP outperforms its competitors notably, demonstrating that our model can better characterize the postings patterns and offer a more stable and consistent clustering than other methods. Figure 4.2 shows the histograms of the number of learned clusters for each method. For the Twitter dataset, the median numbers of learned clusters are 3, 5, and 8 for MM-MPP, DMHP, and DF respectively. Besides, the distribution of the number of clusters from our method seems to be the least variable, indicating robustness in clustering. The robustness of our method may be partly attributed to the flexibility of the latent conditional intensity functions that account for multi-level deviations within each account. In contrast, other methods that fail to account for different sources of deviations may treat them as sources of heterogeneity and consequently result in more clusters.

Table 4.3: Clustering Consistency on Real-World Datasets.

Method	DF	DMHP	MM-MPP
Twitter	0.096	0.275	**0.394**
Credit Card	0.102	0.331	**0.378**
Chicago Taxi	0.045	0.142	**0.153**

More stories can be told by the estimated posting patterns. Given a predicted membership of account i by $c_i = \arg\max_c \mathbb{E}_\omega[\omega_{c,i}|\mathcal{S};\hat{\Omega}]$, Figure 4.3 displays the estimated curves of $\hat{\mu}^r_{x,c}$ for tweet

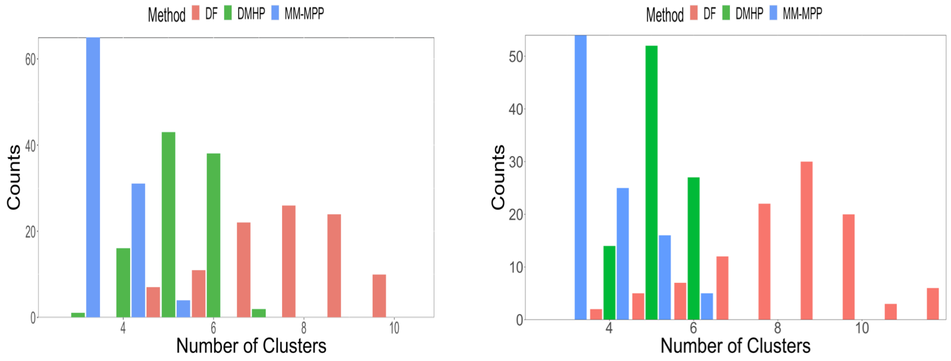

Figure 4.2: Histogram of the number of clusters. Left: Twitter dataset; Right: Credit Card dataset;

events ($r = 1$), retweet events ($r = 2$) and reply events ($r = 3$) respectively for $C = 3$. Recall $\hat{\mu}_{x,c}^{r}$ is interpreted as the baseline of intensity functions. This figure shows three different activity modes for the selected Twitter accounts. The universities in cluster 1 marked by red curves in Figure 4.3 in general have a lower frequency of posting retweets and replies, especially during the daytime. This group includes the most top university in America, such as MIT, Harvard, and Stanford. In contrast, the accounts in cluster 2 are relatively more active in all three types of postings. We further find that many accounts in this cluster belong to the universities with middle ranks.

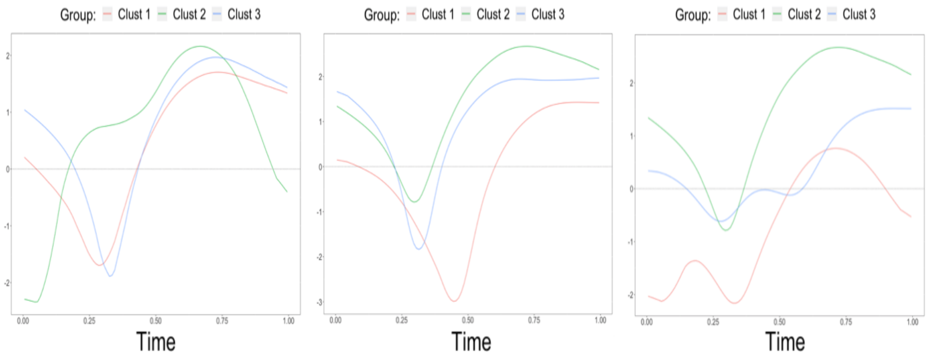

Figure 4.3: Curves of $\hat{\mu}_{x,c}^{r}(t)$. Left: tweet events; Mid: retweet events; Right: reply events;

We further applied the proposed MM-MPP to the Chicago Taxi dataset. As suggested by BIC, the 9000 taxi drivers are clustered into 9 groups, whose averaged daily pick-up log intensity functions are illustrated in Figure 4.4(a). We can see that the taxi drivers are clustered not only according to their pick-up frequency but also by their working schedules. For example, the black curves on Figure 4.4(a) corresponds to the most dominating group, which occupies 23.2% of the sample. Figure 4.3(b) displays the curves of average log intensity (the black line) and log intensity for each driver (gray lines) in the selected cluster. Figure 4.4(c-e) show the estimated $\hat{\mu}_x(t)$ for pick-up in commercial, residence and manufacturing districts, respectively. While the pick-up events are more likely to occur in commercial districts for this group during the daytime, they also tend to pick up passengers at the residential district in the morning and to appear at the manufacturing district in the afternoon. These patterns are consistent with the schedules of passengers who commute between homes and workplaces.

More results and discussions on chase credit dataset are included in our Supplementary file.

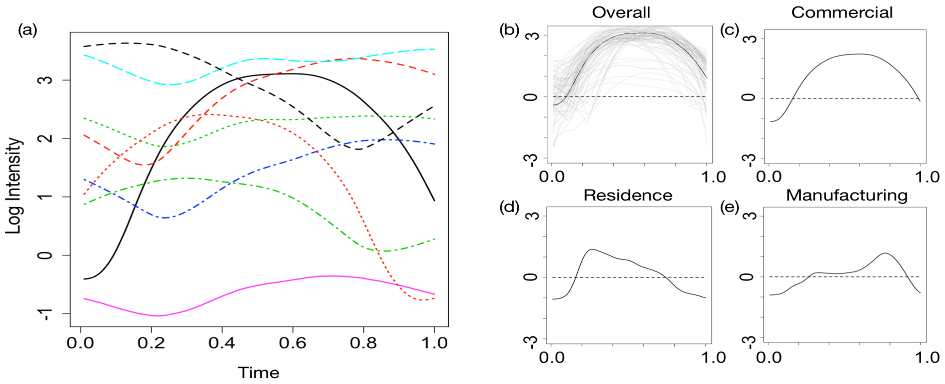

Figure 4.4: Left: Overall log-intensities for all clusters; Right: Log-intensity for one selected cluster;

4.6 Conclusions

We propose a mixture of multi-level marked point processes to cluster repeatedly observed marked event sequences. A novel and efficient learning algorithm is developed based on a semi-parametric ES algorithm. The proposed method is demonstrated to significantly outperform other competing methods in simulation experiments and real data analyses.

The current model only focuses on events over temporal domains. However, clustering of spatial patterns on 2- or 3-dimensional domains has also attracted much research interest (Hildeman et al., 2018; Yin et al., 2020; Hessellund et al., 2021). It will be an interesting research topic to extend the current model to such settings.

This work has no foreseeable negative societal impacts, but users should be cautious when giving interpretation on clustering results to avoid any misleading conclusions.

APPENDIX A

SUPPLEMENTARY FOR FUSED SPATIAL POINT PROCESS INTENSITY ESTIMATION WITH VARYING COEFFICIENTS ON COMPLEX CONSTRAINED DOMAINS

A.1 Quadrat Scheme

For the quadrature approximation in (2.4), we need to divide the domain into small subdivisions or quadrats. For the 2-D square domain, it is easy to add dummy points and draw regular equal-sized rectangular quadrats on the domain using `dummy.ppm` and `quad.ppm` in the `R` package `spatstat`. For planar domains with irregular boundary, we follow the routine quadrature approximation methods for irregular domains in $\mathbb{R}^2$ (Shen et al., 2009), where the domain is masked by regular pixel grids, and we expand the domain slightly to include the whole pixel if it intersects with domain boundary. See the left panel of Figure A.1 for an example.

As for the partition on a linear work, Chapter 9 of Okabe and Sugihara (2012) discussed the implementation of both equal-length and unequal-length network cells. Furuta et al. (2008); Shiode (2008) proposed computational methods for dividing a network into equal-length network cells. However, their methods are not a guarantee of success if we want to insert enough dummy points. We propose to randomly draw dummy points from a homogeneous Poisson point process on the linear network with intensity function $\delta(u) = (M - m)/|D|$. Then we obtain the quadrats on linear networks using the network Voronoi tessellation method (Chapter 4 of Okabe and Sugihara, 2012), each of which contains one dummy point as its centroid. See the right panel of Figure A.1 for an example.

A.2 Additional Numerical Results

We report the integrated squared bias (ISB) and variance (IV) for each $\hat{\beta}_1(u)$, $\hat{\beta}_2(u)$ and $\hat{\beta}_0(u)$ under both Scenario 1 and Scenario 2(b) in Table A.1. Specifically, ISB and IV are defined as: ISB $(\beta_k(u)) = \frac{1}{|D|}\int_D \left(\hat{\beta}_k(u) - \mathbb{E}\hat{\beta}_k(u)\right)^2 du$; IV $(\beta_k(u)) =$ MISE $(\beta_k(u)) -$ ISB $(\beta_k(u))$. The results generally agree well with the findings based on Rand index and the MISE$_\beta$, in the sense

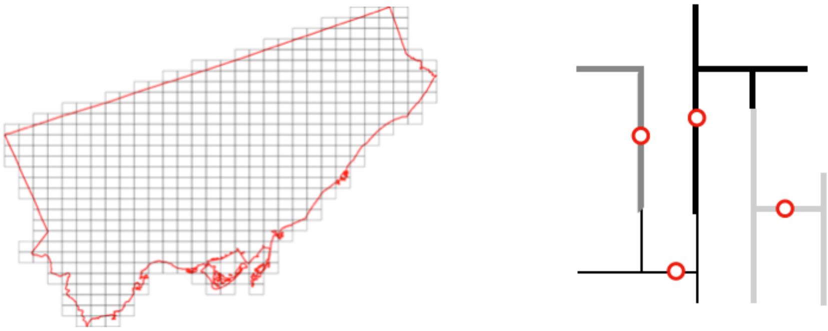

Figure A.1: Illustration of quadrat schemes for Toronto city (left) and a toy linear network example (right); Left panel: boundary lines (red) of Toronto city and the grids (grey) covering the irregular domain; Right panel: dummy points (red nodes) and their corresponding Voronoi quadrats (marked by different shades of grey).

that SVCI-LRL achieves a slightly smaller bias and variance compared with SVCI-PL.

Table A.1: Scenario 1: The integrated squared bias and variance (in parentheses) of $\hat{\beta}_1(u)$, $\hat{\beta}_2(u)$ and $\hat{\beta}_0(u)$ respectively, in the case that $m = 2400$, and $\texttt{nd}^2 = m$ based on 5-NN connection graphs.

	$\beta_1(u)$	$\beta_2(u)$	$\beta_0(u)$
Scenario 1			
SVCI-PL	0.115(0.035)	0.116(0.032)	0.078(0.038)
SVCI-LRL	0.098(0.034)	0.105(0.033)	0.057(0.030)
Scenario 2(b)			
SVCI-PL	0.099(0.028)	0.104(0.030)	0.108(0.035)
SVCI-LRL	0.105(0.028)	0.101(0.027)	0.093(0.030)

Under the setting of Scenario 1, we examine the performance of SVCI-PL based on K-NN graphs with $K = 3, 4, 5, 6, 9$ and r-NN graphs with $r = 0.02R$ or $0.03R$. Figure A.2 shows the MISE of $\hat{\boldsymbol{\beta}}$ versus the number of edges for each graph. We notice that MISE does not always decrease as the graph includes more neighbors by increasing K of K-NN or r of r-NN. Both K-

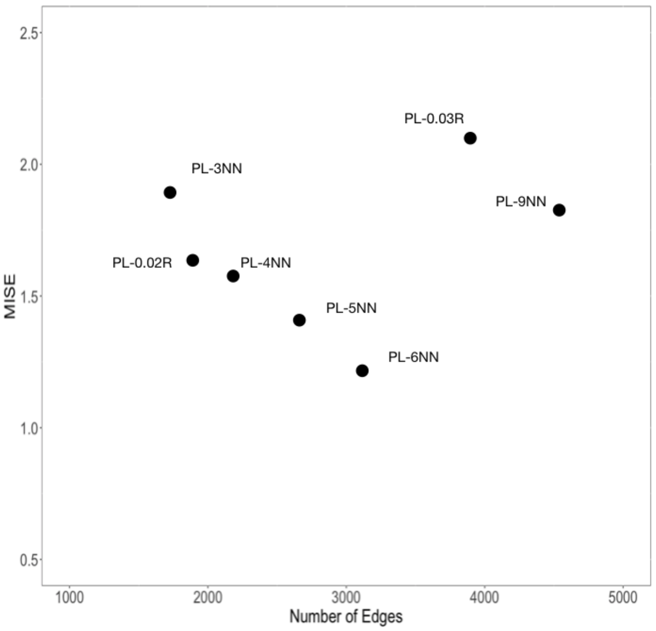

Figure A.2: Illustration of MISE$_\beta$ versus the number of edges, under the setting of Scenario 1, with $m = 2400$ and $\mathtt{nd}^2 = m$. ϵR indicates the SVCI-PL method based on an r-NN graph with a radius $\epsilon \times R$.

NN and r-NN lose some estimation accuracy if there are too many neighbors. It also seems that K-NN slightly outperforms r-NN in terms of MISE when K and r are chosen such that the two graphs have a comparable number of edges.

Under the setting of Scenario 2(a), we compare the performance of SVCI-LRL using the 3-NN graphs constructed based on the shortest-path distance and Euclidean distance metrics, respectively. Overall, we observe very similar results between these two choices of distance metrics, especially when m and $\mathtt{nd}^2$ go up. For example, when $m = 2400$, the MISEs of SVCI-LRL using shortest-path distance and Euclidean distance metrics are 0.121 and 0.124, respectively, which are very close to the result reported in Table 2.4 obtained using the graph constructed by connecting natural neighbors on the Chicago network.

Table A2 reports the computation time of different methods with one tuning parameter, under the setting of Scenario 2(a). We notice that the computation time of KDE.lpp and Voronoi.lpp vary notably with their tuning parameter, so for these two methods we select a range of tuning parameters in the proximity of the optimal tuning parameter that minimizes MISE and report the average computation times.

Table A.2: Scenario 2: Comparison of computation times (in seconds).

Method	$m = 800$	$m = 1600$	$m = 2400$	$m = 3600$	$m = 6000$
		Scenario 2(a): $\rho(u) = \exp\{\beta_0(u)\}$			
SVCI-LRL	0.64	0.73	1.11	1.40	1.92
KDE.lpp	4.76	3.74	3.15	2.65	2.47
KDEQuick.lpp	0.055	0.062	0.081	0.094	0.102
Voronoi.lpp	3.24	3.73	4.35	4.52	4.83

For the Toronto Homicide data considered in Section 2.5.1, we compare the estimated log intensity surfaces obtained by SVCI and LGCP in Figure A.3. Both methods seem to be capable of capturing the inhomogeneity pattern in intensities and agree well with each other in most areas. The most notable difference between the two methods occurs near Toronto islands, where we observe more variations in intensity estimations by LGCP than those by SVCI possibly due to the piece-wise homogeneity assumptions made in the latter.

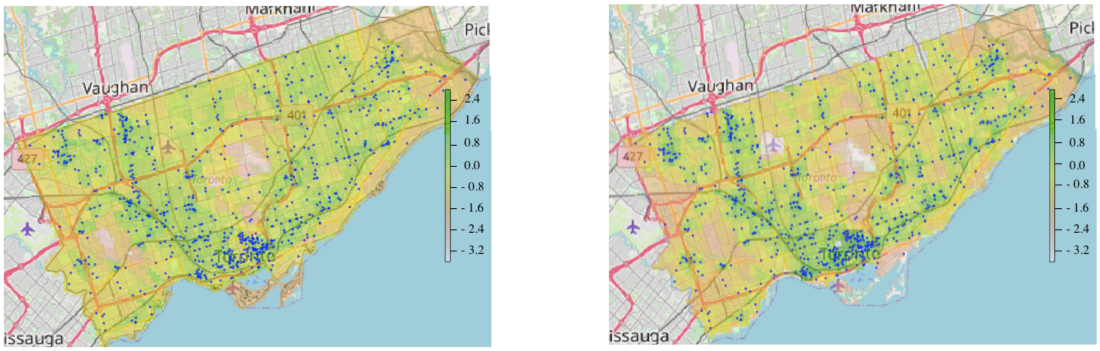

Figure A.3: Illustrate of the estimated log intensity function of the Toronto data with the observed locations overlaid in blue dots; Left: the estimated log intensity surface by SVCI; Right: the posterior mean estimate of the log intensity surface by LGCP;

APPENDIX B

SUPPLEMENTARY FOR ROW-CLUSTERING OF A POINT PROCESS-VALUED MATRIX

B.1 Computational Details

B.1.1 ES Algorithm

The Expectation-Solution (ES) algorithm (Elashoff and Ryan, 2004; McLachlan and Krishnan, 2007) is a general extension of the Expectation-Maximization (EM) algorithm. It is an iterative approach built upon estimating equations that involve missing data or unobserved variables. In the E-step of each iteration, ES calculates the conditional expectations of estimating equations given observed data and current parameter estimates. In S-step, it updates parameter values by finding the solutions to the expected estimating equations. Since the estimating equations can be constructed from a likelihood, a quasi-likelihood, or other forms, the ES algorithm is more flexible and general than the EM algorithm. In particular, when estimating equations are well designed such that analytical solutions are available in S-step, ES algorithm may achieve an improved computational efficiency over EM algorithms, which often involve expensive numerical optimizations of the expected log-likelihood in each M-step.

We follow the notations and expressions in Elashoff and Ryan (2004). Let $\boldsymbol{y}$ denote the observed data vector, $\boldsymbol{z}$ denote the unobserved data, and $\boldsymbol{x} = \{\boldsymbol{y}, \boldsymbol{z}\}$ be the complete-data. Let $\boldsymbol{\Omega}$ denote a d-dimensional vector of parameters. Given d-dimensional estimating equations with the complete data as:

$$U_c(\boldsymbol{x}; \boldsymbol{\Omega}) = \boldsymbol{0}$$

the ES algorithm entails a linear decomposition like:

$$\begin{aligned} U_c(\boldsymbol{x}; \boldsymbol{\Omega}) &= U_1(\boldsymbol{y}, \boldsymbol{S}(\boldsymbol{x}); \boldsymbol{\Omega}) \\ &= \sum_{j=1}^{q} \boldsymbol{a}_j(\boldsymbol{\Omega}) S_j(\boldsymbol{x}) + \boldsymbol{b}_{\Omega}(\boldsymbol{y}), \end{aligned} \tag{B.1}$$

where $\boldsymbol{a}_j$'s are vectors of size d only depending on parameters $\boldsymbol{\Omega}$, and $\boldsymbol{S}$ is a q-dimensional function with components S_j only depending on the complete data. $\boldsymbol{S}(\boldsymbol{x})$ is referred to as a "complete-data summary statistic". Given the parameters $\boldsymbol{\Omega}^*$, we calculate the expectation over $\boldsymbol{z}$ condition on $\boldsymbol{y}$ and parameters $\boldsymbol{\Omega}$ in E-step as following

$$h(\boldsymbol{y};\boldsymbol{\Omega}^*) = \mathbb{E}_z[\boldsymbol{S}(\boldsymbol{x})|y;\boldsymbol{\Omega}^*]$$

In view of the linearity in (B.1), we consider the conditionally expected estimation equations,

$$\mathbb{E}_z[U_c(\boldsymbol{x};\boldsymbol{\Omega})|\boldsymbol{y};\boldsymbol{\Omega}^*] = U_1(\boldsymbol{y}, h(\boldsymbol{y};\boldsymbol{\Omega}^*);\boldsymbol{\Omega}) = \boldsymbol{0} \tag{B.2}$$

In the S-step, we update the parameters $\boldsymbol{\Omega}$ by finding the solution to (B.2). We outline the ES procedure in Algorithm 3.

Algorithm 3 ES Algorithm

```
Presupposition: Given estimating equations $U_c(\boldsymbol{x};\boldsymbol{\Omega})$ with a linear decomposition (B.1);
Input: Observed data $\boldsymbol{y}$;
Output: Estimates of model parameters $\boldsymbol{\Omega}$;
Initialize $\boldsymbol{\Omega}^*$ randomly;
Repeat:
    E-Step:    Calculate $h(\boldsymbol{y};\boldsymbol{\Omega}^*) = \mathbb{E}_z[\boldsymbol{S}(\boldsymbol{x})|y;\boldsymbol{\Omega}^*]$;
    S-Step:    Find $\boldsymbol{\Omega}$ that solve $U_1(\boldsymbol{y}, h(\boldsymbol{y};\boldsymbol{\Omega}^*);\boldsymbol{\Omega}) = \boldsymbol{0}$ in (B.2);
    End;
Until: Reach the convergence criteria.
```

B.1.2 Sampling Strategy

The E-step in Section 4.4.1 involves the sampling of random functions $\boldsymbol{X}_c^{(q)}$ for calculating the Monte Carlo integration in (4.9). Given cluster-specific parameters $\Omega_{x,c}$, our goal is to draw multiple independent realizations of $\boldsymbol{X}_i(t)|\omega_{c,i} = 1$, denoted as $\boldsymbol{X}_c^{(q)}(t) = \{X_c^{1(q)}(t), \cdots, X_c^{R(q)}(t)\}'$. Recall that the cross covariance functions of $\boldsymbol{X}_i(t)$ is $\Gamma_{x,c}^{r,r'}(s,t) = \text{Cov}[X_i^r(s), X_i^{r'}(t)|\omega_{c,i} = 1]$,

$s, t \in [0, T]$, for $i = 1, \cdots, n$. When $r = r'$, the covariance function $\Gamma_{x,c}^{r,r}(s,t)$ is a symmetric, continuous and nonnegative definite kernel function on $[0,T] \times [0,T]$. Then Mercer's theorem asserts that there exists the following spectral decomposition:

$$\Gamma_{x,c}^{r,r}(s,t) = \sum_{k=1}^{\infty} \eta_{x,c,k}^{r} \phi_{x,c,k}^{r}(s) \phi_{x,c,k}^{r}(t),$$

where $\eta_{x,c,1}^{r} \geq \eta_{x,c,2}^{r} \geq \cdots > 0$ are eigenvalues of $\Gamma_{x,c}^{r,r}(s,t)$ and $\phi_{x,c,k}^{r}(t)$'s are the corresponding eigenfunctions which are pairwise orthogonal in $L^2([0,T])$. The eigenvalues and eigenfunctions satisfy the integral eigenvalue equation,

$$\eta_{x,c,k}^{r} \phi_{x,c,k}^{r}(s) = \int_0^T \Gamma_{x,c}^{r,r}(s,t) \phi_{x,c,k}^{r}(t) dt$$

Accordingly, using the Karhunen-Loève expansion (Watanabe, 1965), $X_c^{r(q)}(t)$ admits a decomposition,

$$X_c^r(t) = \mu_{x,c}^r(t) + \sum_{k=1}^{\infty} \xi_{x,c,k}^{r} \phi_{x,c,k}^{r}(t), \tag{B.3}$$

where $\{\xi_{x,c,k}^{r}\}_{k=1}^{\infty}$ are independent normal random variables with mean 0 and variance $\{\eta_{x,c,k}^{r}\}_{k=1}^{\infty}$. The expression in (B.3) has an infinite dimensional parameter space, which is infeasible for estimation. One solution is to approximate (B.3) by only keeping leading principal components,

$$X_c^r(t) \approx \mu_{x,c}^r(t) + \sum_{k=1}^{p_c^r} \xi_{x,c,k}^{r} \phi_{x,c,k}^{r}(t) \tag{B.4}$$

where p_c^r is a rank chosen to characterize the dominant characteristics of X_c^r while reducing computational complexity. It leads to a reduced-rank representation of $\Gamma_{x,c}^{r,r}(s,t)$ as:

$$\Gamma_{x,c}^{r,r}(s,t) \approx \sum_{k=1}^{p_c^r} \eta_{x,c,k}^{r} \phi_{x,c,k}^{r}(s) \phi_{x,c,k}^{r}(t).$$

Similarly, when $r \neq r'$, we can also approximate $\Gamma_{x,c}^{r,r'}(s,t)$ using the truncated decomposition,

$$\Gamma_{x,c}^{r,r'}(s,t) \approx \sum_{k=1}^{p_c^r} \sum_{k'=1}^{p_c^{r'}} \eta_{x,c,k,k'}^{r,r'} \phi_{x,c,k}^{r}(s) \phi_{x,c,k}^{r'}(t). \tag{B.5}$$

We denote $\boldsymbol{\xi}_{x,c}^r = \{\xi_{x,c,1}^r, \cdots, \xi_{x,c,p_c^r}^r\}'$ and investigate the cross-covariance matrix of $\boldsymbol{\xi}_{x,c} = \{\boldsymbol{\xi}_{x,c}^1, \cdots, \boldsymbol{\xi}_{x,c}^R\}$, denoted as

$$\boldsymbol{\Sigma}_{x,c} = \begin{pmatrix} \Sigma_{x,c}^{1,1} & \cdots & \Sigma_{x,c}^{1,R} \\ \vdots & \ddots & \vdots \\ \Sigma_{x,c}^{R,1} & \cdots & \Sigma_{x,c}^{R,R} \end{pmatrix},$$

where $\Sigma_{x,c}^{r,r'} = \text{Cov}[\boldsymbol{\xi}_{x,c}^r, \boldsymbol{\xi}_{x,c}^{r'}]$.

From Karhunen-Loève expansion in (B.3), we know $\Sigma_{x,c}^{r,r} = \text{diag}(\eta_{x,c,1}^r, \cdots, \eta_{x,c,p_c^r}^r)$ for each event type r. When $c \neq c'$, we assume that $\boldsymbol{\xi}_{x,c}^r$ and $\boldsymbol{\xi}_{x,c'}^{r'}$ are independent. However, when considering two different event types (i.e., $r \neq r'$) within the same cluster, it is reasonable to account for the correlation between $\boldsymbol{\xi}_{x,c}^r$ and $\boldsymbol{\xi}_{x,c}^{r'}$ to characterize interactions among events of different types. Therefore, from (B.3) and (B.5), the (k, k')-th entry of the covariance matrix $\Sigma_{x,c}^{r,r'}$ is $\eta_{x,c,k,k'}^{r,r'}$ when $r \neq r'$.

Now we can draw the samples $\boldsymbol{\xi}_{x,c}^{(q)} = \{\boldsymbol{\xi}_{x,c}^{1(q)}, \cdots, \boldsymbol{\xi}_{x,c}^{R(q)}\}$ from the multivariate normal distribution with a mean zero and a covariance matrix is $\Sigma_{x,c}$, based on which we obtain the samples $\boldsymbol{X}_c^{(q)}$ using expansion (B.4).

B.1.3 GPU Acceleration

One computational bottleneck in our approach is the Monte Carlo (MC) approximation of the high-dimensional integration in (4.9). Although we have employed the low-rank representations by FPCA in Section B.1.2 to facilitate MC sampling, this step remains as the most computationally expensive part if using a naive direct calculation, due to the massive number of sampling points for a precise MC integration.

Many researchers have embarked their efforts on improving the performance of MC integration. One of the most popular frameworks is VEGAS (Lepage, 1978; Ohl, 1999) due to its user-friendly interface. However, VEGAS, which is CPU-based, may be over-stretched with dimensionality going up since the required MC samples consequently increase dramatically. As notable progress, GPU-based programs, like `VegasFlow`(Carrazza and Cruz-Martinez, 2020), extremely boosts the computation speed compared to the CPU-version program. It accelerates the computation with the `Numpy`-like API syntax, such as `Tensorflow`, which is easy to communicate to GPU. Similar treatments are implemented in our work, and the key is to transfer the summation loop in (4.10) into the form of array programming (Harris et al., 2020). For example, when we calculate $\boldsymbol{X}_c^q(t)$ in (B.4), the computation involves total $p_c^r \times Q$ sampled $\xi_{x,c,k}^r$ and $p_c^r \times I \times n$ of $\phi_{x,c,k}^r(u)$, if given c and r. It will greatly reduce the running time if we utilize array programming. For example, in the case when we have $n = 500$ sequences and $10,000$ MC points, our MS-MPP algorithm costs on average 30.09 seconds to run 20 ES iterations on RTX-8000 48G GPU. In contrast, it costs 275.87 seconds on i7-7700HQ CPU if not using array programming.

B.2 Additional Simulation Studies

Setting of $X_i^r(\cdot)$'s. In our synthetic data, we sample event sequences from C heterogeneous clusters ($C = 2, 3, 4$ or 5). Each cluster contains 500 event sequences, and each event sequence contains $R = 5$ event types. We experiment with each setting for $J = 100$ times and investigate the average performance. In each trial, we set,

$$\mu_{x,c}^r(t) = 1 + \sum_{k=0}^{50} \zeta_k Z_{c,k}^r \cos(k\pi t) + \sum_{k=0}^{50} \zeta_k Z_{c,k}^{\prime r} \sin(k\pi t), \quad t \in [0, 2]$$

for $r = 1, \cdots, R$ and $c = 1, \cdots, C$, where $Z_{c,k}^r$'s and $Z_{c,k}^{\prime r}$'s are all independently sampled from the uniform distribution $\mathrm{U}(-1, 1)$ and $\zeta_k = (-1)^{k+1}(k+1)^{-2}$. We set the covariance function of $X_i^r(t)$ as,

$$\Gamma_{x,c}^{r,r}(s,t) = \sum_{k=1}^{50} \tilde{Z}_{c,k}^r |\zeta_k| \sin(k\pi s + \pi \tilde{Z}_{c,k}^r) \sin(k\pi t + \pi \tilde{Z}_{c,k}^r)$$

for $r = 1, \cdots, R$ and $c = 1 \cdots, C$, where $\tilde{Z}_{c,k}^r$'s are independently sampled from uniform distribution $\mathrm{U}(0, 0.3)$. Meanwhile, we set the interventions among different event types as,

$$\Gamma^{r,r'}_{x,c}(s,t) = \sum_{k=1}^{50}\sum_{k'=1}^{50} \check{Z}^{r,r'}_{c,k,k'} \sqrt{\tilde{Z}^{[j]}_{c,k}\tilde{Z}^{[j']}_{c,k'}|\zeta_k\zeta_{k'}|}\sin(k\pi s + \pi\tilde{Z}^{r}_{c,k})\sin(k\pi t + \pi\tilde{Z}^{r}_{c,k}),$$

for $r \neq r'$, where $\check{Z}^r_{c,k}$'s are independently sampled from uniform distribution $\mathrm{U}(-1, 1)$. The latent variable $X^r_i(t)$'s are generated from Gaussian processes on $[0, 2]$ with the parameters above.

Setting of $Y^r_j(\cdot)$'s and $Z^r_{ij}(\cdot)$'s. Furthermore, we generate event sequences for m ($m = 1$, 20 or 100) days. When $m = 1$, the event sequences are generated from the single-level model in (4.2), which didn't involve the variation $Y_j(t)$ and $Z_{i,j}(t)$. When $m = 20$ or 100, we incorporate $Y_j(t)$ and $Z_{i,j}(t)$ in the intensity function and generate data with the multi-level model in (4.1).

We further describe the setup of the distributions of $Y^r_j(t)$'s and $Z^r_{i,j}(t)$'s. We let,

$$\tilde{Y}^r_j(t) = \sum_{k=1}^{2} \xi^Y_{r,j,k}\phi^Y_k(t), \quad Z^r_{i,j}(t) = \sum_{k=1}^{4} \xi^Z_{r,i,j,k}\phi^Z_k(t)$$

where $\xi^Y_{r,j,k}$'s and $\xi^Z_{r,i,j,k}$'s are all independent mean-zero normal variables. We set $Var[\xi^Y_{r,j,k}] = 0.2$ and $Var[\xi^Z_{r,i,j,k}] = 0.05$. We set $\{\phi^Y_1(t), \phi^Y_2(t)\} = \{1, \sin(2\pi t)\}$ and $\{\phi^Z_1(t), \ldots, \phi^Z_4(t)\} = \{1_{[0,0.5]}, 1_{(0.5,1]}, 1_{(1,1.5]}, 1_{(1.5,2]}\} \times 2\sin(4\pi t)$. Moreover, in order to model the dependence among different days, we let $Y^r_j(t) = 0.8\tilde{Y}^{(}_j t) + 0.6\tilde{Y}^r_{j-1}(t)$ for $j > 1$.

Evaluation Metric. For synthetic data, we introduce the criterion *clustering purity* (Schütze et al., 2008) to evaluate the clustering accuracy.

$$\text{Purity} = \frac{1}{n}\sum_{c=1}^{C} \max_{j\in\{1,\cdots,C\}} |\mathcal{W}_c \cap \mathcal{C}_j|,$$

where $\mathcal{W}_c$ is the estimated index set of sequences belonging to the cth group, $\mathcal{C}_j$ is the true index set of sequence belonging to the jth cluster, and $|\cdot|$ is the cardinality counting the number of elements in a set. The value of *clustering purity* resides in $[0, 1]$ with a higher value indicating a more accurate clustering (=1 if the estimated clusters completely overlap with the truth).

B.3 Additional Real Data Examples and Details

Evaluation Metric

In the real data example, we evaluate and compare clustering stability based on a measure called *clustering consistency* via K-trial cross validations (Tibshirani and Walther, 2005; von Luxburg, 2009), as there is no ground truth clustering labels.

It works with the following rationale: because random sampling does not change the clustering structure of data, a clustering method with high consistency should preserve the pairwise relationships of samples in different trials. Specifically, we perform the clustering with K trials. In the k-th trial, we randomly separate the accounts into two folds. One fold contains 80% of accounts and serves as the training set, and we predict the cluster memberships of remaining accounts with the trained model. Let $\mathcal{M}_k = \{(i, i')|i, i'$ belong to the same cluster$\}$ enumerate all pairs of accounts with the same cluster index in the k-th trial. Then we define the *clustering consistency* as:

$$\text{Clustering Consistency} = \min_{k \in \{1, \cdots, K\}} \sum_{k' \neq k} \sum_{(i,i') \in \mathcal{M}_k} \frac{1\{c_i^k = c_{i'}^{k'}\}}{|K-1||\mathcal{M}_k|}$$

where $1\{\cdot\}$ is an indicator function and c_i^j denote the learned cluster index of the account i in the k-th trial.

Additional Results on Chase Credit Card Dataset.

In the credit card transaction dataset, there is a large variation in the frequencies in credit card use across users. We removed the users with fewer than 100 total transactions. The BIC suggests clustering the users into 3 groups. In each cluster, we obtained the estimated surface of covariance function $\Gamma_{x,c}(s, t)$, which is displayed in Figure B.1. Compared with clust 2 and 3, the latent process $\boldsymbol{X}_i(t)$ in clust 1 has relatively larger variation. To offer a more straightforward view of the correlation among events, we computed the average correlations as,

$$\overline{\text{Corr}}(r) = \frac{\sum_{|t-s|=r} \widehat{\text{Corr}}(t, s)}{\sum_{|t-s|=r} 1}$$

Where $\widehat{\text{Corr}}(t, s) = \widehat{\Gamma}_{x,c}(t, s)/\sqrt{\widehat{\Gamma}_{x,c}(t, t)\widehat{\Gamma}_{x,c}(s, s)}$. Figure B.2 displays the averaged correla-

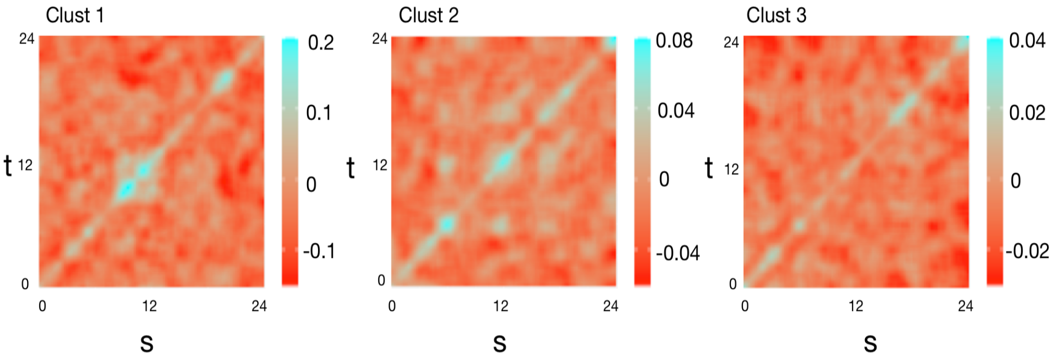

Figure B.1: Credit Card Dataset: Estimated $\Gamma_{x,c}(s, t)$ for each cluster;

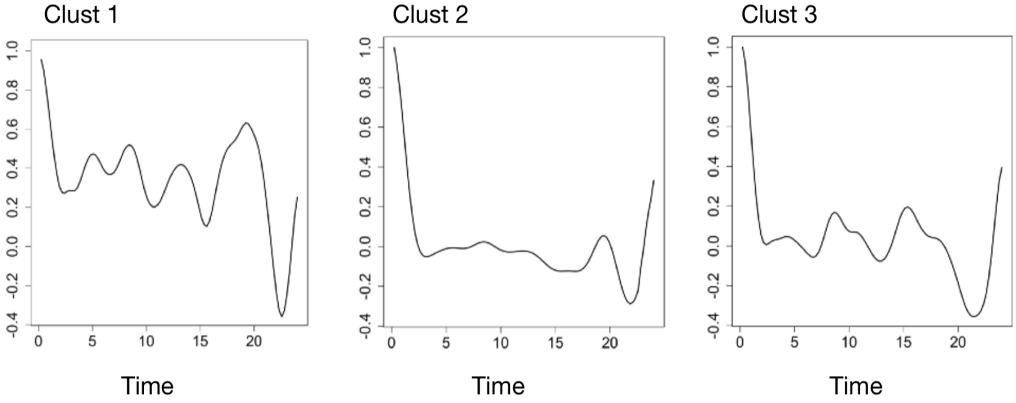

Figure B.2: Credit Card Dataset: Averaged correlations versus time lags;

tions versus time lags. There appears to be a periodic pattern in credit card use for clust 1 and 3. The users in clust 1 seemed to use their credit cards most frequently since the plot of clust 1 has the most number of crests. It is consistent with our facts that users in clust 1 averagely used credit cards 3.7 times a day, versus 1.3 times and 2.2 times a day for clust 2 and 3 respectively.

REFERENCES

Lihao Yin and Huiyan Sang. Fused spatial point process intensity estimation with varying coefficients on complex constrained domains. *Spatial Statistics*, 46:100547, 2021.

Lihao Yin, Huiyan Sang, Douglas J Schnoebelen, Brian Wels, Don Simmons, Alyssa Mattson, Michael Schueller, Michael Pentella, and Susie Y Dai. Risk based arsenic rational sampling design for public and environmental health management. *Chemometrics and Intelligent Laboratory Systems*, 211:104274, 2021a.

Lihao Yin, Ganggang Xu, Huiyan Sang, and Yongtao Guan. Row-clustering of a point process-valued matrix. *Advances in Neural Information Processing Systems*, 34, 2021b.

Jesper Møller and Rasmus P Waagepetersen. Modern statistics for spatial point processes. *Scandinavian Journal of Statistics*, 34(4):643–684, 2007.

Yongtao Guan. A composite likelihood approach in fitting spatial point process models. *Journal of the American Statistical Association*, 101(476):1502–1512, 2006.

Yongtao Guan, Abdollah Jalilian, and Rasmus Waagepetersen. Quasi-likelihood for spatial point processes. *Journal of the Royal Statistical Society. Series B (Statistical methodology)*, 77(3): 677, 2015.

Thomas J Leininger, Alan E Gelfand, et al. Bayesian inference and model assessment for spatial point patterns using posterior predictive samples. *Bayesian Analysis*, 12(1):1–30, 2017.

Flávio B Gonçalves and Dani Gamerman. Exact Bayesian inference in spatiotemporal Cox processes driven by multivariate Gaussian processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(1):157–175, 2018.

Shinichiro Shirota and Sudipto Banerjee. Scalable inference for space-time Gaussian Cox processes. *Journal of Time Series Analysis*, 40(3):269–287, 2019.

Peter Diggle. A kernel method for smoothing point process data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 34(2):138–147, 1985.

M Chris Jones, James S Marron, and Simon J Sheather. A brief survey of bandwidth selection for density estimation. *Journal of the American statistical association*, 91(433):401–407, 1996.

Christopher D Barr and Frederic Paik Schoenberg. On the Voronoi estimator for the intensity of an inhomogeneous planar Poisson process. *Biometrika*, 97(4):977–984, 2010.

A Stewart Fotheringham, Chris Brunsdon, and Martin Charlton. *Geographically weighted regression: the analysis of spatially varying relationships*. John Wiley & Sons, 2003.

Greg McSwiggan, Adrian Baddeley, and Gopalan Nair. Kernel density estimation on a linear network. *Scandinavian Journal of Statistics*, 44(2):324–345, 2017.

M Mehdi Moradi, Francisco J Rodríguez-Cortés, and Jorge Mateu. On kernel-based intensity estimation of spatial point patterns on linear networks. *Journal of Computational and Graphical Statistics*, 27(2):302–311, 2018.

Suman Rakshit, Tilman Davies, M Mehdi Moradi, Greg McSwiggan, Gopalan Nair, Jorge Mateu, and Adrian Baddeley. Fast kernel smoothing of point patterns on a large network using two-dimensional convolution. *International Statistical Review*, 87(3):531–556, 2019.

Adrian Baddeley, Gopalan Nair, Suman Rakshit, Greg McSwiggan, and Tilman M Davies. Analysing point patterns on networks—a review. *Spatial Statistics*, page 100435, 2020.

M Mehdi Moradi, Ottmar Cronie, Ege Rubak, Raphael Lachieze-Rey, Jorge Mateu, and Adrian Baddeley. Resample-smoothing of Voronoi intensity estimators. *Statistics and Computing*, 29 (5):995–1010, 2019.

Robert Bassett and James Sharpnack. Fused density estimation: theory and methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 81(5):839–860, 2019.

Adrian Baddeley, Ya-Mei Chang, Yong Song, and Rolf Turner. Nonparametric estimation of the dependence of a spatial point process on spatial covariates. *Statistics and its interface*, 5(2): 221–236, 2012.

Greg McSwiggan. Spatial point process methods for linear networks with applications to road accident analysis. *(PhD thesis) University of Western Australia*, 2019.

Jony Arrais Pinto Junior, Dani Gamerman, Marina Silva Paez, and Regina Helena Fonseca Alves. Point pattern analysis with spatially varying covariate effects, applied to the study of cerebrovascular deaths. *Statistics in Medicine*, 34(7):1214–1226, 2015.

Alan E Gelfand, Hyon-Jung Kim, CF Sirmans, and Sudipto Banerjee. Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association*, 98(462): 387–396, 2003.

Alan E Gelfand, Peter Diggle, Peter Guttorp, and Montserrat Fuentes. *Handbook of spatial statistics*. CRC press, 2010.

Rasmus Plenge Waagepetersen. An estimating function approach to inference for inhomogeneous Neyman–Scott processes. *Biometrics*, 63(1):252–258, 2007.

Adrian Baddeley, Jean-François Coeurjolly, Ege Rubak, and Rasmus Waagepetersen. Logistic regression for spatial Gibbs point processes. *Biometrika*, 101(2):377–392, 2014.

Achmad Choiruddin, Jean-François Coeurjolly, Frédérique Letué, et al. Convex and non-convex regularization methods for spatial point processes intensity estimation. *Electronic Journal of Statistics*, 12(1):1210–1255, 2018.

Yongtao Guan and Ye Shen. A weighted estimating equation approach for inhomogeneous spatial point processes. *Biometrika*, 97(4):867–880, 2010.

Mark Berman and T Rolf Turner. Approximating point process likelihoods with glim. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 41(1):31–38, 1992.

Jesper Moller and Rasmus Plenge Waagepetersen. *Statistical inference and simulation for spatial point processes*. CRC Press, 2003.

Ryan J Tibshirani, Jonathan Taylor, et al. The solution path of the generalized lasso. *The Annals of Statistics*, 39(3):1335–1371, 2011.

Taylor B Arnold and Ryan J Tibshirani. Efficient implementations of the generalized Lasso dual path algorithm. *Journal of Computational and Graphical Statistics*, 25(1):1–27, 2016.

Furong Li and Huiyan Sang. Spatial homogeneity pursuit of regression coefficients for large datasets. *Journal of the American Statistical Association*, 114(527):1050–1062, 2019.

Achmad Choiruddin, Jean-François Coeurjolly, and Rasmus Waagepetersen. Information criteria for inhomogeneous spatial point processes. *Australian & New Zealand Journal of Statistics*, 2021.

Hui Zou. The adaptive Lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429, 2006.

Jianqing Fan and Runze Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

Cun-Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, 38:894–942, 2010.

Gene H Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

Jiahua Chen and Zehua Chen. Extended BIC for small-n-large-P sparse GLM. *Statistica Sinica*, pages 555–574, 2012.

Blake Shaw and Tony Jebara. Structure preserving embedding. *International Conference on Machine Learning (ICML)*, 2009.

Der-Tsai Lee. Two-dimensional Voronoi diagrams in the L_p-metric. *Journal of the ACM*, 27(4): 604–618, 1980.

Finn Lindgren, Håvard Rue, and Johan Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498, 2011.

Jingru Mu, Guannan Wang, and Li Wang. Estimation and inference in spatially varying coefficient models. *Environmetrics*, 29(1):e2485, 2018.

Oscar Hernan Madrid Padilla, James Sharpnack, James G Scott, and Ryan J Tibshirani. The DFS fused Lasso: linear-time denoising over general graphs. *Journal of Machine Learning Research*, 18:176–1, 2018.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Amir Beck and Marc Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE transactions on image processing*, 18(11):2419–2434, 2009.

Bo Wahlberg, Stephen Boyd, Mariette Annergren, and Yang Wang. An ADMM algorithm for a class of total variation regularized estimation problems. *IFAC Proceedings Volumes*, 45(16): 83–88, 2012.

Jesper Møller, Anne Randi Syversveen, and Rasmus Plenge Waagepetersen. Log gaussian cox processes. *Scandinavian journal of statistics*, 25(3):451–482, 1998.

Kathryn Bullock Davis. Mean integrated square error properties of density estimates. *The Annals of Statistics*, 5:530–535, 1977.

Adrian Baddeley and Rolf Turner. Spatstat: an R package for analyzing spatial point patterns. *Journal of Statistical Software*, 12(6):1–42, 2005.

Patrick E Brown. Model-based geostatistics the easy way. *Journal of Statistical Software*, 63(12): 1–24, 2015.

Patrick E Brown et al. Model-based geostatistics the easy way. *Journal of Statistical Software*, 63 (12):1–24, 2015.

Joel Podgorski and Michael Berg. Global threat of arsenic in groundwater. *Science*, 368(6493): 845–850, 2020.

Leslie A DeSimone and Pixie A Hamilton. *Quality of water from domestic wells in principal aquifers of the United States, 1991-2004.* US Department of the Interior, US Geological Survey, 2009.

Kirsten S Almberg, Mary E Turyk, Rachael M Jones, Kristin Rankin, Sally Freels, Judith M Graber, and Leslie T Stayner. Arsenic in drinking water and adverse birth outcomes in Ohio. *Environmental research*, 157:52–59, 2017.

Marie Vahter. Effects of arsenic on maternal and fetal health. *Annual review of nutrition*, 29: 381–399, 2009.

Nazmul Sohel, Lars Åke Persson, Mahfuzar Rahman, Peter Kim Streatfield, Muhammad Yunus, Eva-Charlotte Ekström, and Marie Vahter. Arsenic in drinking water and adult mortality: a population-based cohort study in rural Bangladesh. *Epidemiology*, pages 824–830, 2009.

Maria Argos, Habibul Ahsan, and Joseph H Graziano. Arsenic and human health: epidemiologic progress and public health implications. *Reviews on environmental health*, 27(4):191–195, 2012.

Michael S Bloom, Simona Surdu, Iulia A Neamtiu, and Eugen S Gurzau. Maternal arsenic exposure and birth outcomes: a comprehensive review of the epidemiologic literature focused on drinking water. *International journal of hygiene and environmental health*, 217(7):709–719, 2014.

National Groundwater Association. Groundwater use in the United States of America, 2020. URL `https://www.ngwa.org/docs/default-source/default-document-library/groundwater/usa-groundwater-use-fact-sheet.pdf?sfvrsn=5c7a0db8_4`.

Douglas J Schnoebelen, Sophia Walsh, Brian Hanft, Oscar E Hernandez-Murcia, and Chad Fields. Elevated Arsenic in Private Wells of Cerro Gordo County, Iowa: Causes and Policy Changes. *Journal of Environmental Health*, 79(9), 2017.

Pentti Minkkinen. Practical applications of sampling theory. *Chemometrics and intelligent laboratory systems*, 74(1):85–94, 2004.

US Environmental Protection Agency (USEPA). Guidance on choosing a sampling design for environmental data collection, 2002.

Luzia Gonçalves, M Rosário de Oliveira, Cláudia Pascoal, and Ana Pires. Sample size for estimating a binomial proportion: comparison of different methods. *Journal of Applied Statistics*, 39 (11):2453–2473, 2012.

Kyung-Min Lee, Timothy J Herrman, and Susie Y Dai. Application and validation of a statistically derived risk-based sampling plan to improve efficiency of inspection and enforcement. *Food Control*, 64:135–141, 2016.

Nuno Sepúlveda and Chris Drakeley. Sample size determination for estimating antibody seroconversion rate under stable malaria transmission intensity. *Malaria journal*, 14(1):141, 2015.

Lawrence Joseph and Caroline Reinhold. Statistical inference for continuous variables. *American*

Journal of Roentgenology, 184(4):1047–1056, 2005.

Manouchehr Amini, Karim C Abbaspour, Michael Berg, Lenny Winkel, Stephan J Hug, Eduard Hoehn, Hong Yang, and C Annette Johnson. Statistical modeling of global geogenic arsenic contamination in groundwater. *Environmental science & technology*, 42(10):3669–3675, 2008.

Joseph D Ayotte, Bernard T Nolan, John R Nuckols, Kenneth P Cantor, Gilpin R Robinson, Dalsu Baris, Laura Hayes, Margaret Karagas, William Bress, Debra T Silverman, et al. Modeling the probability of arsenic in groundwater in New England as a tool for exposure assessment. *Environmental science & technology*, 40(11):3578–3585, 2006.

Lenny Winkel, Michael Berg, Manouchehr Amini, Stephan J Hug, and C Annette Johnson. Predicting groundwater arsenic contamination in Southeast Asia from surface parameters. *Nature Geoscience*, 1(8):536–542, 2008.

Joel E Podgorski, Syed Ali Musstjab Akber Shah Eqani, Tasawar Khanam, Rizwan Ullah, Heqing Shen, and Michael Berg. Extensive arsenic contamination in high-pH unconfined aquifers in the Indus Valley. *Science advances*, 3(8):e1700935, 2017.

Lenny HE Winkel, Pham Thi Kim Trang, Vi Mai Lan, Caroline Stengel, Manouchehr Amini, Nguyen Thi Ha, Pham Hung Viet, and Michael Berg. Arsenic pollution of groundwater in vietnam exacerbated by deep aquifer exploitation for more than a century. *Proceedings of the National Academy of Sciences*, 108(4):1246–1251, 2011.

Luis Rodríguez-Lado, Guifan Sun, Michael Berg, Qiang Zhang, Hanbin Xue, Quanmei Zheng, and C Annette Johnson. Groundwater arsenic contamination throughout China. *Science*, 341 (6148):866–868, 2013.

Qiang Yang, Hun Bok Jung, Robert G Marvinney, Charles W Culbertson, and Yan Zheng. Can arsenic occurrence rates in bedrock aquifers be predicted? *Environmental science & technology*, 46(4):2080–2087, 2012.

Joseph D Ayotte, Laura Medalie, Sharon L Qi, Lorraine C Backer, and Bernard T Nolan. Estimating the high-arsenic domestic-well population in the conterminous United States. *Environmental science & technology*, 51(21):12443–12454, 2017.

Joseph D Ayotte, Bernard T Nolan, and Jo Ann Gronberg. Predicting arsenic in drinking water wells of the Central Valley, California. *Environmental Science & Technology*, 50(14):7555–7563, 2016.

Melinda L Erickson, Sarah M Elliott, CA Christenson, and Aliesha L Krall. Predicting geogenic Arsenic in Drinking Water Wells in Glacial Aquifers, North-Central USA: Accounting for Depth-Dependent Features. *Water Resources Research*, 54(12):10–172, 2018.

Zhen Tan, Qiang Yang, and Yan Zheng. Machine Learning Models of Groundwater Arsenic Spatial Distribution in Bangladesh: Influence of Holocene Sediment Depositional History. *Environmental Science & Technology*, 54(15):9454–9463, 2020.

Zhengyuan Zhu and Michael L Stein. Spatial sampling design for prediction with estimated parameters. *Journal of agricultural, biological, and environmental statistics*, 11(1):24, 2006.

Peter J Diggle, Raquel Menezes, and Ting-li Su. Geostatistical inference under preferential sampling. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 59(2):191–232, 2010.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

Iowa Department of Public Health. Iowa Administrative Code 641, Chapter 24, Private Well Testing, Reconstruction, and Plugging- Grants to Counties , 2016.

Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, 2011.

Oscar Hernan Madrid Padilla, James Sharpnack, Yanzhen Chen, and Daniela M Witten. Adaptive nonparametric regression with the k-nearest neighbour fused lasso. *Biometrika*, 107(2):293–310, 2020.

Gideon Schwarz et al. Estimating the dimension of a model. *Annals of statistics*, 6(2):461–464, 1978.

Jiahua Chen and Zehua Chen. Extended Bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3):759–771, 2008.

Robert G Newcombe. Two-sided confidence intervals for the single proportion: comparison of seven methods. *Statistics in medicine*, 17(8):857–872, 1998.

Jean Cutler Prior, Janice L Boekhoff, Mary R Howes, Robert D Libra, and Paul E VanDorpe. Iowa's Groundwater Basics: A geological guide to the occurence, use, and vulnerability of Iowa's aquifers. 2003.

Thomas A Lasko. Efficient inference of gaussian-process-modulated renewal processes with application to medical event data. In *Uncertainty in artificial intelligence: proceedings of the... conference. Conference on Uncertainty in Artificial Intelligence*, volume 2014, page 469. NIH Public Access, 2014.

Ganggang Xu, Ming Wang, Jiangze Bian, Hui Huang, Timothy Burch, Sandro Andrade, Jingfei Zhang, and Yongtao Guan. Semi-parametric learning of structured temporal point processes. *Journal of machine learning research*, 2020.

Husna Sarirah Husin, Lishan Cui, Herny Ramadhani Husny Hamid, and Norhaiza Ya Abdullah. Time series analysis of web server logs for an online newspaper. In *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication*, pages 1–4, 2013.

T Warren Liao. Clustering of time series data—a survey. *Pattern recognition*, 38(11):1857–1874, 2005.

Elizabeth Ann Maharaj. Cluster of time series. *Journal of Classification*, 17(2):297–314, 2000.

Jarke J Van Wijk and Edward R Van Selow. Cluster and calendar based visualization of time series data. In *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis' 99)*, pages 4–9. IEEE, 1999.

Konstantinos Kalpakis, Dhiral Gada, and Vasundhara Puttagunta. Distance measures for effective clustering of arima time-series. In *Proceedings 2001 IEEE international conference on data mining*, pages 273–280. IEEE, 2001.

Xianping Ge and Padhraic Smyth. Deformable markov model templates for time-series pattern matching. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 81–90, 2000.

Dixin Luo, Hongteng Xu, Yi Zhen, Bistra Dilkina, Hongyuan Zha, Xiaokang Yang, and Wenjun Zhang. Learning mixtures of markov chains from aggregate data with structural constraints. *IEEE Transactions on Knowledge and Data Engineering*, 28(6):1518–1531, 2016.

Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: volume I: elementary theory and methods*. Springer, 2003.

Alan G Hawkes and David Oakes. A cluster process representation of a self-exciting process. *Journal of Applied Probability*, pages 493–503, 1974.

Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, pages 359–370, 1994.

Paul S Bradley and Usama M Fayyad. Refining initial points for k-means clustering. In *ICML*, volume 98, pages 91–99. Citeseer, 1998.

Tao Pei, Xi Gong, Shih-Lung Shaw, Ting Ma, and Chenghu Zhou. Clustering of temporal event processes. *International Journal of Geographical Information Science*, 27(3):484–510, 2013.

Hongteng Xu and Hongyuan Zha. A dirichlet mixture model of hawkes processes for event sequence clustering. *arXiv preprint arXiv:1701.09177*, 2017.

Dixin Luo, Hongteng Xu, Yi Zhen, Xia Ning, Hongyuan Zha, Xiaokang Yang, and Wenjun Zhang. Multi-task multi-dimensional hawkes processes for modeling event sequences. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

Jie Peng, Hans-Georg Müller, et al. Distance-based clustering of sparsely observed stochastic processes, with applications to online auctions. *Annals of Applied Statistics*, 2(3):1056–1077, 2008.

Weichang Wu, Junchi Yan, Xiaokang Yang, and Hongyuan Zha. Discovering temporal patterns for event sequence clustering via policy mixture model. *IEEE Transactions on Knowledge and Data Engineering*, 2020a.

Murray Aitkin and Donald B Rubin. Estimation and hypothesis testing in finite mixture models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 47(1):67–75, 1985.

Michael Elashoff and Louise Ryan. An EM algorithm for estimating equations. *Journal of Computational and Graphical Statistics*, 13(1):48–65, 2004.

JO Ramsay and BW Silverman. Principal components analysis for functional data. *Functional data analysis*, pages 147–172, 2005.

Satosi Watanabe. Karhunen-loeve expansion and factor analysis: theoretical remarks and application. In *Trans. on 4th Prague Conf. Information Theory, Statistic Decision Functions, and Random Processes Prague*, pages 635–660, 1965.

Hong-Zhong Wu, Jun-Jie Zhang, Long-Gang Pang, and Qun Wang. Zmcintegral: A package for multi-dimensional monte carlo integration on multi-gpus. *Computer Physics Communications*, 248:106962, 2020b.

Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008.

Charles R Harris, K Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020.

Michael W Kearney. rtweet: Collecting and analyzing twitter data. *Journal of Open Source Software*, 4(42):1829, 2019.

Robert Tibshirani and Guenther Walther. Cluster validation by prediction strength. *Journal of Computational and Graphical Statistics*, 14(3):511–528, 2005.

Ulrike von Luxburg. Clustering stability: An overview. *Machine Learning*, 2(3):235–274, 2009.

Anders Hildeman, David Bolin, Jonas Wallin, and Janine B Illian. Level set Cox processes. *Spatial statistics*, 28:169–193, 2018.

Fan Yin, Guanyu Hu, and Weining Shen. Analysis of professional basketball field goal attempts via a bayesian matrix clustering approach. *arXiv preprint arXiv:2010.08495*, 2020.

Kristian Bjørn Hessellund, Ganggang Xu, Yongtao Guan, and Rasmus Waagepetersen. Semi-parametric multinomial logistic regression for multivariate point pattern data. *Journal of the American Statistical Association*, pages 1–16, 2021.

LH Shen, DL Young, DC Lo, and CP Sun. Local differential quadrature method for 2-D flow and forced-convection problems in irregular domains. *Numerical Heat Transfer, Part B: Fundamentals*, 55(2):116–134, 2009.

Atsuyuki Okabe and Kokichi Sugihara. *Spatial analysis along networks: statistical and computational methods*. John Wiley & Sons, 2012.

Takehiro Furuta, Atsuo Suzuki, and Atsuyuki Okabe. A Voronoi heuristic approach to dividing networks into equal-sized sub-networks. *Forma*, 23(2):73–79, 2008.

Shino Shiode. Analysis of a distribution of point events using the network-based quadrat method. *Geographical Analysis*, 40(4):380–400, 2008.

Geoffrey J McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.

G Peter Lepage. A new algorithm for adaptive multidimensional integration. *Journal of Computational Physics*, 27(2):192–203, 1978.

Thorsten Ohl. Vegas revisited: Adaptive monte carlo integration beyond factorization. *Computer physics communications*, 120(1):13–19, 1999.

Stefano Carrazza and Juan M Cruz-Martinez. Vegasflow: accelerating monte carlo simulation across multiple hardware platforms. *Computer Physics Communications*, 254:107376, 2020.