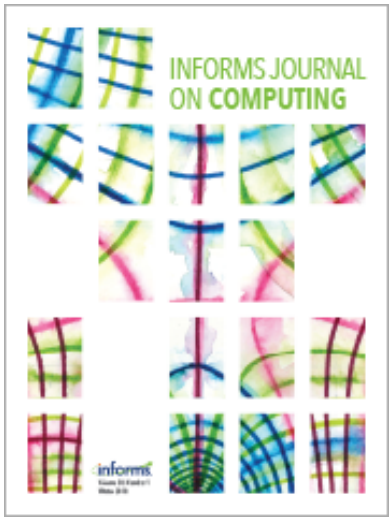



INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

# Clustering Then Estimation of Spatio-Temporal Self-Exciting Processes

Haoting Zhang, Donglin Zhan, James Anderson, Rhonda Righter, Zeyu Zheng

To cite this article:
Haoting Zhang, Donglin Zhan, James Anderson, Rhonda Righter, Zeyu Zheng (2024) Clustering Then Estimation of Spatio-Temporal Self-Exciting Processes. INFORMS Journal on Computing

Published online in Articles in Advance 05 Sep 2024

. https://doi.org/10.1287/ijoc.2022.0351

https://pubsonline.informs.org/journal/ijoc

INFORMS JOURNAL ON COMPUTING
*Articles in Advance*, pp. 1–20
ISSN 1091-9856 (print), ISSN 1526-5528 (online)

# Clustering Then Estimation of Spatio-Temporal Self-Exciting Processes

**Haoting Zhang,[a] Donglin Zhan,[b] James Anderson,[b] Rhonda Righter,[a] Zeyu Zheng[a,*]**

[a] Industrial Engineering and Operations Research Department, University of California Berkeley, Berkeley, California 94720; [b] Department of Electrical Engineering, Columbia University, New York, New York 10027

*Corresponding author

**Contact:** haoting_zhang@berkeley.edu, https://orcid.org/0000-0003-0058-6788 (HZ); donglin.zhan@columbia.edu, https://orcid.org/0009-0002-1201-3979 (DZ); james.anderson@columbia.edu, https://orcid.org/0000-0001-8210-6527 (JA); rrighter@berkeley.edu, https://orcid.org/0000-0001-6948-8145 (RR); zyzheng@berkeley.edu, https://orcid.org/0000-0001-5653-152X (ZZ)

**Received:** November 18, 2022
**Revised:** July 7, 2023; March 19, 2024; July 11, 2024
**Accepted:** July 15, 2024
**Published Online in Articles in Advance:** September 5, 2024

**https://doi.org/10.1287/ijoc.2022.0351**



**Abstract.** We propose a new estimation procedure for general spatio-temporal point processes that include a self-exciting feature. Estimating spatio-temporal self-exciting point processes with observed data is challenging, partly because of the difficulty in computing and optimizing the likelihood function. To circumvent this challenge, we employ a Poisson cluster representation for spatio-temporal self-exciting point processes to simplify the likelihood function and develop a new estimation procedure called "clustering-then-estimation" (CTE), which integrates clustering algorithms with likelihood-based estimation methods. Compared with the widely used expectation-maximization (EM) method, our approach separates the cluster structure inference of the data from the model selection. This has the benefit of reducing the risk of model misspecification. Our approach is computationally more efficient because it does not need to recursively solve optimization problems, which would be needed for EM. We also present asymptotic statistical results for our approach as theoretical support. Experimental results on several synthetic and real data sets illustrate the effectiveness of the proposed CTE procedure.

**History:** Accepted by Ram Ramesh, Area Editor for Data Science & Machine Learning.
**Funding:** J. Anderson is supported by NSF [Grant ECCS-2144634]. R. Righter is supported by the Ron Wolff Chaired Professorship. Z. Zheng is supported by NSF [Grant DMS-2220537].
**Supplemental Material:** The software that supports the findings of this study is available within the paper and its Supplemental Information (https://pubsonline.informs.org/doi/suppl/10.1287/ijoc.2022.0351) as well as from the IJOC GitHub software repository (https://github.com/INFORMSJoC/2022.0351). The complete IJOC Software and Data Repository is available at https://informsjoc.github.io/.

**Keywords:** spatio-temporal self-exciting point process • maximum likelihood estimation • clustering algorithm

## 1. Introduction

Point processes have been widely adopted in operations management research to model the times at which arrivals enter a system. The most common model for the arrival process of a queuing system is a Poisson process (Brown et al. 2005; Kim and Whitt 2014a, b; Zheng and Glynn 2017; Chen et al. 2019, 2024). See Zhang et al. (2014), Daw and Pender (2018), Gao and Zhu (2018), Liu et al. (2019a), and Chen (2021) for other arrival process models. These arrivals (or occurrences) are represented by points in a mathematical space (e.g., a vector space). When the locations of arrivals are taken into consideration, as is done in areas such as transportation and the sharing economy, a spatio-temporal point process is required (Diggle 2006, Zhou et al. 2015, Zhang and Zheng 2020). The spatial information of the arrival is modeled as *marks* of the associated points. In this work, we specifically consider spatio-temporal point processes with a *self-exciting* feature. The self-exciting feature captures triggering and clustering behaviors that are frequently observed in practical applications, such as in finance, epidemiology, commerce with network effects seismology, and criminology. See Reinhart (2018) for a review.

We study spatio-temporal self-exciting processes determined by the *conditional intensity function* (Daley and Vere-Jones 2003, 2007). This function is defined as the limiting ratio of the expected number of occurrences to the "volume" of the concerned infinitesimal time period times infinitesimal spatial area, conditional on the history of the point process. The statistical property of a point process is fully captured by the conditional intensity function through the finite-dimensional distribution (Daley and Vere-Jones 2003, 2007). Given observed data, the statistical estimation of the conditional intensity function is typically based on the maximum likelihood estimation

(MLE) method, but the corresponding optimization problem is difficult to solve and computationally intractable for two reasons. First, the likelihood function involves a summation of logarithms of conditional intensities, which themselves involve summations over previous points, making analytical maximization intractable. Second, the likelihood function can be nearly flat in large regions of the parameter space, causing problems for numerical maximization algorithms and making convergence extremely slow; see Ozaki (1979) and Veen and Schoenberg (2008).

### 1.1. Existing Approaches and Challenges

To address this challenge of maximizing the log-likelihood function, Veen and Schoenberg (2008) exploited a Poisson cluster structure to facilitate the estimation. As first explored by Hawkes and Oakes (1974), a point of a self-exciting process can be attributed to either the background underlying process or the triggering of a previous point. If the point is attributed to the background process, then it is among the *immigration* points. The set of points that are attributed to the points in the immigration is the first generation, the second generation is the set of points that are attributed to the points in the first generation, and so on. The attribution of all the points in a self-exciting process is known as the *branching structure*. The likelihood function is greatly simplified when the branching structure is included because the summation term involved in each logarithm reduces to one specific term. This in turn makes the optimization problem is amenable to an analytical or numerical solution (Veen and Schoenberg 2008). In most scenarios, the branching structure of a spatio-temporal self-exciting process is not directly observed from the given data. Therefore, the expectation-maximization (EM) method (Dempster et al. 1977) has been widely used to estimate the spatio-temporal self-exciting process by modeling the unobserved branching structure using latent variables.

However, methods based on EM are not without their own challenges. First, regarding the branching structure inference, EM relies on the model specification of the spatio-temporal self-exciting process. Thus, the accuracy of the branching structure inference suffers from any model misspecification. Second, from a computational point of view, iterating over the E and M steps requires solving optimization problems recursively and is therefore computationally inefficient and time-consuming. Finally, the Poisson cluster representation enables the self-exciting process to possess different triggering functions across different generations (Mehrdad and Zhu 2014, Fierro et al. 2015). On the other hand, because the EM method cannot exactly determine whether a given point is attributed to the background underlying process or the triggering of a previous point, it is not capable of estimating models with different triggering functions between different generations because no explicit generations are determined.

### 1.2. Contribution

The challenges faced by existing methods, including (1) the risk of model misspecification, (2) computational burdens, and (3) a lack of model flexibility, motivate us to propose a new method named "clustering-then-estimation" (CTE) for estimating spatio-temporal self-exciting processes. The method operates as follows. We first apply clustering algorithms to analyze the clustering behaviors of the points in a spatio-temporal self-exciting process. In particular, we select hard clustering algorithms that provide the exact attribution of each point, as opposed to probabilistic estimates such as those obtained by the EM method. Clustering algorithms are applied recursively, facilitating the inference of the branching structure. We then input the inferred branching structure into the log-likelihood function of the data and estimate the model by maximizing the log-likelihood function, simplified by the branching structure.

By separating the inference of the branching structure and model estimation, the proposed CTE method offers the following advantages. First, the inference of the branching structure through clustering algorithms is fully data driven and does not require knowledge of the self-exciting process model. Thus, it is less likely for CTE to suffer from model misspecification. Second, in the CTE method, the maximization of the simplified likelihood function is performed once after the branching structure is inferred by clustering algorithms. Without the need for recursive optimization procedures, CTE is more efficient to implement. Finally, the CTE method provides an explicit branching structure in which the attribution of each point is deterministic. As a result, different exciting features can be estimated through different pairs of generations, with different clusters provided. In other words, CTE enhances the model flexibility of the self-exciting process estimated from data.

Our contribution is summarized as follows:

1. We propose a "clustering-then-estimation (CTE)" approach to estimate the spatio-temporal self-exciting process. CTE utilizes clustering algorithms to infer the branching structure and simplifies the log-likelihood function to facilitate model estimation.

2. We prove the consistency and asymptotic normality of the proposed CTE estimators. By incorporating the branching structure, we provide regularity conditions that exhibit greater ease of verification compared with existing theoretical results.

3. We also introduce the tree-edit distance to evaluate the self-exciting model estimation. We show through experimental results that, compared with existing methods, the CTE method exhibits (1) better accuracy on the model estimation and branching structure inference, (2) less risk of model misspecification, (3) higher efficiency in practice without the necessity of recursively solving optimization problems, and (4) more flexibility of different triggering effects between different pairs of generations.

## 2. Literature Review

In this section, we first discuss the literature on self-exciting processes and then describe clustering algorithms that can be used to infer the branching structure.

### 2.1. Modeling and Learning Self-Exciting Processes

The self-exciting point process (also known as a *Hawkes process*) was first introduced by Hawkes (1971) as a temporal point process. The term "self-exciting" refers to the property that the occurrence of each event enhances the likelihood of future events, thereby creating a clustering behavior (Lima 2023). In some applications, there is a need to incorporate spatial dimensions, leading to the creation of a spatio-temporal self-exciting process. Spatio-temporal self-exciting processes have been widely used in modeling seismic events (Ogata 1998), crime activity (Mohler et al. 2011), ecology (Balderama et al. 2012), social network analysis (Yang and Zha 2013, Zhou et al. 2013a, Zipkin et al. 2016, Farajtabar et al. 2017, Rizoiu et al. 2017), financial markets (Errais et al. 2010, Filimonov and Sornette 2012), and "viral" processes on the Internet (Crane and Sornette 2008, Zhou et al. 2013b). Given the broad range of applications for spatio-temporal self-exciting processes, refining the estimation method for these processes can significantly enhance forecasting accuracy, bolster strategic decision-making, and catalyze new insights across diverse business sectors.

In recent years, the theory and technologies of machine learning have been extensively incorporated with the point process model in order to enhance model flexibility and prediction accuracy. For example, Recurrent Neural Networks (RNN) have been applied to construct the conditional intensity function of the self-exciting process; see Du et al. (2016), Mei and Eisner (2017), and Xiao et al. (2017b) for reference. In addition, Xiao et al. (2017a, 2018) proposed incorporating the Wasserstein-GAN model to model the intensity-free point processes. The modeling of the point process is transformed into a reinforcement learning problem by regarding the event as the action and the intensity function learning as the policy learning problem; see Li et al. (2018). The spatio-temporal versions of these advanced point process models can be found in Zhu et al. (2020, 2021a), Zhu et al. (2021b), Zhu and Xie (2022), and Dong et al. (2023). These advanced models enhance the prediction power of the self-exciting process and therefore have become an active and quickly developing research field. We note that some of these models could be incorporated into the framework of the "clustering-then-estimation" method we propose, but this is beyond the scope of the present work.

### 2.2. Clustering Algorithms

Given a data set, the selection of a clustering algorithm depends largely on the modeling assumptions. Typical cluster models (and associated algorithms) include (1) connectivity models (hierarchical clustering), (2) centroid models (K-means clustering), (3) distribution models (EM method), and (4) density models (DBSCAN algorithm). We refer to Gan et al. (2020) for a detailed review of clustering algorithms.

Clustering algorithms are generally divided into two categories: hard and soft clustering. Hard clustering algorithms definitively assign each data point to a specific cluster, whereas soft clustering algorithms offer the likelihood of a data point's affiliation to a particular cluster. For this analysis, hard clustering algorithms are the focus because of their necessity for recursive clustering. This requirement stems from the need for precise point attribution to each cluster, which in turn determines the next level of clustering in the branching structure inference. On the other hand, the EM method for estimating the spatio-temporal self-exciting process is an example of a soft clustering algorithm, offering a probability for a point's attribution. Additional soft clustering algorithms and probabilistic models have been combined with the self-exciting process, including the Dirichlet process. These models are typically Bayesian, and the models' posterior distributions can be approximated using methods such as Markov Chain Monte Carlo or Variational Bayesian inference. However, these methods are aimed predominantly at enhancing clustering algorithms and may modify the self-exciting process model, which is not the focus of the present analysis. Here, we refer to Du et al. (2015), Xu and Zha (2017), and Li and Bhowmick (2020).

Some management problems require decision-making for complex operating systems (Bollapragada et al. 2006, Li et al. 2016, Adikari and Dutta 2019, Guo et al. 2019, Bichler et al. 2021), and the status of the operating system requires estimation from the data (Tari et al. 2010, Chen and Liu 2014, Fan et al. 2017, Manrique-Vallier and Hu 2018, Liu et al. 2019b, Guo et al. 2020, Lin et al. 2022, Ahn et al. 2023). Clustering algorithms can be used to explore the intrinsic structure of the data, thus facilitating the estimation and management of operating systems. Research on clustering for management problems includes Gopal and Ramesh (1995), Brice et al. (2011), Seref et al. (2014), Hu et al. (2018), Ungun et al. (2019), Chen and Xie (2022), and Meng et al. (2022).

## 3. Model Description and Problem Statement

In Section 3.1, we present notions of the spatio-temporal self-exciting processes. Then, in Section 3.2, we formally state the problem we address.

### 3.1. Spatio-Temporal Self-Exciting Process

A spatio-temporal point process is a random point field that models both temporal and spatial dispersions of points. Each point represents the arrival of an event or an entity at a specific time and location. Let $s = (x_1, x_2, \dots, x_d) \in \mathcal{S} \subset \mathbb{R}^d$ denote the spatial variable and $t \in [0, T]$ denote the time variable. The finite-dimensional distributions of a spatio-temporal point process are uniquely determined by the associated conditional intensity function (Daley and Vere-Jones 2003, 2007), which is defined as

$$\lambda(s,t|\mathcal{H}_t) = \lim_{\Delta s, \Delta t \to 0} \frac{\mathbb{E}[N(B(s,\Delta s) \times [t, t+\Delta t))|\mathcal{H}_t]}{|B(s,\Delta s)|\Delta t},$$

where $\mathcal{H}_t$ denotes the history of the process, $N(\cdot)$ is the counting measure, and $|B(s,\Delta s)|$ denotes the Lebesgue measure of a ball $B(s,\Delta s)$ centered at $s$ with radius $\Delta s$. For notational simplicity, we will omit the history $\mathcal{H}_t$ in the conditional intensity function. This function characterizes the full dynamics of the associated spatio-temporal point process.

For a spatio-temporal self-exciting process, given a parameter $\theta$, the spatial locations $s_i$, and the time epochs $t_i$ of the observed points, the conditional intensity consists of two parts,

$$\lambda(s,t;\theta) = \mu(s,t;\theta) + \sum_{t_i < t} g(s - s_i, t - t_i; \theta), \tag{1}$$

where both the background intensity function $\mu(s,t;\theta)$ and the triggering function $g(s,t;\theta)$ are nonnegative functions defined on $\mathcal{S} \times [0,\infty)$. That is, the conditional intensity function is random and depends on the history $\mathcal{H}_t$ of the process. The background intensity function $\mu(s,t;\theta)$ generates a baseline Poisson process. For the model specification of the background intensity function, we refer the reader to the Poisson process-related literature (Henderson 2003, Chen and Schmeiser 2019, Morgan et al. 2019, Nelson and Leemis 2020). The nonnegative triggering function $g(s,t;\theta)$ models the "self-exciting" feature. We present some examples of spatio-temporal self-exciting processes in the supplements.

The spatio-temporal self-exciting process has a Poisson cluster representation. Here, we provide a formal description.

**Definition 1** (Poisson Cluster Process and Branching Structure). Consider a spatio-temporal point process so that (1) $\mathfrak{N}_0$ is a spatio-temporal Poisson process with intensity function $\mu(s,t;\theta)$ and (2) for any $n \in \{1,2,\dots\}$, $\mathfrak{N}_n$ is a spatio-temporal Poisson process with intensity function $\sum_{(s_i,t_i)\in\mathfrak{N}_{n-1}} g_n(s - s_i, t - t_i; \theta)$. The superposition $\cup_{n=0}^{\infty} \mathfrak{N}_n$ is a Poisson cluster process. That is, a Poisson process $\mathfrak{N}_0$, referred to as the *immigration* process, is generated with intensity function $\mu(s,t;\theta)$. Each point in $\mathfrak{N}_0$ triggers a Poisson process centered at itself with intensity function $g_1(s,t;\theta)$. Points triggered by the immigration are denoted as the first-generation $\mathfrak{N}_1$. The first generation then triggers the second generation, and so on. This parent-offspring relationship between the points is known as the *branching structure*.

**Proposition 1.** *Suppose a Poisson cluster process satisfying that* (1) $\int_{\mathcal{S}} \mu(s,t;\theta)\,\mathrm{d}s < \infty$, (2) $g_n(s,t;\theta) \equiv g(s,t;\theta)$, *and* (3) $\tilde{n} \doteq \int_0^{\infty} \int_{\mathcal{S}} g(s,t;\theta)\,\mathrm{d}s\,\mathrm{d}t < 1$. *Then, the conditional intensity function of this process is exactly* (1).

Proposition 1 describes the Poisson cluster representation of the spatio-temporal self-exciting process. That is, for a spatio-temporal self-exciting process (1), there exists a Poisson cluster process defined as in Definition 1 that shares the same conditional intensity function as well as the parameter $\theta$. In this way, although the points in a spatio-temporal self-exciting process cannot be attributed exactly to either the background intensity or the previous points, the branching structure of the Poisson cluster process can be employed to impose a parent-offspring relationship. This branching structure facilitates the estimation of spatio-temporal self-exciting processes, which

we will describe in Section 3.2. For simplicity, the branching structure of a spatio-temporal self-exciting process is referred to as its Poisson cluster representation in the remaining text. Furthermore, we also provide the statistical property of the clusters in the branching structure; see the supplements.

### 3.2. Problem Definition

Our goal is to estimate the spatio-temporal self-exciting process given the observed data (a realization of the process). Consider a realization of the process over $\mathcal{S} \times [0, T]$, sorted by the time coordinate in ascending order: $\mathfrak{R} = \{\mathfrak{I}_1, \mathfrak{I}_2 \ldots, \mathfrak{I}_n\}$, where $\mathfrak{I}_i$ denotes the spatio-temporal information of the $i$-th arrival $(s_i, t_i)$, so $0 < t_1 < t_2 < \cdots < t_n < T$. The estimator attained by the maximum likelihood estimation (MLE) is

$$\hat{\theta}_{1;T}, \hat{\theta}_{2;T} = \arg\max_{\theta_1, \theta_2} \left\{ \sum_{i=1}^{n} \log \left( \mu(s_i, t_i; \theta_1) + \sum_{j:t_j < t_i} g(s_i - s_j, t_i - t_j; \theta_2) \right) - \int_0^T \int_{\mathcal{S}} \left( \mu(s, t; \theta_1) + \sum_{j:t_j < t} g(s - s_j, t - t_j; \theta_2) \right) \mathrm{d}s\mathrm{d}t \right\}, \tag{2}$$

where $\theta_1$ and $\theta_2$ denote the parameters involved in the background intensity $\mu$ and the triggering function $g$, respectively. That is, we consider the scenario when $\theta = (\theta_1, \theta_2)$ is separate over the background intensity and the triggering function. The optimization problem defined in (2) can be computationally prohibitive to solve. First, the form of the likelihood function involves a summation of logarithms of conditional intensities, which themselves involve summations over previous points, making analytical maximization intractable. Second, the complexity of evaluating the objective function is $\mathcal{O}(n^2)$ because of the double summation, where $n$ is the number of observed arrivals. When $n$ is large, it is computationally expensive to maximize the log-likelihood function. Finally, the likelihood function can be nearly flat in large regions of the parameter space, causing numerical problems and making convergence extremely slow; see also Ozaki (1979) and Veen and Schoenberg (2008).

In contrast, if the branching structure is known and incorporated, the log-likelihood function simplifies to

$$\ell(\theta) = \left( \sum_{i:(s_i, t_i) \in \mathfrak{R}_0} \log(\mu(s_i, t_i; \theta_1)) - \int_0^T \int_{\mathcal{S}} \mu(s, t; \theta_1) \mathrm{d}s\mathrm{d}t \right) + \sum_{j=1}^{n} \left( \sum_{i \in \mathfrak{D}_j} \log(g(s_i - s_j, t_i - t_j; \theta_2)) - \int_{t_j}^T \int_{\mathcal{S}} g(s - s_j, t - t_j; \theta_2) \mathrm{d}s\mathrm{d}t \right), \tag{3}$$

where $\mathfrak{D}_j$ denotes the set of indexes of the points directly triggered by the $j$-th point $\mathfrak{I}_j = (s_j, t_j)$. In most application scenarios, the observed data do not directly reveal the branching structure of a spatio-temporal self-exciting process. Therefore, the expectation-maximization (EM) method has been widely employed to estimate the spatio-temporal self-exciting process by modeling the unknown branching structure as the latent variable. However, the EM method suffers from (1) the risk of model misspecification, (2) computational burdens of iteratively solving optimization problems, and (3) limited model flexibility with an identical triggering function among different generations. See the supplements for a detailed review. Our "clustering-then-estimation" (CTE) method circumvents these challenges and facilitates the estimation of spatio-temporal self-exciting processes.

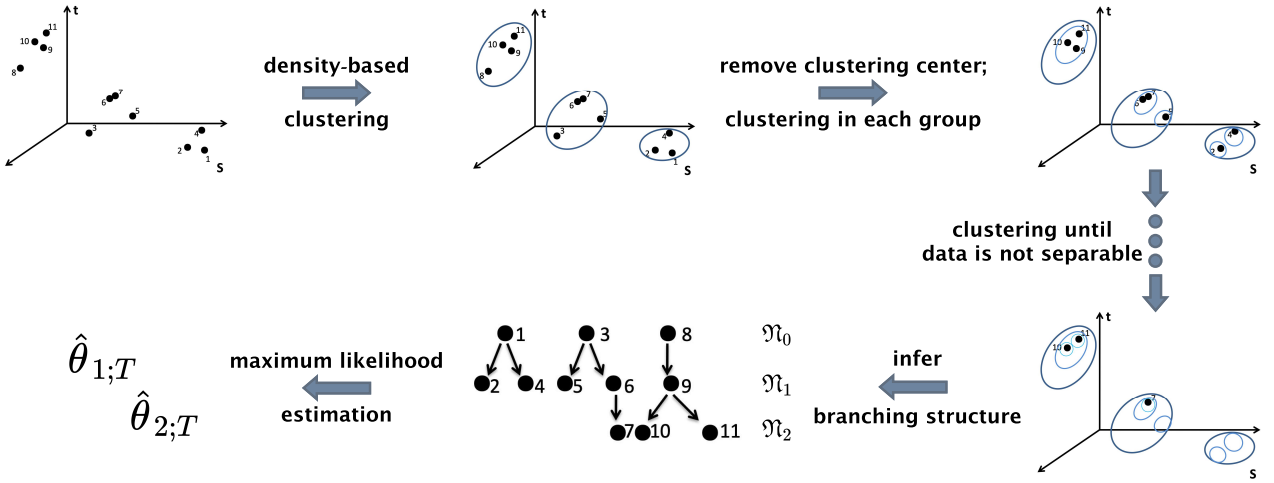
## 4. Methodology

In this section, we present a detailed description of the clustering-then-estimation (CTE) method to estimate the spatio-temporal self-exciting process. First, in Section 4.1, we describe the procedure of recursively applying clustering algorithms to infer the branching structure. With the inferred branching structure, we present the estimation procedure based on the simplified log-likelihood function in Section 4.2. The complete procedure of the CTE method is summarized in Figure 1.

### 4.1. Clustering for Branching Structure Inference

In this section, we present the procedure of recursive clustering to infer the branching structure. Here, we first present the general procedure of recursively performing the clustering algorithm for the branching structure inference in Section 4.1.1. Then, we describe a specific existing clustering algorithm named *density-based spatial clustering of applications with noise* (DBSCAN) as a representative in Section 4.1.2 and give the specific details of employing the DBSCAN algorithm in the CTE method in Section 4.1.3.

**Figure 1.** (Color online) Graphic Description of the CTE Method




**4.1.1. General Recursive Clustering Procedure.** We first justify our method by explaining (1) why the spatio-temporal self-exciting process exhibits clustering behaviors and (2) why we apply clustering algorithms recursively to infer the branching structure.

- **Spatio-temporal self-exciting processes exhibit clustering behaviors.** For each point $\mathfrak{I}_j = (s_j, t_j)$, we denote the set of points that are directly or indirectly generated by $\mathfrak{I}_j$ as $\mathcal{D}_j$. Because the triggering functions serve as the intensity function of a Poisson process centered around $\mathfrak{I}_j$ and the triggering functions are (generally) decreasing with time and spatial distance, the points in $\mathcal{D}_j$ are (generally) near $\mathfrak{I}_j$ in both space and time. Thus, the set $\{\mathfrak{I}_j\} \cup \mathcal{D}_j$ composes a cluster with center $\mathfrak{I}_j$.
- **Recursive clustering algorithms infer the branching structure.** From the bottom-up view, if a point $\mathfrak{I}_j = (s_j, t_j)$ is directly generated by another point, say $\mathfrak{I}_{j'}$, the cluster $\{\mathfrak{I}_j\} \cup \mathcal{D}_j$ then belongs to the cluster that is centered at the point $\mathfrak{I}_{j'}$. That is, $\{\mathfrak{I}_j\} \cup \mathcal{D}_j \subset \{\mathfrak{I}_{j'}\} \cup \mathcal{D}_{j'}$. Recursively, all the points are directly or indirectly attributed to a point in the immigration $\mathfrak{N}_0$. On the other hand, from a top-down view, the cluster excluding the cluster center, say $\mathcal{D}_j$, is composed of smaller clusters if $\mathcal{D}_j \neq \emptyset$. Based on the branching structure, these smaller clusters are respectively centered at those points that are directly generated by $\mathfrak{I}_j$, say $\mathfrak{D}_j$. In this way, the cluster excluding the cluster center $\mathcal{D}_j$ is further divided into several "smaller" clusters.

Thus, if we obtain the recursive structure of the clustering behaviors, the branching structure of the spatio-temporal self-exciting process can then be inferred.

In most applications, the branching structure of these clusters is not observed. Therefore, we recursively perform the clustering algorithm to infer the branching structure. The procedure is conducted top-down and is summarized as follows:

1. **Initialization:** Regard the entire observed data set $\mathfrak{R} = \{(s_1, t_1), (s_2, t_2), \ldots, (s_n, t_n)\}$ as the initial clustering result $\text{Clus}^{(0)}$.
2. **First-level clustering:** Perform the clustering algorithm on $\text{Clus}^{(0)}$, dividing it into several disjoint subsets (clusters) $C_j^{(1)}$ as $\text{Clus}^{(1)} = \{C_1^{(1)}, C_2^{(1)}, \ldots, C_{|\text{Clus}^{(1)}|}^{(1)}\}$. The point that occurs first in time within each cluster is specified as the cluster center $c_j^{(1)}$.
3. **Immigration specification:** The immigration, $\mathfrak{N}_0 = \{c_1^{(1)}, \ldots, c_{|\text{Clus}^{(1)}|}^{(1)}\}$, is specified as the set of all first-level cluster centers.
4. **Second-level clustering:** For each cluster obtained from the first-level clustering, we perform the clustering algorithm again after removing the cluster center. This results in subsets of each $C_j^{(1)}$, which are the second-level clusters, represented as $\text{Clus}_j^{(2)} = \{C_{(j),1}^{(2)}, C_{(j),2}^{(2)}, \ldots C_{(j),|\text{Clus}_j^{(2)}|}^{(2)}\}$. The set of new cluster centers in each $C_j^{(1)}$ is $\{c_{(j),1}^{(2)}, c_{(j),2}^{(2)}, \ldots c_{(j),|\text{Clus}_j^{(2)}|}^{(2)}\}$. The second level clustering result is $\text{Clus}^{(2)} = \cup_{j=1}^{|\text{Clus}^{(1)}|} \text{Clus}_j^{(2)}$.
5. **First generation specification:** The first generation,

$$\mathfrak{N}_1 = \cup_{j=1}^{|\text{Clus}^{(1)}|} \left\{ c_{(j),1}^{(2)}, c_{(j),2}^{(2)}, \ldots c_{(j),|\text{Clus}_j^{(2)}|}^{(2)} \right\},$$

is specified as the set of all second-level cluster centers.

**Table 1.** An Illustrative Example on Recursive Clustering Results for the Branching Structure Inference, Consistent with the 11 Points Shown in Figure 1

| | | |
|---|---|---|
| Observed data | $\text{Clus}^{(0)} = \{\Im_1, \Im_2, \Im_3, \Im_4, \Im_5, \Im_6, \Im_7, \Im_8, \Im_9, \Im_{10}, \Im_{11}\}$ | |
| First-level clustering | $\text{Clus}^{(1)} = \{\{\Im_1, \Im_2, \Im_4\}, \{\Im_3, \Im_5, \Im_6, \Im_7\}, \{\Im_8, \Im_9, \Im_{10}, \Im_{11}\}\}$ | $\mathfrak{N}_0 = \{\Im_1, \Im_3, \Im_8\}$ |
| Second-level clustering | $\text{Clus}^{(2)} = \{\{\Im_2\}, \{\Im_4\}, \{\Im_5\}, \{\Im_6, \Im_7\}, \{\Im_9, \Im_{10}, \Im_{11}\}\}$ | $\mathfrak{N}_1 = \{\Im_2, \Im_4, \Im_5, \Im_6, \Im_9\}$ |
| Third-level clustering | $\text{Clus}^{(3)} = \{\{\Im_7\}, \{\Im_{10}\}, \{\Im_{11}\}\}$ | $\mathfrak{N}_2 = \{\Im_7, \Im_{10}, \Im_{11}\}$ |

6. **Recursive clustering and generation specification:** Perform clustering and generation specification recursively. For each $k$-th level clustering result $\text{Clus}^{(k)}$, apply the clustering procedure on each $C_j^{(k)}$ after removing the cluster center, leading to the $(k+1)$-st level clusters. The centers of these clusters form the $k$-th generation.

7. **Termination:** Repeat the recursive clustering and generation specification process until the observed data are no longer separable; that is, every data point $\Im_j$ forms a cluster by itself. An illustrative example consistent with the 11 data points shown in Figure 1 is contained in Table 1.

#### 4.1.2. DBSCAN Clustering Algorithm.

In our CTE method, we do not impose specific restrictions on the choice of (hard) clustering algorithms, but we select the *density-based spatial clustering of applications with noise* (DBSCAN) algorithm (Ester et al. 1996, Schubert et al. 2017) as a representative to describe our method. This algorithm has been widely applied in clustering tasks for the following reasons. First, DBSCAN does not require specifying the number of clusters in advance, as opposed to the K-means algorithm. Second, DBSCAN takes the noise in the data into consideration and is robust to outliers. Third, DBSCAN can find arbitrarily shaped clusters. Most other density-based clustering algorithms are extensions of DBSCAN.

To describe the algorithm, we let $\mathcal{P}$ be a set of points in $d$-dimensional space $\mathbb{R}^d$. Given two points $p, q \in \mathbb{R}^d$, we denote by $dist(p, q)$ the Euclidean distance between $p$ and $q$. Denote by $B(p, r)$ the ball centered at a point $p \in \mathbb{R}^d$ with radius $r$. The DBSCAN algorithm takes two parameters, $\epsilon$ and *MinPts*, where $\epsilon$ is the radius of the neighborhood of a point and *MinPts* is the threshold for a data point to become the core point, defined below.

**Definition 1.** A point $p \in \mathcal{P}$ is named as a core point if $B(p, \epsilon)$ covers at least *MinPts* points of $\mathcal{P}$, including $p$ itself. If not, $p$ is then said to be a noncore point.

**Definition 2.** A point $q \in \mathcal{P}$ is density-reachable from $p \in \mathcal{P}$ if there is a sequence of points $p_1, p_2, \ldots, p_t \in P$ (for some integer $t \geqslant 2$) such that (1) $p_1 = p$ and $p_t = q$, (2) $p_1, p_2, \ldots, p_{t-1}$ are core points, and (3) $p_{i+1} \in B(p_i, \epsilon)$ for each $i \in [1, t-1]$.

**Definition 3.** Two points $p$ and $q$ are density connected if they are both density reachable from some point $o$. This definition is symmetric.

**Definition 4.** A cluster $C$ is a nonempty subset of $\mathcal{P}$ such that if a core point $p \in C$, then all the points that are density reachable from $p$ belong to $C$ as well.

With these definitions, the DBSCAN algorithm starts by first finding a core point, say $p$, and searches for all the density-reachable points from $p$. All these points, including the core point $p$, compose a cluster. After one certain cluster is found, another cluster is determined by starting from another core point that is outside the existing clusters. The algorithm ends when all the remaining points that do not belong to a cluster are noise points. We provide the detailed procedure of the DBSCAN algorithm in the supplements. In addition to DBSCAN, we also perform spatio-temporal DBSCAN, agglomerative clustering, and self-organizing map clustering for comparison. We exhibit the experimental results in Section 5.1 and present descriptions of these methods in the supplements.

#### 4.1.3. Recursive Clustering with DBSCAN.

We describe how the DBSCAN algorithm is employed in the CTE method to recursively determine the clusters. The complete clustering procedure for the branching structure inference is accomplished recursively until the data set is no longer separable. At each level of the clustering, we perform the DBSCAN method on each set of points to be divided into clusters. Recall that, as in Section 2.2, there are two parameters required to perform the DBSCAN: *MinPts* and $\epsilon$. During the recursions, *MinPts* is set to be 1 throughout because one point is able to form a cluster in the Poisson cluster representation of a spatio-temporal self-exciting process. Note that this setting will make all the points core points, and each point belongs to exactly one cluster. Meanwhile, this setting makes "density reachable" and "density connected" equivalent. Thus, for each set of points, the DBSCAN algorithm performs as follows with a determined radius $\epsilon$.

First, randomly select a point $p$ (definitely a core point), find all the points within the radius $\epsilon$, and denote the set as $C$ (including $p$). Second, for each point in $C$, say $p'$, find and include all the points within the radius from $p'$ into $C$. The inclusion procedure continues until all the remaining points excluded from $C$ are not within the radius

of any point in $C$. Finally, this set $C$ composes a cluster. The algorithm then finds another point outside $C$ to repeat the inclusion procedure. Note that all the points will finally be assigned to a cluster, possibly a singleton.

The hyperparameter $\epsilon$ in the DBSCAN algorithm must be adapted to each data set for clustering. To enable a data-driven algorithm, we set up a mechanism for fine-tuning the $\epsilon$ in each level of the clustering. The $\epsilon$ is computed based on the sample mean of distances between points in the data set, say $\tilde{d}$. Given a data set with size $\mathfrak{n}$, a total of $\binom{\mathfrak{n}}{2}$ pairs of data are selected to calculate the distance. To enhance efficiency, we randomly select some pairs of data with equal probability to approximate the distance between points in the set so that we do not need to calculate the distances of all $\binom{\mathfrak{n}}{2}$ pairs of data. The number of pairs is decided by a positive integer-valued ratio function $f(x)$ such that $f(\mathfrak{n}) < \binom{\mathfrak{n}}{2}$, for example, $f(x) = [(x-d-1)/(d+1)] + d + 1$, where $d$ is the dimension of the spatial space $\mathcal{S}$. The $\epsilon$ is set as a perturbation of the average of the distance between pairwise points $\tilde{d}$. Specifically, we set $\epsilon = r_0\tilde{d}$ for the first-level clustering and $\epsilon = \tilde{d} + r_1$ for all the remaining levels of clustering, where $r_0$ and $r_1$ are two user-selected parameters. We provide the detailed procedure of determining $(r_0, r_1)$ in the supplements.

We introduce another variable *StoppingList* to determine whether the algorithm should be stopped or not, that is, whether all the clusters at the latest level are no longer separable. Specifically, at the beginning of each level of clustering, the *StoppingList* will be set as an empty list. Then, for the current clustering result $\text{Clus}^{(k)} = \{C_1^{(k)}, C_2^{(k)}, \ldots, C_{|\text{Clus}^{(k)}|}^{(k)}\}$, a variable *True* or *False* will be appended to *StoppingList* if $|C_j^{(k)}| > 2$ or otherwise. The reason is that, when $|C_j^{(k)}| > 2$, the cluster excluding the cluster center $C_j^{(k)} \setminus \{c_j^{(k)}\}$ contains at least two points and therefore is separable. If $|C_j^{(k)}| = 2$, then the former point triggers the latter one, and no further clustering is required. If $|C_j^{(k)}| = 1$, then the single point constitutes a cluster itself. Therefore, when all the variables in *StoppingList* are *False*, the recursive clustering is stopped. The complete procedure of the clustering for the branching structure inference with the DBSCAN algorithm is presented in Algorithm 1.

**Algorithm 1** (Clustering for Branching Structure Inference with DBSCAN)

**Input:** A set of arrivals $\mathfrak{R} = \{(s_1, t_1), \ldots, (s_n, t_n)\}$; sampling ratio function $f(x)$; perturbation parameters $r_0$, $r_1$;
**Output:** The branching structure tree $P$;

Set $\text{Clus}^{(0)} = \mathfrak{R}$ and $C_1^{(0)} = \mathfrak{R}$;
Set $k = 0$ and *StoppingList* = [*True*]
**while** *StoppingList* is not all *False*, **do**
  *StoppingList* = [ ];
  $\text{Clus}^{(k+1)}$ = [ ];
  **for** $C_j^{(k)} \in \text{Clus}^{(k)}$, $j = 1, \ldots, |\text{Clus}^{(k)}|$, **do**
    **if** $|C_j^{(k)}| > 2$, **then**
      Obtain the cluster center $c_j^{(k)}$ and record it in $P$; (skip this step if $k = 0$)
      Randomly select $f(|C_j^{(k)}|)$ pairs of points $(\mathfrak{I}_t, \mathfrak{I}_{t'}) \in C_j^{(k)} \setminus \{c_j^{(k)}\}$;
      Compute average distance of the selected pairs $d = (\sum dist(\mathfrak{I}_t, \mathfrak{I}_{t'}))/f(|C_j^{(k)}|)$;
      Apply DBSCAN to $C_j^{(k)} \setminus \{c_j^{(k)}\}$, with $MinPts = 1$ and $\epsilon = r_0 d$ when $k = 0$ or $\epsilon = d + r_1$ otherwise;
      Append clustering results to $\text{Clus}^{(k+1)}$;
      *StoppingList* append *True*;
    **else**
      **if** $|C_j^{(k)}| = 2$, **then**
        Obtain the cluster center $c_j^{(k)}$ and record it in $P$;
        *StoppingList* append *False*;
      **else**
        Record the only point $c_j^{(k)} \in C_j^{(k)}$ in $P$;
        *StoppingList* append *False*;
      **end if**
    **end if**
  **end for**
  $k = k + 1$;
**end while**
**return** $P$

## 4.2. Estimation Procedure

With the inferred branching structure, we now present the estimation procedure of the CTE method. In Section 4.2.1, we explicitly present the CTE estimator and illustrate the reason why the incorporated branching structure facilitates the model estimation. In Section 4.2.2, we discuss the asymptotic properties of the CTE estimator as the time horizon $T \to \infty$. In addition, to illustrate how the Poisson cluster representation enhances the model flexibility of a spatio-temporal self-exciting process, we present the CTE method with different triggering functions in Section 4.2.3. We note that the CTE method is not restricted to a specific conditional intensity function for the spatio-temporal self-exciting process. Nearly all the likelihood-based estimation procedures for the spatio-temporal self-exciting processes could be adapted to our CTE method. We present the incorporation of a nonparametric conditional intensity function (Li et al. 2019, Yuan et al. 2019, Fuentes-Santos et al. 2021) with CTE in the supplements.

**4.2.1. Estimation with Simplified Likelihood.** Recall from (3) that incorporating the branching structure into the log-likelihood function simplifies the analysis because the MLE then decomposes into two decoupled problems. We denote the parameters involved in the background as $\theta_1$ and the parameters involved in the triggering function as $\theta_2$. Consider a realization of the spatio-temporal self-exciting process $\mathfrak{R} = \{(s_1, t_1), (s_2, t_2), \ldots, (s_n, t_n)\}$ on $\mathcal{S} \times [0, T]$ with the inferred branching structure. The estimators are

$$\hat{\theta}_{1;T} = \arg\max_{\theta_1 \in \Theta_1} \left\{ \sum_{i:(s_i,t_i)\in\mathfrak{R}_0} \log(\mu(s_i, t_i; \theta_1)) - \int_0^T \int_{\mathcal{S}} \mu(s, t; \theta_1)\, \mathrm{d}s\, \mathrm{d}t \right\} \tag{4}$$

and

$$\hat{\theta}_{2;T} = \arg\max_{\theta_2 \in \Theta_2} \left\{ \sum_{j=1}^{n} \left( \sum_{i\in\mathfrak{T}_j} \log(g(s_i - s_j, t_i - t_j; \theta_2)) - \int_{t_j}^T \int_{\mathcal{S}} g(s - s_j, t - t_j; \theta_2)\, \mathrm{d}s\, \mathrm{d}t \right) \right\}, \tag{5}$$

where $\Theta_1$ and $\Theta_2$ are two compact sets of feasible parameters and are assumed to contain the ground-truth parameters. Compared with the original MLE (2), the explicit branching structure simplifies the log-likelihood function so that the logarithm terms do not involve any summation. Moreover, the complexity of evaluating the objective function is $\mathcal{O}(n)$[1] instead of the complexity of $\mathcal{O}(n^2)$ for evaluating (2). In addition, analytically deriving the estimators is possible in some scenarios when the intensity function has simple dependence regarding between parameters (see details in the supplements). When analytic derivation is not feasible, numerical optimization algorithms can be applied to (4) and (5), such as Newton's method, gradient descent, and Nelder-Mead method.

**4.2.2. Asymptotic Statistical Results.** Next, we provide the asymptotic properties of the CTE estimators as the time horizon $T \to \infty$. We denote the ground-truth background intensity parameter and the triggering function parameter as $\theta_1^*$ and $\theta_2^*$, respectively. First, we focus on the estimator of the background intensity function parameters $\hat{\theta}_{1;T}$ attained in (4) and present the required regularity conditions.

**Assumption 1.** *Assume that:*

1. *The background intensity function $\mu(s, t; \theta_1)$ is continuous in $\theta_1$, $\mu(s, t; \theta_1) > 0 \ \forall \theta_1 \in \Theta_1, s \in \mathcal{S}, t > 0$, and*

$$\int_0^T \int_{\mathcal{S}} \mu(s, t; \theta_1)\, \mathrm{d}s\, \mathrm{d}t < \infty; \quad 0 \leqslant T < \infty.$$

*In addition, $\forall i, j$ the partial derivatives with respect to the parameters in the background intensity function*

$$\dot{\mu}_{(i)}(s, t; \theta_1) \equiv \partial \mu(s, t; \theta_1) / \partial \theta_{1;(i)}$$

*and*

$$\ddot{\mu}_{(i,j)}(s, t; \theta_1) \equiv \partial^2 \mu(s, t; \theta_1) / \partial \theta_{1;(i)} \partial \theta_{1;(j)}$$

*exist and are continuous in $\theta_1 \ \forall \theta_1 \in \Theta_1, s \in \mathcal{S}, t > 0$. Here, $\theta_{1;(i)}$ denotes the i-th entry of $\theta_1$.*

2. *It holds that* $\forall i, j$,

$$\sup_{\theta_1 \in \Theta_1} \sup_t \int_{\mathcal{S}} \frac{(\ddot{\mu}_{(i,j)}(s,t;\theta_1))^2}{\mu(s,t;\theta_1)} \mathrm{d}s < \infty$$

*and*

$$\sup_{\theta_1 \in \Theta_1} \sup_t \int_{\mathcal{S}} \frac{(\dot{\mu}_{(i)}(s,t;\theta_1)\dot{\mu}_{(j)}(s,t;\theta_1))^2}{(\mu(s,t;\theta_1))^3} \mathrm{d}s < \infty.$$

3. *Define* $\Delta_{ij}(s,t;\theta_1) \equiv \{\dot{\mu}_{(i)}(s,t;\theta_1)\dot{\mu}_{(j)}(s,t;\theta_1)/\mu(s,t;\theta_1)\}$. *It holds that*

$$\lim_{T\to\infty} \frac{1}{T} \int_0^T \int_{\mathcal{S}} \Delta_{ij}(s,t;\theta_1^*)\, \mathrm{d}s\, \mathrm{d}t \xrightarrow{u} \sigma_{ij}(\theta_1^*)$$

*and the matrix* $\Sigma_1(\theta_1^*)$, *whose* $(i, j)$*-th element is* $\sigma_{ij}(\theta_1^*)$, *is positive definite. In addition,* $\forall c > 0$, *and* $\forall i, j$,

$$\lim_{T\to\infty} \sup_{\theta_1, \theta'_1 \in \Theta_1; \sqrt{T}|\theta'_1 - \theta_1| \leqslant c} \frac{1}{T} \int_0^T \int_{\mathcal{S}} |\Delta_{ij}(s,t;\theta_1) - \Delta_{ij}(s,t;\theta'_1)|\, \mathrm{d}s\, \mathrm{d}t \xrightarrow{u} 0.$$

*Here* $\xrightarrow{u}$ *denotes uniform convergence for* $\theta_1 \in \Theta_1$.

The first condition in Assumption 1 is satisfied in most reasonable spatio-temporal self-exciting process models. The second condition is required so that the martingale associated with the spatio-temporal Poisson process is square integrable. When the spatial space $\mathcal{S}$ is compact, the background intensity function $\mu(s,t) \geqslant \underline{\mu} > 0$, and the derivatives are bounded, this condition can be easily verified. The third condition gives appropriate growth, convergence, and continuity of the second-order derivatives of the objective function (4). Because we know exactly the form of the background intensity function, this condition can be verified analytically or numerically by calculating the integrals. In contrast, previous theoretical results require verifying regularity conditions with the conditional intensity function $\lambda(s,t;\theta)$, which is random for self-exciting processes and makes verification difficult, as documented by Schoenberg (2005).

We provide two examples of the background intensity function satisfying all the conditions in Assumption 1. The first example is

$$\mu_1(s,t;\theta_1) = \mu e^{-(x-x_c)^2-(y-y_c)^2},$$

where $\mu > 0$ and $(x_c, y_c)$ are unknown parameters to estimate. Also, the spatial space $\mathcal{S} = \mathbb{R}^2$. That is, the background intensity function does not depend on $t$ and exhibits a Gaussian decaying regarding an unknown center $(x_c, y_c)$. Similar models have been used in Zhuang et al. (2004) and Mohler (2014). The other model is

$$\mu_2(s,t;\theta_1) = C_1 + \sin(\alpha t),$$

where $C_1 > 1$ is the unknown parameter to estimate and $\alpha$ is known. This background intensity $\mu_2$ exhibits a cyclical behavior regarding time $t$ (Lee et al. 1991, Kuhl and Wilson 2001). We consider a compact spatial space $\mathcal{S} = [0, X_1] \times [0, Y_1]$. The detailed reasons why these two models satisfy Assumption 1 are included in the supplements.

**Theorem 1.** *Suppose that the immigration* $\mathfrak{N}_0$ *is correctly specified by the clustering algorithm and Assumption 1 holds. The background intensity parameter estimator* $\hat{\theta}_{1;T}$ *in* (4) *is consistent and asymptotically normal as* $T \to \infty$. *That is,*

$$\lim_{T\to\infty} \hat{\theta}_{1;T} \xrightarrow{P} \theta_1^*$$

*and*

$$\lim_{T\to\infty} \sqrt{T}(\hat{\theta}_{1;T} - \theta_1^*) \xrightarrow{D} \mathcal{N}(0, \{\Sigma_1(\theta_1^*)\}^{-1}),$$

*where* $\theta_1^*$ *denotes the ground-truth parameter of the background intensity function, "*$\xrightarrow{P}$*" is "convergence in probability," and "*$\xrightarrow{D}$*" is "convergence in distribution."*

In Theorem 1, we assume that the initial branching structure (in terms of immigration) inferred by the clustering algorithm is exactly the ground-truth branching structure. However, in some scenarios, the clustering algorithm may lead to misspecification of the immigration, which will further influence the estimation. We denote the set of misspecified immigrants as $\mathfrak{N}_0^m$, which contains (1) the points that are triggered by previous points but are classified as immigrants and (2) the immigrants that are classified as the points triggered by the previous points. We note that $|\mathfrak{N}_0^m|$ is nondecreasing with the time horizon $T$. We have the following corollary for the asymptotic behavior of $\hat{\theta}_{1;T}$ when misspecification of immigration exists.

**Corollary 1.** *Suppose that* $\lim_{T\to\infty}|\mathfrak{N}_0^m|/T \to 0$, *that* $\ddot{\mu}_{(i,j)}(s,t;\theta_1^*)/\mu(s,t;\theta_1^*)$ *and* $\dot{\mu}_{(i)}(s,t;\theta_1^*)\dot{\mu}_{(j)}(s,t;\theta_1^*)/(\mu(s,t;\theta_1))^2$ *are bounded* $\forall s \in \mathcal{S}, t \in [0,T]$, *and that Assumption 1 holds. The consistency and the asymptotic normality of the estimator* $\hat{\theta}_{1;T}$ *based on the misspecified immigration still holds as in Theorem 1.*

In other words, when the proportion of misspecified immigrants approaches zero as the time horizon approaches infinity, the asymptotic behavior of the background intensity estimator $\hat{\theta}_{1;T}$ remains as if the specification of the immigration was correct. We provide a procedure to justify the condition $\lim_{T\to\infty}|\mathfrak{N}_0^m|/T \to 0$; see details in the supplements.

In terms of the triggering function, the maximum likelihood estimation (5) is based on multiple sample paths of Poisson processes with intensity function $g(s,\tau)$. Indeed, the number of sample paths of the Poisson processes up to time $t$ is approximately $\int_0^t\int_{\mathcal{S}}\mu(s,\tau)\mathrm{d}s\mathrm{d}\tau/(1-\int_{\mathcal{S}}\int_0^\infty g(s,\tau)\mathrm{d}s\mathrm{d}\tau)$. From the *complete independence* property (see Resnick 1992) of the Poisson process, the superposition of multiple sample paths of a Poisson process with intensity function $g(s,\tau)$ is distributionally equivalent to a sample path of a Poisson process with intensity function multiplied by the number of sample paths. Consequently, if the rescaled triggering function $g^*(s,t) = g(s,t)\int_0^t\int_{\mathcal{S}}\mu(s,\tau)\mathrm{d}s\mathrm{d}\tau/(1-\int_{\mathcal{S}}\int_0^\infty g(s,\tau)\mathrm{d}s\mathrm{d}\tau)$ satisfies the same conditions in Assumption 1 as $\mu(s,t)$, and if the branching structure inference attained by the clustering algorithm is correct, then the consistency and the asymptotic normality of $\hat{\theta}_{2;T}$ are also guaranteed; that is, analogs of Theorem 1 and Corollary 1 hold. Their detailed proofs are included in the supplements.

Finally, the variance-covariance matrix provided by the asymptotic normality helps to quantify the estimation uncertainty. We note that because of the third condition in Assumption 1, the variance-covariance matrix of CTE can be derived by integrating deterministic functions, where the estimated value approximates the ground-truth parameter. In comparison, for the previous MLE estimator (Rathbun 1996), the variance-covariance matrix estimator involves not only the parameter estimator $\hat{\theta}$ but also the estimated conditional intensity function $\lambda(s,t;\theta)$, which further depends on the observed data. That is, prior results require more approximation and, therefore, involve more uncertainty. We also provide a parametric bootstrap procedure to quantify the estimation uncertainty; see details in the supplements.

#### 4.2.3. Estimation with Different Triggering Functions.

The Poisson cluster process representation enables the self-exciting process to possess different triggering functions in different generations (Fierro et al. 2015). That is, the $(m+1)$-th generation $\mathfrak{N}_{m+1}$ is triggered by the $m$-th generation $\mathfrak{N}_m$ with the triggering function $g_m$. The model flexibility is enhanced by allowing the triggering functions $\{g_m\}_{m=0}^\infty$ to possess different values of parameters or even different analytic forms.

With the explicit branching structure, where the attribution of each point to its appropriate cluster is exact, the CTE method is capable of estimating the triggering functions for different pairs of generations. Specifically, the parameters of the triggering function that incites the $(m+1)$-th generation (by the $m$-th generation) are estimated by maximizing the log-likelihood

$$\ell_m(\theta_{2;m}) = \sum_{j:(s_j,t_j)\in\mathfrak{N}_m}\left(\sum_{i\in\mathfrak{D}_j}\log(g_m(s_i-s_j,t_i-t_j;\theta_{2;m})) - \int_{t_j}^T\int_{\mathcal{S}} g_m(s-s_j,t-t_j;\theta_{2;m})\mathrm{d}s\mathrm{d}t\right).$$

Here, $\mathfrak{D}_j$ denotes the points that are directly triggered by $\mathfrak{I}_j$, and $\mathfrak{N}_m$ denotes the $m$-th generation. In addition to estimating the parameters of the triggering functions separately for different generations, the CTE method can also handle the model where there is a trend in the triggering effects. For example, the triggering function between the $m$-th and $(m+1)$-th generations may possess the form

$$g_m(s,t;\theta_2) = g(s,t;\theta_2)e^{-\gamma m}, \quad \gamma > 0,$$

so the triggering effects are decaying as the generations increase. In this model, the parameters $(\theta_2,\gamma)$ are

estimated by maximizing the log-likelihood function

$$\ell(\theta_2;\gamma)=\sum_{m=0}\left(\sum_{j:(s_j,t_j)\in\mathfrak{R}_m}\left(\sum_{i\in\mathfrak{D}_j}\log(g_m(s_i-s_j,t_i-t_j;\theta_2))-\int_{t_j}^{T}\int_{\mathcal{S}}g_m(s-s_j,t-t_j;\theta_2)\mathrm{d}s\,\mathrm{d}t\right)\right).$$

This is an example of the model flexibility of the CTE method.

In addition to the frequentist view, where the MLE is employed to estimate the parameters, the proposed CTE method can also be used with the Bayesian inference of the spatio-temporal self-exciting processes. We present the discussion in the supplements.

## 5. Experiments

We conduct numerical experiments to demonstrate the effectiveness and superiority of the proposed clustering-then-estimation (CTE) method compared with (1) the maximum likelihood estimation (MLE) of the original log-likelihood function using different numerical optimization algorithms, (2) the expectation-maximization (EM) method, and (3) the EM-declustering method that samples a deterministic branching structure in each iteration of EM. The details of these baseline approaches are included in the supplements. Below, we first apply the estimation methods to the synthetic data generated by simulation experiments and then apply the CTE method to real-world data. The experiments were run with Python on an Intel i-7 CPU with a clock speed of 2.60GHz. The implementation of our numerical experiments can be found in Zhang et al. (2024).

### 5.1. Experiments on Synthetic Data

Our experiments are based on synthetic data simulated with the conditional intensity function

$$\lambda(s,t)=\mu+\sum_{t_i<t}\alpha e^{-\beta(t-t_i)-\frac{1}{2}\left(\frac{(x-x_j)^2}{\sigma_x^2}+\frac{(y-y_j)^2}{\sigma_y^2}\right)}, \tag{6}$$

where $\mu,\alpha,\beta,\sigma_x,\sigma_y>0$ and $2\pi\sigma_x\sigma_y\alpha<\beta$. We consider the two-dimensional space $\mathcal{S}=[0,10]\times[0,10]$ with time horizon $T=10$. The detailed calculation procedures of the estimators (CTE and EM) for this model are in the supplements. The simulation algorithm is based on the Poisson cluster representation of the spatio-temporal self-exciting process; see details in the supplements. The detailed procedures of simulating Poisson processes can be found in Pasupathy (2010) and Saltzman et al. (2012). We compare the estimated parameters with the ground-truth parameter set and the branching structure inferred by the CTE method and the EM method with the ground-truth branching structure. The metric utilized for the comparison between branching structures is the tree-edit distance (TD), which was proposed in Zhang and Shasha (1989)[2] and is described in the supplements. We note that the branching structure derived from the EM method (as well as EM-declustering) is the probability of the attribution of each point, not an explicit deterministic tree. Therefore, to evaluate the branching structure inference of the EM method, we should apply the stochastic declustering procedure introduced in Zhuang et al. (2002) to get a batch of sample trees (size of 30 in our experiments) from the stochastic branching structure. We then take the sample mean of the tree distances to estimate the distance between the stochastic branching structure and the ground-truth branching structure. We note that MLE with the original likelihood function does not generate the inference of branching structure. Each set of experimental results in this section is based on 30 simulation experiments.

**5.1.1. General Comparison.** We compare the CTE method and baseline approaches in three ways: (1) parameter estimation (± standard deviation), (2) branching structure inference, and (3) computation time.[3] The EM method requires an initialization of the branching structure; we set $\mathbb{P}(u_i=j)=1/i$ for $j=0,1,\ldots,i-1$.

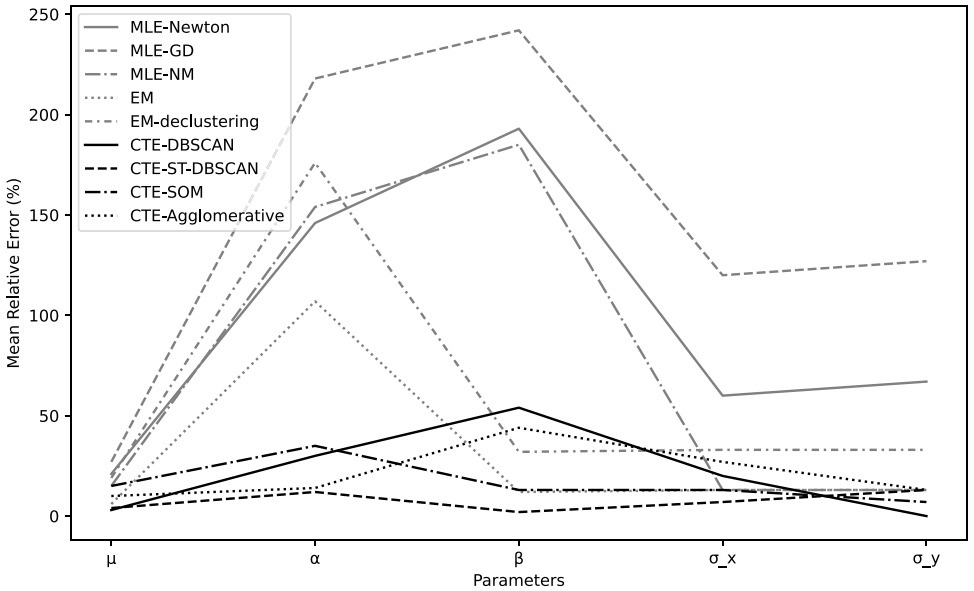
We use the following clustering algorithms with the CTE method with different clustering algorithms: (1) DBSCAN, (2) spatio-temporal DBSCAN (ST-DBSCAN) (Birant and Kut 2007), (3) the self-organizing mapping (SOM) (Kohonen 1990), and (4) the agglomerative hierarchical clustering (Nielsen 2016). The description of the latter three clustering algorithms is given in the supplements. In terms of the numerical optimization algorithms used to maximize the original log-likelihood function (2), we employ (1) Netwton's method, (2) the gradient descent (GD) method, and (3) the Nelder–Mead (NM) method. Experimental results are presented in Table 2. We also plot the mean relative error $|(\hat{\theta}-\theta^*)/\theta^*|\times 100\%$ of the estimated parameters in Figure 2, where $\hat{\theta}$ denotes the estimator of a general scalar parameter and $\theta^*$ is the corresponding ground-truth value. The proposed CTE method with different clustering algorithms is shown in black, whereas the compared baseline methods are in gray. The experimental results provide the following insights.

**Table 2.** General Comparisons on (1) Parameter Estimation, (2) Tree-Edit Distance for Branching Structure Inference (TD), and (3) Computation Time Between CTE and Baseline Methods

| Simulation setting<br>Methods | $\mu = 0.02$,<br>TD | $\alpha = 10, \beta = 5$,<br>$\mu$ | $\sigma_x = 0.2$,<br>$\alpha$ | $\sigma_y = 0.2$<br>$\beta$ | $\sigma_x$ | $\sigma_y$ | Time |
|---|---|---|---|---|---|---|---|
| MLE-Newton | N/A | 0.023 ± 0.007 | 12.65 ± 0.51 | 5.92 ± 0.35 | **0.20 ± 0.02** | 0.21 ± 0.02 | 82.90 |
| MLE-GD | N/A | 0.024 ± 0.009 | 13.44 ± 0.57 | 6.64 ± 0.43 | 0.19 ± 0.01 | 0.21 ± 0.02 | 129.31 |
| MLE-NM | N/A | 0.023 ± 0.008 | 12.03 ± 0.54 | 5.68 ± 0.33 | 0.21 ± 0.01 | 0.20 ± 0.01 | 84.47 |
| EM | 1,070.0 | **0.020 ± 0.007** | 2.04 ± 0.34 | 1.93 ± 0.29 | 0.18 ± 0.02 | 0.18 ± 0.02 | 206.99 |
| EM-declustering | 1,257.5 | 0.031 ± 0.010 | 1.98 ± 0.39 | 4.38 ± 0.29 | 0.18 ± 0.03 | 0.17 ± 0.02 | 131.51 |
| CTE-DBSCAN | 427.0 | 0.018 ± 0.004 | 8.48 ± 0.30 | 5.33 ± 0.21 | 0.22 ± 0.01 | **0.20 ± 0.01** | 8.45 |
| CTE-ST-DBSCAN | **402.6** | 0.017 ± 0.004 | 10.10 ± 0.28 | 4.53 ± 0.19 | 0.22 ± 0.01 | 0.19 ± 0.01 | **7.89** |
| CTE-SOM | 456.7 | 0.018 ± 0.003 | **9.91 ± 0.30** | **5.11 ± 0.20** | 0.22 ± 0.01 | 0.19 ± 0.01 | 10.43 |
| CTE-Agglomerative | 460.0 | 0.022 ± 0.004 | 8.04 ± 0.29 | 4.35 ± 0.19 | 0.21 ± 0.01 | 0.19 ± 0.01 | 8.27 |
| Simulation setting<br>Methods | $\mu = 0.05$,<br>TD | $\alpha = 10, \beta = 5$,<br>$\mu$ | $\sigma_x = 0.2$,<br>$\alpha$ | $\sigma_y = 0.2$<br>$\beta$ | $\sigma_x$ | $\sigma_y$ | Time |
| MLE-Newton | N/A | 0.058 ± 0.012 | 13.10 ± 0.66 | 5.40 ± 0.45 | 0.23 ± 0.03 | 0.24 ± 0.03 | 323.17 |
| MLE-GD | N/A | 0.060 ± 0.015 | 13.54 ± 0.73 | 5.29 ± 0.54 | 0.30 ± 0.03 | 0.29 ± 0.04 | 487.25 |
| MLE-NM | N/A | 0.056 ± 0.011 | 12.23 ± 0.54 | 5.30 ± 0.43 | 0.23 ± 0.03 | 0.22 ± 0.03 | 296.90 |
| EM | 1,980.0 | 0.055 ± 0.012 | 6.86 ± 0.48 | 4.04 ± 0.37 | 0.23 ± 0.03 | 0.22 ± 0.03 | 317.38 |
| EM-declustering | 2,343.3 | 0.062 ± 0.017 | 5.65 ± 0.60 | 7.19 ± 0.45 | 0.17 ± 0.04 | 0.15 ± 0.05 | 263.23 |
| CTE-DBSCAN | 1,044.0 | 0.053 ± 0.005 | **9.62 ± 0.37** | 6.10 ± 0.29 | **0.22 ± 0.02** | 0.22 ± 0.03 | **19.68** |
| CTE-ST-DBSCAN | 1,127.3 | **0.051 ± 0.004** | 8.82 ± 0.39 | **5.13 ± 0.28** | 0.23 ± 0.02 | 0.21 ± 0.02 | 20.41 |
| CTE-SOM | **1,090.0** | 0.055 ± 0.005 | 7.91 ± 0.41 | 5.46 ± 0.31 | 0.18 ± 0.03 | **0.20 ± 0.02** | 35.78 |
| CTE-agglomerative | 1,276.0 | 0.056 ± 0.004 | 8.83 ± 0.38 | 5.37 ± 0.31 | 0.23 ± 0.02 | 0.23 ± 0.02 | 24.08 |
| Simulation setting<br>Methods | $\mu = 0.1$,<br>TD | $\alpha = 5, \beta = 2.5$,<br>$\mu$ | $\sigma_x = 0.15$,<br>$\alpha$ | $\sigma_y = 0.15$<br>$\beta$ | $\sigma_x$ | $\sigma_y$ | Time |
| MLE-Newton | N/A | 0.121 ± 0.019 | 12.32 ± 0.87 | 7.32 ± 0.65 | 0.24 ± 0.04 | 0.25 ± 0.04 | 897.63 |
| MLE-GD | N/A | 0.127 ± 0.023 | 15.88 ± 0.98 | 8.54 ± 0.73 | 0.33 ± 0.05 | 0.34 ± 0.05 | 1,351.29 |
| MLE-NM | N/A | 0.115 ± 0.017 | 12.70 ± 0.83 | 7.12 ± 0.60 | 0.23 ± 0.04 | 0.23 ± 0.04 | 813.82 |
| EM | 2,908.0 | 0.106 ± 0.014 | 10.36 ± 0.64 | 2.19 ± 0.49 | 0.17 ± 0.04 | 0.17 ± 0.04 | 506.86 |
| EM-declustering | 3,356.7 | 0.119 ± 0.024 | 13.81 ± 0.81 | 3.31 ± 0.62 | 0.20 ± 0.04 | 0.20 ± 0.04 | 430.49 |
| CTE-DBSCAN | 1,376.0 | **0.103 ± 0.010** | 6.50 ± 0.49 | 3.86 ± 0.36 | 0.17 ± 0.03 | **0.15 ± 0.03** | 22.67 |
| CTE-ST-DBSCAN | **792.0** | 0.096 ± 0.009 | **4.40 ± 0.46** | **2.55 ± 0.35** | **0.14 ± 0.03** | 0.14 ± 0.03 | **21.94** |
| CTE-SOM | 1,088.0 | 0.115 ± 0.010 | 3.23 ± 0.53 | 2.83 ± 0.40 | 0.17 ± 0.03 | 0.16 ± 0.04 | 50.36 |
| CTE-agglomerative | 1,334.0 | 0.090 ± 0.009 | 5.69 ± 0.50 | 3.60 ± 0.37 | 0.18 ± 0.03 | 0.17 ± 0.03 | 29.91 |

*Note.* The value representing the best performance is highlighted in bold.

**Figure 2.** Mean Relative Error of the Estimated Parameters with Different Estimation Methods




*Note.* The setting is $\mu = 0.1, \alpha = 5, \beta = 2.5, \sigma_x = 0.15, \sigma_y = 0.15$.

1. The EM method generally performs better than the original MLE in terms of model estimation, as documented by previous literature (Veen and Schoenberg 2008). The CTE methods with different clustering algorithms outperform the classical EM method in terms of (1) more accurate parameter estimation, (2) smaller TDs (indicating more accurate branching structure inference), and (3) less computation time.

2. The baseline methods, which include both MLE and EM, can relatively accurately estimate the background intensity $\mu$, but their performance on the estimation of triggering functions is suboptimal. Compared with EM, the EM-declustering method is more computationally efficient. However, the EM-declustering method suffers from lower accuracy in both branching structure inference and model estimation. It also exhibits a higher standard deviation, which aligns with the fact that EM-declustering can be regarded as a Monte Carlo Markov Chain approximation for the EM method (Li et al. 2019).

3. Among all clustering algorithms, ST-DBSCAN achieves the most stable and effective performance across all sets of experiments because ST-DBSCAN has two separate thresholds, a spatial threshold ($\epsilon_1$) and a temporal threshold ($\epsilon_2$), to distinguish the time dimension from the spatial dimensions.

4. As the background intensity $\mu$ increases, all the baseline approaches' errors increase because of the increasing complexity of evaluating the log-likelihood functions. In contrast, the performance of CTE remains acceptable, and its superiority becomes more significant for larger $\mu$.

Differences in the computation time for CTE methods result from different clustering algorithms. After the branching structure is inferred, the estimation procedure (maximization of the likelihood) of the CTE method is accomplished efficiently.

**5.1.2. Risk of Model Misspecification.** Recall that for the CTE method, the branching structure inference is separated from the model specification. In this section, we show through experiments that CTE suffers less from model misspecification than the EM method. That is, we retain the model assumption (6) for estimation while using a different spatio-temporal self-exciting process model to generate synthetic data. Specifically, we use

$$\lambda(s,t) = \mu(s,t) + \sum_{t_i < t} e^{-\gamma \|\Delta_i(s,t)\|_2}$$

to generate data, where

$$\Delta_i(s,t) = (x - x_i, y - y_i, t - t_i)^\top,$$

$$\mu(s,t) = \begin{cases} \mu_1, & t \in [0, T/2], s \in \mathcal{S} \\ \mu_2, & t \in [T/2, T], s \in \mathcal{S}, \end{cases}$$

and $\gamma$ denotes the rate of the triggering effects. For the branching structure inference, CTE directly applies the clustering algorithms (DBSCAN as a representative in this section) to the data set, whereas the EM method relies on the model assumption (6). We display the Tree-edit Distance (TD) for the methods in Table 3. We also perform the estimation steps for the CTE method with the parametric model (6). Because we cannot compare the estimated parameters with the ground-truth parameters, we instead present the attained values of log-likelihood functions in Table 3.

The experimental results presented in Table 3 indicate that the CTE method suffers less from the model misspecification with higher log-likelihood function values and lower TDs.

**5.1.3. Sensitivity to Branching Structure Inference.** In this section, we show through experiments that the results provided by the CTE method depend on the inferred branching structure, which further depends on the hyperparameter of the clustering algorithms. We present a procedure to determine the hyperparameter in the supplements. In this procedure, each sample path of the observed spatio-temporal self-exciting process is divided into two periods, with one used for estimating the model and the other for validating the estimated model.

We also propose a methodology to alleviate the error of the CTE method when the branching structure is not accurately inferred by the clustering algorithm. In particular, we take the attained CTE estimator as the

**Table 3.** Tree-Edit Distance and Likelihood of Misspecified Models

| Simulation setting | | | EM | | EM-declustering | | CTE-DBSCAN | |
|---|---|---|---|---|---|---|---|---|
| $\mu_1$ | $\mu_2$ | $\gamma$ | TD | Log-Likelihood | TD | Log-likelihood | TD | Log-likelihood |
| 0.05 | 0.03 | 0.3 | 955.3 | −560.56 | 1,145.3 | −601.35 | **542.0** | **−345.17** |
| 0.05 | 0.08 | 0.3 | 2,252.5 | −842.29 | 2,637.8 | −1,039.67 | **1,013.0** | **−488.80** |
| 0.08 | 0.03 | 0.3 | 1,242.0 | −668.18 | 1,787.5 | −704.29 | **661.2** | **−415.39** |

*Note.* The value representing the best performance is highlighted in bold.

**Table 4.** Branching Structure Inference and Parameter Estimation by CTE-DBSCAN with Different Clustering Hyperparameters

| Simulation setting<br>Methods | $\mu = 0.05$,<br>TD | $\alpha = 5, \beta = 2.5$,<br>$\mu$ | $\sigma_x = 0.15$,<br>$\alpha$ | $\sigma_y = 0.15$<br>$\beta$ | $\sigma_x$ | $\sigma_y$ |
|---|---|---|---|---|---|---|
| CTE-ground truth | 0 | $0.051 \pm 0.001$ | $6.70 \pm 0.21$ | $3.13 \pm 0.16$ | $0.14 \pm 0.01$ | $0.15 \pm 0.01$ |
| MLE | N/A | $0.056 \pm 0.010$ | $7.37 \pm 0.49$ | $4.04 \pm 0.41$ | $0.17 \pm 0.03$ | $0.17 \pm 0.03$ |
| EM-random initialization | 1,316.6 | $0.055 \pm 0.011$ | $7.15 \pm 0.43$ | $4.76 \pm 0.39$ | $0.16 \pm 0.03$ | $0.16 \pm 0.03$ |
| EM-ground truth | 52.5 | $0.050 \pm 0.002$ | $6.70 \pm 0.29$ | $3.13 \pm 0.21$ | $0.14 \pm 0.01$ | $0.15 \pm 0.01$ |
| CTE-DBSCAN[1] | 687.0 | $0.052 \pm 0.003$ | $6.68 \pm 0.32$ | $4.51 \pm 0.27$ | $0.17 \pm 0.02$ | $0.18 \pm 0.02$ |
| CTE-DBSCAN[1]-EM | 677.5 | $0.051 \pm 0.002$ | $6.67 \pm 0.30$ | $4.51 \pm 0.27$ | $0.17 \pm 0.02$ | $0.18 \pm 0.02$ |
| CTE-DBSCAN[2] | 709.0 | $0.052 \pm 0.004$ | $9.23 \pm 0.38$ | $4.99 \pm 0.32$ | $0.16 \pm 0.02$ | $0.16 \pm 0.02$ |
| CTE-DBSCAN[2]-EM | 702.3 | $0.051 \pm 0.004$ | $9.16 \pm 0.34$ | $4.82 \pm 0.30$ | $0.16 \pm 0.03$ | $0.15 \pm 0.03$ |
| CTE-DBSCAN[3] | 796.0 | $0.037 \pm 0.005$ | $3.22 \pm 0.45$ | $1.94 \pm 0.40$ | $0.18 \pm 0.03$ | $0.19 \pm 0.03$ |
| CTE-DBSCAN[3]-EM | 755.5 | $0.052 \pm 0.004$ | $4.55 \pm 0.43$ | $2.52 \pm 0.41$ | $0.16 \pm 0.02$ | $0.16 \pm 0.03$ |

*Note.* The EM method takes results attained by associated CTE method as starting points and serves as calibration.

initialization of the EM method. In this way, by iterating the E-step and the M-step, the inferred branching structure will be modified toward the ground-truth branching structure. We write this methodology as CTE-EM. We present the results in Table 4, where we specifically select the DBSCAN algorithm as the clustering algorithm used in the CTE method. We perform the DBCSCAN algorithm with three sets of hyperparameters,[4] and the associated "-EM" indicates that the EM method takes the branching structure attained by the CTE method as the initialization. We also present the results when ground-truth branching structure is taken into the CTE method and EM method, respectively. The experiments provide the following insights:

1. For the CTE method, there exists a positive correlation between the accuracy of the inferred branching structure (indicated by smaller tree-edit distances) and the accuracy of the corresponding model estimation. Notably, the accuracy peaks when the ground-truth branching structure is incorporated into the CTE method.
2. The implementation of the EM method following the CTE method enhances model estimation. However, when the branching structure inferred by the CTE method is sufficiently accurate, the enhancement offered by the EM method is minimal, as evidenced by the ground-truth case. Given that the EM method is sensitive to the initialization of the branching structure, the results of the CTE-EM method are also contingent on the branching structure inferred by the clustering algorithm. Overall, the outcomes derived solely from implementing the CTE method outperform those from the existing approaches.

#### 5.1.4. Different Triggering Functions.

In this section, we show through experiments that the CTE method is capable of estimating the spatio-temporal self-exciting process with different triggering functions, which is infeasible for either the original MLE or the EM method. Theoretical support of the self-exciting process with different triggering functions can be found in Mehrdad and Zhu (2014) and Fierro et al. (2015). The experimental results presented in Table 5 indicate that the CTE method is capable of handling different triggering functions for different generations. Also, the CTE method with the ST-DBSCAN algorithm generally outperforms the CTE method with DBSCAN.

### 5.2. Experiments on Real-World Data

In this section, we further illustrate the effectiveness of the CTE method by experimenting with four real-world data sets: (1) 911-calls, (2) earthquakes, (3) bike-sharing services, and (4) online retail transactions. For ease of comparison, we normalize the space region of both data sets to the same space, $\mathcal{S} \times [0, T]$, where $T = 10$ and $\mathcal{S} = [0, 10] \times [0, 10]$.

**Table 5.** Parameter Estimation with Different Triggering Functions ($\mu = 0.050$)

| Simulation setting | | | | | CTE-DBSCAN | | | | CTE-ST-DBSCAN | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Generation | $\alpha$ | $\beta$ | $\sigma_x$ | $\sigma_y$ | $\alpha$ | $\beta$ | $\sigma_x$ | $\sigma_y$ | $\alpha$ | $\beta$ | $\sigma_x$ | $\sigma_y$ |
| 1st-G | 8 | 4 | 0.20 | 0.20 | 6.714 | 3.580 | 0.241 | 0.217 | 7.081 | 4.339 | 0.232 | 0.250 |
| 2nd-G | 8 | 3 | 0.22 | 0.22 | 6.183 | 1.745 | 0.253 | 0.230 | 6.711 | 2.109 | 0.304 | 0.288 |
| 3rd-G | 6 | 2 | 0.26 | 0.26 | 5.575 | 0.980 | 0.304 | 0.330 | 4.010 | 1.107 | 0.263 | 0.290 |
| 4th-G | 10 | 5 | 0.28 | 0.28 | 5.406 | 3.220 | 0.331 | 0.318 | 6.325 | 3.523 | 0.301 | 0.287 |

The spatio-temporal self-exciting process we consider here is a nonparametric model,

$$\mu(s,t) = \alpha u(s)v(t), \quad g(s,t) = \beta h(s,t),$$

where $u(s), v(t)$ and $h(s,t)$ are scale parameters that are estimated by nonparametric methods (e.g., kernel density estimation), and $\alpha$ and $\beta$ are estimated by maximum likelihood estimation. The detailed descriptions of CTE with a nonparametric model are in the supplements. In our experiments, we specifically select Gaussian kernel density estimation as the nonparametric method. In terms of the clustering algorithms used in the CTE method, we employ the ST-DBSCAN algorithm because this clustering algorithm achieves the most stable and effective performance in synthetic data experiments. We also perform the EM method as the baseline approach. We record the value of the log-likelihood functions with the estimated model, which is a commonly selected comparison metric with real data (Zhu et al. 2021a).

**5.2.1. 911-Calls Historic Data.**[5] 911 calls follow a particular causal relationship with each other. Each 911-call report is associated with the time (accurate to the second) and the geolocation (latitude and longitude), indicating when and where the call occurred. We extract 1,000 reported calls from the data set for modeling.

**5.2.2. Italy's Earthquake in 2016.**[6] This data set contains data about earthquakes that hit the center of Italy between August 24 and November 30, 2016, collected by the Italian Earthquakes National Center. The data set contains the time, latitude, longitude, depth/km, and magnitude of 8,087 earthquake events. We selected 1,000 relatively significant (in magnitude) events for testing our CTE method to mine the spatio-temporal patterns and estimate parameters.

**5.2.3. Citi Bike Data.**[7] The Citi Bike data set contains the transaction details of the bike-sharing service in New York City. We select the first 1,000 records to test the models' performance. The start times, start station latitudes, and start station longitudes are inputs to the model as time $t$ spatial coordinates $x$ and $y$.

**5.2.4. Online Retail Transactions.**[8] This is a transnational data set containing all the transactions occurring between January 12, 2010, and September 12, 2011, for a U.K.-based and registered non-store online retail company. The company mainly sells all-occasion gifts, and many customers are wholesalers. Each transaction includes the time, the country, the customer identity number, and the unit price and quantity of each good. The information vector, excluding the time, is projected into two-dimensional Euclidean space by Principal Component Analysis (Jolliffe and Cadima 2016). This two-dimensional vector represents the spatial information of the point.

The results in Table 6 indicate that the CTE method attains larger log-likelihoods on all data sets compared with the EM method, indicating a better fit. Moreover, the superiority of the CTE method over the classical EM method is much more significant in real-data experiments than in synthetic data experiments (Section 5.1).

We now include some further analysis based on the estimated model.

1. **Temporal analysis:** In a spatio-temporal model, scrutinizing changes over time can be advantageous because it may reveal cyclical patterns, emerging trends, or abrupt shifts. Such temporal analysis paves the way for a more comprehensive understanding of the process dynamics. For instance, after obtaining an accurate estimated model of earthquake activity with the spatio-temporal self-exciting process, we can study the temporal patterns, seeking to identify any periodic characteristics, which may facilitate the supply chain design considering disruptions (Cui et al. 2010, Yamin et al. 2022).

**Table 6.** Real-World Data Experiment Results of Earthquake (2 Upper Rows), 911-Call (2 Middle-Upper Rows), Citi Bike (2 Middle-Lower Rows), and Online Retails (2 Lower Rows)

| Estimation/Metric | EM | CTE-ST-DBSCAN |
|---|---|---|
| Parameters$(\alpha,\beta)$ | 255.23, 0.76 | 151.81, 0.85 |
| Log-likelihood | 513.73 | **764.57** |
| Parameters$(\alpha,\beta)$ | 212.85, 0.80 | 263.08, 0.74 |
| Log-likelihood | 708.21 | **1,164.38** |
| Parameters$(\alpha,\beta)$ | 235.06, 0.71 | 132.61, 0.88 |
| Log-likelihood | 1,036.95 | **1,261.91** |
| Parameters$(\alpha,\beta)$ | 179.80, 0.15 | 182.30, 0.83 |
| Log-likelihood | 1,081.24 | **1,465.61** |

*Note.* The value representing the best performance is highlighted in bold.

2. **Spatial analysis:** Assessing the spatial dimension of the estimated model also provides insight. Considering the Citi Bike data set, it becomes evident that specific spatial regions have considerable impacts. Moreover, some areas display notable heterogeneity. These insights serve to aid the companies in refining their bike provisioning policies (Liu et al. 2018).

3. **Future prediction and decision making:** The estimated model can be employed to make predictions about the spatial and temporal information of upcoming arrivals. Moreover, spatio-temporal self-exciting processes serve as the input to other stochastic systems, so accurately modeling those inputs can lead to better decisions in those systems. For instance, a spatio-temporal self-exciting process model for online retail transactions helps to assess various shipping and warehousing strategies in terms of profitability and environmental sustainability. In terms of methodology, the estimated spatio-temporal self-exciting process model can be used, for example, in simulation optimization; see Jian and Henderson (2015) and Wang et al. (2022), the details of which are beyond the scope of this work.

## 6. Conclusion

In this paper, we present an estimation procedure for the spatio-temporal self-exciting processes called "clustering-then-estimation" (CTE). In our methodology, we first apply the density-based clustering algorithm to the data set, directly inferring the branching structure. Then, we obtain estimators of parameters by maximizing the likelihood function simplified by the inferred branching structure. We show the consistency and asymptotic normality of the CTE estimators. We conduct experiments to compare the CTE method with baseline approaches. Numerical results on both synthetic data and real-world data indicate that the CTE method demonstrates (1) better accuracy on the model estimation and branching structure inference, (2) less risk of model misspecification, (3) higher efficiency in practice without the necessity of recursively solving optimization problems, and (4) more flexibility in terms of capturing different triggering effects between different pairs of generations.

In future work, we plan to consider the design of clustering algorithms specific to the structure of spatio-temporal self-exciting processes. Existing clustering algorithms largely assume that the cluster is a "ball" around a center, whereas we observe that the cluster for a spatio-temporal self-exciting process exhibits a "cone" shape. Thus, the adjustment of clustering algorithms to take this into account may improve the branching structure inference and model estimation. In addition, other potential future directions include (1) generalizing our approach to multivariate self-exciting processes, (2) taking the estimation of marks of each arrival into consideration, and (3) incorporating a neural network-based conditional intensity function.

### Acknowledgments

Haoting Zhang and Donglin Zhan contributed equally to the manuscript.

### Endnotes

[1] The complexity $\mathcal{O}(n)$ results from the fact that there are $|\mathfrak{N}_0|$ summation terms in (4) and $2n - |\mathfrak{N}_0|$ summation terms in (5).

[2] The implementation of calculating TD between two tree-structures is in https://zhang-shasha.readthedocs.io.

[3] The computation time of CTE includes both times for performing the clustering algorithm and for estimating the parameters through the likelihood functions.

[4] DBSCAN[1] uses $r_0 = 0.2$ and $r_1 = 0.64$. DBSCAN[2] uses $r_0 = 0.3$ and $r_1 = 0.64$. DBSCAN[3] uses $r_0 = 0.3$ and $r_1 = 0.5$.

[5] https://www.kaggle.com/mchirico/montcoalert/notebooks.

[6] https://www.kaggle.com/blackecho/italy-earthquakes.

[7] https://www.kaggle.com/datasets/sujan97/citibike-system-data.

[8] https://archive.ics.uci.edu/dataset/352/online+retail#dataset.

### References

Adikari S, Dutta K (2019) A new approach to real-time bidding in online advertisements: Auto pricing strategy. *INFORMS J. Comput.* 31(1):66–82.

Ahn D, Shin D, Zeevi A (2023) Feature misspecification in sequential learning problems. Research paper, Columbia Business School, New York.

Balderama E, Schoenberg FP, Murray E, Rundel PW (2012) Application of branching models in the study of invasive species. *J. Amer. Statist. Assoc.* 107(498):467–476.

Bichler M, Hammerl A, Morrill T, Waldherr S (2021) How to assign scarce resources without money: Designing information systems that are efficient, truthful, and (pretty) fair. *Inform. Systems Res.* 32(2):335–355.

Birant D, Kut A (2007) St-dbscan: An algorithm for clustering spatial–temporal data. *Data Knowl. Eng.* 60(1):208–221.

Bollapragada R, Li Y, Rao US (2006) Budget-constrained, capacitated hub location to maximize expected demand coverage in fixed-wireless telecommunication networks. *INFORMS J. Comput.* 18(4):422–432.

Brice P, Jiang W, Wan G (2011) A cluster-based context-tree model for multivariate data streams with applications to anomaly detection. *INFORMS J. Comput.* 23(3):364–376.

Brown L, Gans N, Mandelbaum A, Sakov A, Shen H, Zeltyn S, Zhao L (2005) Statistical analysis of a telephone call center: A queueing-science perspective. *J. Amer. Statist. Assoc.* 100(469):36–50.

Chen X (2021) Perfect sampling of Hawkes processes and queues with Hawkes arrivals. *Stoch. Syst.* 11(13):264–283.

Chen N, Liu Y (2014) American option sensitivities estimation via a generalized infinitesimal perturbation analysis approach. *Oper. Res.* 62(3):616–632.

Chen H, Schmeiser BW (2019) Mise-optimal intervals for MNO–PQRS estimators of Poisson rate functions. *2019 Winter Simulation Conf. (WSC)* (IEEE, Piscataway, NJ), 368–379.

Chen S, Xie W (2022) On cluster-aware supervised learning: Frameworks, convergent algorithms, and applications. *INFORMS J. Comput.* 34(1):481–502.

Chen N, Lee DK, Negahban SN (2019) Super-resolution estimation of cyclic arrival rates. *Ann. Statist.* 47(3):1754–1775.

Chen N, Gürlek R, Lee DK, Shen H (2024) Can customer arrival rates be modelled by sine waves? *Service Sci.* 16(2):70–84.

Crane R, Sornette D (2008) Robust dynamic classes revealed by measuring the response function of a social system. *Proc. Natl. Acad. Sci. USA* 105(41):15649–15653.

Cui T, Ouyang Y, Shen ZJM (2010) Reliable facility location design under the risk of disruptions. *Oper. Res.* 58(4-part-1):998–1011.

Daley DJ, Vere-Jones D (2003) *An Introduction to the Theory of Point Processes: Volume I: Elementary Theory and Methods* (Springer, New York), 211–275.

Daley DJ, Vere-Jones D (2007) *An Introduction to the Theory of Point Processes: Volume II: General Theory and Structure* (Springer Science & Business Media, New York).

Daw A, Pender J (2018) Queues driven by Hawkes processes. *Stoch. Syst.* 8(3):192–229.

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the em algorithm. *J. Royal Statist. Soc. Series B (Methodological)* 39(1):1–38.

Diggle PJ (2006) Spatio-temporal point processes: Methods and applications. *Monogr. Statist. Appl. Probab.* 107:1–46.

Dong Z, Zhu S, Xie Y, Mateu J, Rodríguez-Cortés FJ (2023) Non-stationary spatio-temporal point process modeling for high-resolution COVID-19 data. *J. Royal Statist. Soc. Ser. C App. Statist.* 72(2):368–386.

Du N, Farajtabar M, Ahmed A, Smola AJ, Song L (2015) Dirichlet-Hawkes processes with applications to clustering continuous-time document streams. *Proc. 21th ACM SIGKDD internat. Conf. knowledge Discovery Data Mining* (ACM, New York), 219–228.

Du N, Dai H, Trivedi R, Upadhyay U, Gomez-Rodriguez M, Song L (2016) Recurrent marked temporal point processes: Embedding event history to vector. *Proc. 22nd ACM SIGKDD Internat. Conf. Knowledge Discovery Data Mining* (ACM, New York), 1555–1564.

Errais E, Giesecke K, Goldberg LR (2010) Affine point processes and portfolio credit risk. *SIAM J. Financial Math.* 1(1):642–665.

Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. *Second Internat. Conf. Knowledge Discovery Data Mining* 96(34):226–231.

Fan S, Li X, Zhao JL (2017) Collaboration process pattern approach to improving teamwork performance: A data mining-based methodology. *INFORMS J. Comput.* 29(3):438–456.

Farajtabar M, Wang Y, Gomez Rodriguez M, Li S, Zha H, Song L (2017) Coevolve: A joint point process model for information diffusion and network co-evolution. Cortes C, Lawrence N, Lee D, Sugiyama M, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 28 (Curran Associates, Inc., Red Hook, NY).

Fierro R, Leiva V, Møller J (2015) The Hawkes process with different exciting functions and its asymptotic behavior. *J. Appl. Probab.* 52(1):37–54.

Filimonov V, Sornette D (2012) Quantifying reflexivity in financial markets: Toward a prediction of flash crashes. *Phys. Rev. E* 85(5):056108.

Fuentes-Santos I, Gonzaléz-Manteiga W, Zubelli J (2021) Nonparametric spatiotemporal analysis of violent crime. A case study in the Rio de Janeiro metropolitan area. *Spat. Stat.* 42:100431.

Gan G, Ma C, Wu J (2020) *Data Clustering: Theory, Algorithms, and Applications* (Society for Industrial and Applied Mathematics, Philadelphia).

Gao X, Zhu L (2018) Functional central limit theorems for stationary Hawkes processes and application to infinite-server queues. *Queueing Syst.* 90:161–206.

Gopal RD, Ramesh R (1995) The query clustering problem: A set partitioning approach. *IEEE Trans. Knowledge Data Eng.* 7(6):885–899.

Guo Z, Li J, Ramesh R (2019) Optimal management of virtual infrastructures under flexible cloud service agreements. *Inform. Systems Res.* 30(4):1424–1446.

Guo Z, Li J, Ramesh R (2020) Scalable, adaptable, and fast estimation of transient downtime in virtual infrastructures using convex decomposition and sample path randomization. *INFORMS J. Comput.* 32(2):321–345.

Hawkes AG (1971) Spectra of some self-exciting and mutually exciting point processes. *Biometrika* 58(1):83–90.

Hawkes AG, Oakes D (1974) A cluster process representation of a self-exciting process. *J. Appl. Probab.* 11(3):493–503.

Henderson SG (2003) Estimation for nonhomogeneous Poisson processes from aggregated data. *Oper. Res. Lett.* 31(5):375–382.

Hu J, Reiter JP, Wang Q (2018) Dirichlet process mixture models for modeling and generating synthetic versions of nested categorical data. *Bayesian Anal.* 13(1):183–200.

Jian N, Henderson SG (2015) An introduction to simulation optimization. 2015 *Winter Simulation Conf.* (*WSC*) (IEEE, Piscataway, NJ), 1780–1794.

Jolliffe IT, Cadima J (2016) Principal component analysis: A review and recent developments. *Philos. Trans. Roy. Soc. A* 374(2065):20150202.

Kim SH, Whitt W (2014a) Are call center and hospital arrivals well modeled by nonhomogeneous Poisson processes? *Manufacturing Service Oper. Management* 16(3):464–480.

Kim SH, Whitt W (2014b) Choosing arrival process models for service systems: Tests of a nonhomogeneous Poisson process. *Naval Res. Logist.* 61(1):66–90.

Kohonen T (1990) The self-organizing map. *Proc. IEEE* 78(9):1464–1480.

Kuhl ME, Wilson JR (2001) Modeling and simulating Poisson processes having trends or nontrigonometric cyclic effects. *European J. Oper. Res.* 133(3):566–582.

Lee S, Wilson JR, Crawford MM (1991) Modeling and simulation of a nonhomogeneous Poisson process having cyclic behavior. *Comm. Statist. Simulation Comput.* 20(2–3):777–809.

Li H, Li H, Bhowmick SS (2020) Brunch: Branching structure inference of hybrid multivariate Hawkes processes with application to social media. *Adv. Knowledge Discovery Data Mining: 24th Pacific-Asia Conf., PAKDD 2020, Proc.*, vol. 24, Part I (Springer International Publishing, New York), 553–566.

Li C, Song Z, Wang X (2019) Nonparametric method for modeling clustering phenomena in emergency calls under spatial-temporal self-exciting point processes. *IEEE Access* 7:24865–24876.

Li J, Liu C, Yu JX, Chen Y, Sellis T, Culpepper JS (2016) Personalized influential topic search via social network summarization. *IEEE Trans. Knowledge Data Eng.* 28(7):1820–1834.

Li S, Xiao S, Zhu S, Du N, Xie Y, Song L (2018) Learning temporal point processes via reinforcement learning. *Proc. 32nd Internat. Conf. Neural Inform. Processing Systems* (Curran Associates Inc., Red Hook, NY), 10804–10814.

Lima R (2023) Hawkes processes modeling, inference, and control: An overview. *SIAM Rev.* 65(2):331–374.

Lin SB, Tang S, Wang Y, Wang D (2022) Toward efficient ensemble learning with structure constraints: Convergent algorithms and applications. *INFORMS J. Comput.* 34(6):3096–3116.

Liu Y, Shi J, Chen Y (2019b) Thread structure learning on online health forums with partially labeled data. *IEEE Trans. Comput. Soc. Syst.* 6(6):1273–1282.

Liu Y, Yan T, Chen H (2018) Exploiting graph regularized multi-dimensional Hawkes processes for modeling events with spatio-temporal characteristics. *IJCAI* (AAAI Press, Palo Alto, CA), 2475–2482.

Liu R, Kuhl ME, Liu Y, Wilson JR (2019a) Modeling and simulation of nonstationary non-Poisson arrival processes. *INFORMS J. Comput.* 31(2):347–366.

Manrique-Vallier D, Hu J (2018) Bayesian non-parametric generation of fully synthetic multivariate categorical data in the presence of structural zeros. *J. Roy. Statist. Soc. Ser. A* 181(3):635–647.

Mehrdad B, Zhu L (2014) On the Hawkes process with different exciting functions. Preprint, submitted March 5, https://arxiv.org/abs/1403.0994.

Mei H, Eisner J (2017) The neural Hawkes process: A neurally self-modulating multivariate point process. *Proc. 31st Internat. Conf. Neural Inform. Processing Systems* (Curran Associates Inc., Red Hook, NY), 6757–6767.

Meng Q, Xiao K, Shen D, Zhu H, Xiong H (2022) Fine-grained job salary benchmarking with a nonparametric Dirichlet process–based latent factor model. *INFORMS J. Comput.* 34(5):2443–2463.

Mohler G (2014) Marked point process hotspot maps for homicide and gun crime prediction in Chicago. *Internat. J. Forecast.* 30(3):491–497.

Mohler GO, Short MB, Brantingham PJ, Schoenberg FP, Tita GE (2011) Self-exciting point process modeling of crime. *J. Amer. Statist. Assoc.* 106(493):100–108.

Morgan LE, Nelson BL, Titman AC, Worthington DJ (2019) A spline-based method for modelling and generating a nonhomogeneous Poisson process. *2019 Winter Simulation Conf. (WSC)* (IEEE, Piscataway, NJ), 356–367.

Nelson BL, Leemis LM (2020) The ease of fitting but futility of testing a nonstationary Poisson processes from one sample path. *2020 Winter Simulation Conf. (WSC)* (IEEE, Piscataway, NJ), 266–276.

Nielsen F (2016) *Hierarchical Clustering. Introduction to HPC with MPI for Data Science* (Springer, New York), 195–211.

Ogata Y (1998) Space-time point-process models for earthquake occurrences. *Ann. Inst. Statist. Math.* 50(2):379–402.

Ozaki T (1979) Maximum likelihood estimation of Hawkes' self-exciting point processes. *Ann. Inst. Statist. Math.* 31(1):145–155.

Pasupathy R (2010) *Generating Homogeneous Poisson Processes*, Wiley Encyclopedia of Operations Research and Management Science (John Wiley, Hoboken, NJ).

Rathbun SL (1996) Asymptotic properties of the maximum likelihood estimator for spatio-temporal point processes. *J. Statist. Plann. Inference* 51(1):55–74.

Reinhart A (2018) A review of self-exciting spatio-temporal point processes and their applications. *Statist. Sci.* 33(3):299–318.

Resnick SI (1992) *Adventures in Stochastic Processes* (Birkhäuser, Boston).

Rizoiu MA, Lee Y, Mishra S, Xie L (2017) Hawkes processes for events in social media. *Frontiers of Multimedia Research* (Association for Computing Machinery and Morgan & Claypool, New York), 191–218.

Saltzman EA, Drew JH, Leemis LM, Henderson SG (2012) Simulating multivariate nonhomogeneous Poisson processes using projections. *ACM Trans. Model. Comput. Simul.* 22(3):1–13.

Schoenberg FP (2005) Consistent parametric estimation of the intensity of a spatial-temporal point process. *J. Statist. Plann. Inference* 128(1):79–93.

Schubert E, Sander J, Ester M, Kriegel HP, Xu X (2017) Dbscan revisited, revisited: Why and how you should (still) use dbscan. *ACM Trans. Database Syst.* 42(3):1–21.

Seref O, Fan YJ, Chaovalitwongse WA (2014) Mathematical programming formulations and algorithms for discrete k-median clustering of time-series data. *INFORMS J. Comput.* 26(1):160–172.

Tari L, Tu PH, Hakenberg J, Chen Y, Son TC, Gonzalez G, Baral C (2010) Incremental information extraction using relational databases. *IEEE Trans. Knowledge Data Eng.* 24(1):86–99.

Ungun B, Xing L, Boyd S (2019) Real-time radiation treatment planning with optimality guarantees via cluster and bound methods. *INFORMS J. Comput.* 31(3):544–558.

Veen A, Schoenberg FP (2008) Estimation of space–time branching process models in seismology using an em–type algorithm. *J. Amer. Statist. Assoc.* 103(482):614–624.

Wang S, Ng SH, Haskell WB (2022) A multilevel simulation optimization approach for quantile functions. *INFORMS J. Comput.* 34(1):569–585.

Xiao S, Yan J, Yang X, Zha H, Chu S (2017b) Modeling the intensity function of point process via recurrent neural networks. *Proc. AAAI Conf. Artificial Intelligence* 31(1).

Xiao S, Farajtabar M, Ye X, Yan J, Song L, Zha H (2017a) Wasserstein learning of deep generative point process models. *Proc. 31st Internat. Conf. Neural Inform. Processing Systems* (Curran Associates Inc., Red Hook, NY), 3250–3259.

Xiao S, Xu H, Yan J, Farajtabar M, Yang X, Song L, Zha H (2018) Learning conditional generative models for temporal point processes. *Proc. AAAI Conf. Artificial Intelligence* (AAAI Press, Palo Alto, CA), vol. 32.

Xu H, Zha H (2017) A Dirichlet mixture model of Hawkes processes for event sequence clustering. Guyon I, Von Luxburg U, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 30 (Curran Associates Inc., Red Hook, NY).

Yamin K, Wang H, Montreuil B, Xie Y (2022) Online detection of supply chain network disruptions using sequential change-point detection for Hawkes processes. Preprint, submitted November 22, https://arxiv.org/abs/2211.12091.

Yang SH, Zha H (2013) Mixture of mutually exciting processes for viral diffusion. *Internat. Conf. Machine Learn.* (PMLR, New York), 1–9.

Yuan B, Li H, Bertozzi AL, Brantingham PJ, Porter MA (2019) Multivariate spatiotemporal Hawkes processes and network reconstruction. *SIAM J. Math. Data Sci.* 1(2):356–382.

Zhang K, Shasha D (1989) Simple fast algorithms for the editing distance between trees and related problems. *SIAM J. Comput.* 18(6):1245–1262.

Zhang H, Zheng Z (2020) Simulating nonstationary spatio-temporal Poisson processes using the inversion method. *2020 Winter Simulation Conf. (WSC)* (IEEE, Piscataway, NJ), 492–503.

Zhang X, Hong LJ, Zhang J (2014) Scaling and modeling of call center arrivals. *Proc. Winter Simulation Conf. 2014* (IEEE, Piscataway, NJ), 476–485.

Zhang H, Zhan D, Anderson J, Righter R, Zheng Z (2024) Clustering then estimation of spatio-temporal self-exciting processes. Accessed June 29, 2024, https://github.com/INFORMSJoC/2022.0351.

Zheng Z, Glynn PW (2017) Fitting continuous piecewise linear Poisson intensities via maximum likelihood and least squares. *2017 Winter Simulation Conference (WSC)* (IEEE, Piscataway, NJ), 1740–1749.

Zhou K, Zha H, Song L (2013a) *Learning Social Infectivity in Sparse Low-Rank Networks Using Multi-Dimensional Hawkes Processes. Artificial Intelligence and Statistics* (PMLR, New York), 641–649.

Zhou K, Zha H, Song L (2013b) Learning triggering kernels for multi-dimensional Hawkes processes. *Internat. Conf. Machine Learn.* (PMLR, New York), 1301–1309.

Zhou Z, Matteson DS, Woodard DB, Henderson SG, Micheas AC (2015) A spatio-temporal point process model for ambulance demand. *J. Amer. Statist. Assoc.* 110(509):6–15.

Zhu S, Xie Y (2022) Spatiotemporal-textual point processes for crime linkage detection. *Ann. Appl. Stat.* 16(2):1151–1170.

Zhu S, Li S, Peng Z, Xie Y (2021a) Imitation learning of neural spatio-temporal point processes. *IEEE Trans. Knowledge Data Eng.* 34(11):5391–5402.

Zhu S, Ding R, Zhang M, Van Hentenryck P, Xie Y (2020) Spatio-temporal point processes with attention for traffic congestion event modeling. Preprint, submitted May 15, https://arxiv.org/abs/2005.08665.

Zhu S, Yao R, Xie Y, Qiu F, Wu X (2021b) Quantifying grid resilience against extreme weather using large-scale customer power outage data. Preprint, submitted September 20, https://arxiv.org/abs/2109.09711.

Zhuang J, Ogata Y, Vere-Jones D (2002) Stochastic declustering of space-time earthquake occurrences. *J. Amer. Statist. Assoc.* 97(458):369–380.

Zhuang J, Ogata Y, Vere-Jones D (2004) Analyzing earthquake clustering features by using stochastic reconstruction. *J. Geophys. Res. Solid Earth* 109(B5):1–17.

Zipkin JR, Schoenberg FP, Coronges K, Bertozzi AL (2016) Point-process models of social network interactions: Parameter estimation and missing data recovery. *Eur. J. Appl. Math.* 27(3):502–529.