UNIVERSITY OF CALIFORNIA

Los Angeles

Large-scale and Deep Spatiotemporal Point-Process Models

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Mathematics

by

Baichuan Yuan

2020

ABSTRACT OF THE DISSERTATION

Large-scale and Deep Spatiotemporal Point-Process Models

by

Baichuan Yuan
Doctor of Philosophy in Mathematics
University of California, Los Angeles, 2020
Professor Andrea Bertozzi, Chair

Many accurate spatiotemporal data sets have recently become available for research. Real-world applications create strong demands for a better multivariate point-process modeling. In this thesis, we develop new multivariate models with generalization ability and scalability.

The first two chapters provide a research background, real-world problems and a mathematical introduction to point-process models.

In chapter 3, we develop a nonparametric method for multivariate spatiotemporal Hawkes processes with applications on network reconstruction. In contrast to prior work, which has often focused on exclusively temporal information, our approach uses spatiotemporal information and does not assume a specific parametric form. Our results demonstrate that, in comparison to using only temporal data, our approach yields improved network reconstruction, providing a basis for meaningful subsequent analysis—such as examinations of community structure and motifs—of the reconstructed networks.

In chapter 4, we present a fast and accurate estimation method for multivariate Hawkes processes. Our method, with guaranteed consistency, combines two estimation approaches. Extensive numerical experiments, with synthetic data and real-world social network data, show that our method improves the accuracy, scalability and computational efficiency of prevailing estimation approaches. Moreover, it greatly boosts the performance of Hawkes process-based models on social network reconstruction and helps to understand the spatiotemporal triggering dynamics over social media.

In chapter 5, we focus on multivariate spatial point processes, which can describe heterotopic data over space. However, highly multivariate intensities are computationally challenging due to the curse of dimensionality. To bridge this gap, we introduce a declustering-based hidden-variable model that leads to an efficient inference via a variational autoencoder (VAE). We also prove that this model is a generalization of the VAE-based model for collaborative filtering. This leads to an interesting application of spatial point-process models to recommender systems. Experimental results show the method's utility on both synthetic data and real-world data.

Finally, in chapter 6, we show how multivariate point processes can be applied to opioid overdose events and real-time prediction of the hourly crime rate. In chapter 7, we discuss future directions and conclude the thesis.

The dissertation of Baichuan Yuan is approved.

Paul Jeffrey Brantingham

Mason Alexander Porter

Wotao Yin

Andrea Bertozzi, Committee Chair

University of California, Los Angeles

2020

TABLE OF CONTENTS

1 Introduction **1**

1.1 Motivation 3

1.1.1 Social Networks 3

1.1.2 Crime Forecasting 5

1.2 Directions for Improvement 10

2 Background **12**

2.1 A Brief Review of Point Processes 12

2.1.1 Multivariate Temporal Models 14

2.1.2 Spatial Point Processes 15

2.1.3 Spatiotemporal Point Processes 17

2.1.4 Model Inference 20

2.2 Point Processes on Real-world Problems 21

3 Multivariate Hawkes Processes and Network Reconstruction **24**

3.1 Spatiotemporal Models for Network Reconstruction 25

3.1.1 A Parametric Model 27

3.1.2 A Nonparametric Model 28

3.2 Model Estimation 28

3.2.1 Parametric Model 29

3.2.2 Nonparametric Model 30

3.2.3 Simulations 33

3.3 Numerical Experiments and Results 33

3.3.1 Synthetic Data 35

3.3.2 Gowalla Friendship Network 42
3.3.3 Crime-Topic Network 45
3.3.4 Network of Crime Events 48
3.4 Conclusions and Discussion 52

4 Fast Estimation of Multivariate Hawkes Processes 53

4.1 Multivariate Hawkes Processes and Nonparametric Estimations 55
4.2 Proposed Methods for Multivariate ST-Hawkes 57
4.2.1 ST Triggering Density Estimation 58
4.2.2 Triggering Matrix Estimation 59
4.2.3 Consistency Guarantee 61
4.2.4 Computational Complexity 63
4.3 Regularization for Linear System 64
4.4 Numerical Examples 65
4.4.1 Synthetic Data 66
4.4.2 Location-based Social-Network Reconstruction 74
4.5 Conclusion 81

5 Variational Autoencoders for Highly Multivariate Point Processes . . . 82

5.1 Variational Autoencoders 84
5.2 Multivariate Spatial Point Processes 84
5.2.1 A Nonparametric Model 85
5.2.2 Variational Inference 85
5.2.3 Alternative Model 86
5.3 Experiments 91
5.3.1 Multivariate SPP on Spatial Data 92

5.3.2 Multivariate SPP with a Latent Space 95

5.3.3 Hyperparameters 96

5.3.4 Additional Experiments 97

5.4 Conclusions 98

6 Novel Applications of Our Methods 99

6.1 Drug Movers' Distance-based Hawkes Processes for Overdose Spike Early Warning 99

6.2 Graph-based Deep Modeling and Real-time Forecasting of Sparse Spatiotemporal Data 100

7 Conclusions 103

References 106

LIST OF FIGURES

1.1 Spatiotemporal Twitter events in Los Angeles (LA) about the topic "outlets" and topic "Christmas". The spatial and temporal clustering effects are clearly presented: events about "outlets" are spatially clustered around the location of the outlets and events about "Christmas" are temporally clustering around the Christmas day. 2

1.2 Top: a social network represents the influence between users in a fictional social media. Note that user E is not affected by any user. Bottom: Post-sharing timeline for each user. Here the line edge represents the triggering effect of one user's sharing on other users and the curved edge represents the self-excitation within each user. User E, not influenced by other users, displays an almost uniform distribution of sharing events. 4

1.3 Recent burglary crimes from LAPD's Crime Mapping in part of Santa Monica from 08/14/2019 to 02/09/2020. Here the red number represents the count of repetitive crimes at the same location. The underlying map is from OpenStreetMap. 7

1.4 Temporal (top) and spatial (bottom) triggering effects' decay functions for all crimes in Los Angeles from 2009 to 2014. Here r is distance (in degrees) and t is time (in days). The decay functions are estimated using the multivariate spatiotemporal Hawkes process in [YSB19] with nonparametric kernels. Note that the drop in $f(r)$ when r is between 0.001 and 0.01 might be due to the artifact of our model such as the choice of the support. 8

1.5 Mutual-triggering effects between different kinds of crimes estimated via the model in [YSB19]. Lighter color represents a stronger trigger effect. The crime types are (from 0 to 11) other, theft, grand theft auto, vandalism, burglary/theft from motor vehicle, robbery, burglary, aggravation, homicide, grand theft person, arson, and kidnap. 9

2.1 Self-exciting point-process models capture the dynamics of gang violent crime events. A temporal self-exciting point-process model $\lambda(t) = \mu + \sum_{t_i<t} Kg(t - t_i)$ with exponential kernel $g(t) = \omega e^{-\omega t}$ (Top). Non-retaliatory gang crimes assigned to each condition arise spontaneously at rate μ_j. Retaliations assigned to each condition may be triggered through separate pathways (Bottom). The parameter k_{ij} is an estimate of the average number of retaliations of type j triggered by a single crime of type i. The parameters k_{11} and k_{01} link previous crimes assigned to the treatment and baseline interventions, respectively, to retaliations subsequently assigned to the treatment intervention. The parameters k_{00} and k_{10} link previous crimes assigned to the baseline and treatment interventions, respectively, to retaliations subsequently assigned to the baseline intervention. If treatment interventions (red events) reduce the risk of gang retaliation, then we expect $k_{11} < k_{01}$ and $k_{10} < k_{00}$. 16

2.2 Spatiotemporal point process intensities of gang crimes in South Los Angeles. (A). The log of background intensity function μ for gang violent crimes mapped over space. (B) Contour plot of the density of background gang aggravated assaults and homicides determined by declustering. (C) Point locations of background gang aggravated assaults and homicides determined by declustering. (D) The log of spatiotemporal self-excitation of retaliation $\lambda - \mu$ mapped over space. (E) Contour plot of the density of retaliatory gang aggravated assaults and homicides determined by declustering. (F) Point locations of retaliatory gang aggravated assaults and homicides determined by declustering. Boundaries for ten GRYD IR Zones [BSY17] in South Los Angeles are outlined in black. 19

3.1 Model comparison using synthetic networks. We show the mean ROC curves with error bars (averaged over ten simulations, each with a different triggering matrix) on edge reconstruction. The ROC curve of a better reconstruction should be closer to 1 for a larger range of horizontal-axis values, such that it has a larger area under the curve (AUC), which is equal to the probability that a uniformly-randomly chosen existing edge in a ground-truth network has a larger weight than a uniformly-randomly chosen missing edge in the inferred network. 39

3.2 Model comparison using synthetic networks: Inferred (left) temporal and (right) spatial kernels using three different methods: Temporal Hawkes, Parametric Hawkes, and Nonparametric Hawkes. The dashed curves are (ground-truth) kernels that we used to generate the synthetic data. 41

3.3 ROC curves of four different Hawkes models for reconstructing three Gowalla friendship networks. We show results for Nonparametric Hawkes (blue dashed curves), Temporal Hawkes (yellow dash-dotted curves), Bayesian Hawkes (red solid curves), and Parametric Hawkes (dotted purple curves). 43

3.4 Three different friendships networks in the Gowalla data set. We compare different network-reconstruction methods for these networks. 46

3.5 Crime-topic networks generated by the Nonparametric Hawkes and Temporal Hawkes methods colored by community assignments from modularity maximization (from left to right): Nonparametric Hawkes in Westwood, Temporal Hawkes in Westwood, Nonparametric Hawkes in Wingfoot, and Temporal Hawkes in Wingfoot. 47

3.6 All possible three-node motifs for a network of events in the form of a directed acyclic graph (DAG). The DAG structure arises from the temporal information in the events. We highlight the nodes in the feedforward-loop motif (D) in red. 51

4.1 The estimation results of STHC on $U = 1$ data. Ground-truth spatial triggering density $f(r)$ as red triangles and estimated triggering density as blue circles (left). Temporal triggering density $h(t)$ as red triangles and estimated triggering density as blue circles (right). 67

4.2 Ground-truth $\boldsymbol{K}$ matrix, STHC (the first row, from left to right), NPHC and EM estimation results (the second row, from left to right). 68

4.3 Ground-truth $\boldsymbol{K}$ matrix, STHC, NPHC and EM estimation results (from left to right). 69

4.4 The estimation results of STHC on $U = 10$ data. Ground-truth spatial triggering density $f(r)$ as red triangles and estimated triggering density as blue circles (left). Temporal triggering density $h(t)$ as red triangles and estimated triggering density as blue circles (right). 70

4.5 The estimation results of STHC on $U = 10$ data with a Pareto triggering density in time, a uniform triggering density in time (the first row, from left to right), a power-law triggering density in space and a uniform triggering density in space (the second row, from left to right). Ground-truth spatial triggering density $f(r)$ as red triangles and estimated triggering density as blue circles. 71

4.6 The estimation results of STHC on $U = 10$ data with a Pareto triggering density in time, a uniform triggering density in time (the first row, from left to right), a power-law triggering density in space and a uniform triggering density in space (the second row, from left to right). Ground-truth temporal triggering density $h(t)$ as red triangles and estimated triggering density as blue circles. 72

4.7 ROC curves of different methods (STHC, NPHC and EM) on subnetworks in Gowalla and Brightkite data sets. The dashed line (red) is from random guess. . 76

4.8 Friendship network reconstruction using different methods on Brightkite-SD. Here we zoom in to show a subgraph within the Brightkite-SD network. 77

4.9 Estimated spatial triggering densities for Brightkite-SD, Gowalla-CHI (the first row, from left to right), Brightkite-LA, and Gowalla-SF (the second row, from left to right). The plot is in the log-log scale and we normalize the triggering density for easy comparison. Note that the drop in $f(r)$ when r is between 0.0001 and 0.001 might be due to the artifact of our model such as the choice of the support for triggering densities. 78

4.10 Estimated temporal triggering densities for Brightkite-SD, Gowalla-CHI (the first row, from left to right), Brightkite-LA, and Gowalla-SF (the second row, from left to right). The plot is in the log-log scale and we normalize the triggering density for easy comparison. 79

5.1 Visual illustration of our spatial point-process model via VAE. The spatial events or embeddings X are fed into a neural network with parameter ϕ to get the mean and variance for the approximate posterior $q_\phi(z|X)$, which is then used to calculate the KL divergence term. We can generate samples from the posterior using the reparameterization trick with ϵ from a standard normal distribution. Then the sample z is fed into a neural network with parameter θ to obtain $f(z)$, which combining with the events X and kernel K_σ yields the intensity function λ. Finally, we get the loss function from the KL term and the likelihood function from λ. 87

5.2 Estimated density functions for a Gowalla user in NYC (log scale). The first row from left to right: observed check-in locations (in red), held-out check-in locations (in blue, as missing data) and the estimated intensity from VAE-SPP. The second row from left to right: the estimated intensity (or density) from VAE-CF, KDE and TGCP. 95

6.1 Overview of the system for early warning of opioid spikes. The initial overdose toxicology report shows fentanyl, benzodiazepine, and heroin present. Each drug is vectorized using SMILES and the event belongs to an overdose category using spectral clustering based on earth mover's distance of the drug vectors ("drug movers' distance"). The increase in the intensity of the Hawkes process is determined by the category and it allows for the prediction of an opioid overdose spike, triggered in the branching process by the initial overdose. 100

6.2 An illustration for an initial event and its triggered events in one of the categories (i.e. one of the Hawkes processes). The initial overdose event marked in triangle symbol consists of four drug substances and it triggered four neighboring events consisting of different drug substances. 101

6.3 Flow chart of the algorithm. The spatiotemporal events are fed into a multivariate Hawkes process to obtain the graph of zip codes. The graph is then used in GSRNN with two LSTM networks and a fully connected layer. 102

6.4 Inferred versus exact hourly crime rates for Chicago (left) and Los Angeles (right) data. The unit of time is in hours. 102

LIST OF TABLES

2.1 Notation. 22

3.1 Reciprocity of the triggering matrices that we infer using different methods: a nonparametric spatiotemporal Hawkes model, a temporal Hawkes model, a parametric spatiotemporal Hawkes model, and a fully Bayesian Hawkes model. We report the mean and standard deviation (in parentheses) over ten simulations that use the same (ground-truth) triggering matrix. 37

3.2 Reciprocity of the triggering matrices that we infer using different methods: a nonparametric spatiotemporal Hawkes model, a temporal Hawkes model, a parametric spatiotemporal Hawkes model, and a fully Bayesian Hawkes model. We report the mean and standard deviation (in parentheses) over ten simulations, each with a different (ground-truth) triggering matrix. 37

3.3 The L_1 errors of the inferred spatial and temporal kernels. We simulate ten point processes with the same triggering matrix and triggering kernel. We report the mean and standard deviation (in parentheses) of the L_1 errors averaged over ten simulations that use the same triggering kernel and matrix. Note that the exclusively temporal model does not estimate a spatial kernel. 40

3.4 Normalized mutual information (NMI) between the outputs of different community-detection methods applied to the inferred networks (from four different types of Hawkes models) and the ground-truth community structure (averaged over ten simulations, each with a different triggering matrix). 42

3.5 Mean NMI (with one standard deviation reported in parentheses) between community assignments from several community-detection methods and the classifications from [KBB17] in the 100 neighborhoods in Los Angeles with the most recorded crime events between 1 January 2009 and 19 July 2014. 48

3.6 Comparison of our stochastic declustering results for the Nonparametric Hawkes, Parametric Hawkes, and Temporal Hawkes methods using synthetic point-process data with networks from a WSBM (see section 3.3.1) and background labels from the simulation from algorithm 3. (We do not include results for Bayesian Hawkes, because it does not provide **P** directly.) We report the mean and the standard deviation (in parentheses) of the branching-ratio error, precision, and recall over ten simulations (which we do for ten point processes with the same triggering kernels and matrix). For each simulation, each calculation is the mean over 20 runs of stochastic declustering. 50

4.1 The computation time for different methods on synthetic data sets. Here the time is in seconds. 69

4.2 Error measures for STHC on $U = 10$ data sets with different triggering densities. 70

4.3 The computation time for different methods on Gowalla and Brightkite data sets. Here the time is in seconds. 79

5.1 Testing results on the simulation data sets. Both the mean and standard derivation (in parentheses) are percentages. 93

5.2 Testing results on the Gowalla data sets with uniform grids. Both the mean and standard derivation (in parentheses) are percentages. 94

5.3 Testing results on the Gowalla data sets without discretization. Both the mean and standard derivation (in parentheses) are percentages. 94

5.4 Testing results on the MovieLens data sets. 96

5.5 Testing results on MovieLens-100K. These methods share the same network and are trained with 100 epochs. The test data are used to evaluate the model with the best performance during the validation. VAE-SPP-Separate means that GNN is trained separately with VAE-SPP. 98

ACKNOWLEDGMENTS

I would like to pay special regards to my advisor, Andrea Bertozzi, for her guidance through each stage of the process. She has always been helpful from my undergraduate summer research project to the writing of this thesis. Without her kindness and invaluable assistance, it would have been impossible for me to finish this rewarding Ph.D. journey. I would like to show my gratitude to the other committee members for all of their persistent help during many of my projects: P. Jeffrey Brantingham for sharing data and contributing ideas about crime as a social scientist, Mason A. Porter for introducing ideas in network science and for giving writing suggestions, and Wotao Yin for helping me with the model optimization. I wish to thank all the collaborators whose assistance was a milestone in the completion of each of these projects. I would like to pay special regards to Frederic (Rick) Paik Schoenberg for his great advice throughout my study. I would like to also acknowledge the support and great love from my family and friends, my girlfriend Candice, my mother Haifeng, my father Jianchun, and the many Ph.D. students in our program. They kept me deeply motivated and this thesis would not have been achievable without their help.

Chapter 3 includes *Multivariate Spatiotemporal Hawkes Processes and Network Reconstruction* [YLB19] by Baichuan Yuan, Hao Li, Andrea L. Bertozzi, P. Jeffrey Brantingham, and Mason A. Porter, SIAM J. Math. Data Sci 1.2 (2019): 356–382, DOI:10.1137/18M1226993. BY contributed to the derivation of algorithms, simulation methods, and numerical experiments. HL helped to evaluate simulation data and real-world applications. ALB helped with examples and many suggestions. PJB helped to interpret results related to crime. MAP helped with ideas from network science and writing suggestions. All authors assisted with manuscript preparation.

Chapter 4 contains *Fast Estimation of Multivariate Spatiotemporal Hawkes Processes and Network Reconstruction* [YSB19] by Baichuan Yuan, Frederic (Rick) Paik Schoenberg and Andrea L. Bertozzi, submitted. BY developed the inference approach and run numerical experiments. FPS helped with the consistency proof and the design of experiments. ALB helped with ideas in L_2 regularization and many suggestions. All assisted in manuscript

preparation.

Chapter 5 contains *Variational Autoencoders for Highly Multivariate Spatial Point Processes Intensities* [YWM20] by Baichuan Yuan, Xiaowei Wang, Jianxin Ma, Chang Zhou, Andrea L. Bertozzi and Hongxia Yang, accepted by the International Conference on Learning Representations, 2020. BY contributed to the algorithm formulation, proofs, numerical experiments, and writing. XW helped with figures, the notation table, and writing. JM and CZ provided expertise on variational inference and recommender systems. ALB helped with numerous suggestions on research and writing. HY helped with the problem formulation and writing.

Chapter 6 contains two application papers: *Graph-based deep modeling and real-time forecasting of sparse spatiotemporal data* [WLZ18] by Bao Wang, Xiyang Luo, Fangbo Zhang, Baichuan Yuan, Andrea L. Bertozzi, and P. Jeffery Brantingham, KDD MiLeTS workshop, 2018. BW, XL, and FZ contributed equally to the algorithm and numerical experiments. BY contributed the point process part. PJB helped with crime data and insights about the results. ALB helped with many suggestions on the problem formulation and paper structure. All authors assisted with manuscript preparation; *SOS-EW: System for Overdose Spike Early Warning using Drug Mover's Distance-based Hawkes Processes* [CYL19] by Wen-Hao Chiang, Baichuan Yuan, Hao Li, Bao Wang, Andrea L. Bertozzi, Jeremy Carter, Brad Ray, and George Mohler, ECML-PKDD SoGood Workshop, 2019. WC and BY contributed equally to the model design and experiments. HL and BW helped with data preprocessing. ALB helped with many suggestions on the project and writing. JC and BR provided data and helped with writing. GM helped to guide the research project and run numerical experiments.

I truly appreciate the fellowship from the National Institute of Justice. NIJ Fellowship 2018-R2-CX-0013 provided essential support for my last two years in school in order to finish this thesis.

VITA

2015	B.S. (Mathematics and Applied Mathematics), Zhejiang University, China
2015–2018	Graduate Research Assistant, Department of Mathematics, UCLA.
2018–2020	National Institute of Justice Graduate Research Fellow, Department of Mathematics, UCLA.

PUBLICATIONS

Baichuan Yuan, Frederic P. Schoenberg, Andrea L. Bertozzi. Fast Estimation of Multivariate Spatiotemporal Hawkes Processes and Network Reconstruction. Submitted.

Baichuan Yuan, Xiaowei Wang, Jianxi Ma, Chang Zhou, Andrea L. Bertozzi, Hongxia Yang. Variational Autoencoders for Highly Multivariate Spatial Point Processes Intensities, accepted to International Conference on Learning Representations (ICLR), 2020.

Baichuan Yuan, Hao Li, Andrea L. Bertozzi, P. Jeffrey Brantingham, and Mason Porter, Multivariate Spatiotemporal Hawkes Processes and Network Reconstruction, SIAM J. Mathematics of Data Science, 1(2), pp. 356–382, 2019.

Baichuan Yuan, Yen Joe Tan, Maruti K. Mudunuru, Omar E. Marcillo, Andrew A. Delorey, Peter M. Roberts, Jeremy D. Webster, Christine N. L. Gammans, Satish Karra, George D. Guthrie, Paul A. Johnson; Using Machine Learning to Discern Eruption in Noisy Environments: A Case Study Using CO2-Driven Cold-Water Geyser in Chimayó, New Mexico. Seismological Research Letters, 90 (2A): 591–603, 2019.

Wen-Hao Chiang*, Baichuan Yuan*, Hao Li, Bao Wang, Andrea Bertozzi, Jeremy Carter, Brad Ray, George Mohler; SOS-EW: System for Overdose Spike Early Warning using Drug Mover's Distance-based Hawkes Processes. ECML-PKDD Workshop on Data Science for Social Good, 2019. (* equal contribution)

Bao Wang, Xiyang Luo, Fangbo Zhang, Baichuan Yuan, Andrea L Bertozzi, P Jeffrey Brantingham; Graph-Based Deep Modeling and Real Time Forecasting of Sparse Spatio-Temporal Data. KDD Workshop on Mining and Learning from Time Series, 2018.

Yonatan Dukler, Yurun Ge, Yizhou Qian, Shintaro Yamamoto, Baichuan Yuan*, Long Zhao, Andrea L. Bertozzi, Blake Hunter, Rafael Llerena, and Jesse T. Yen, Automatic decomposition and mitral valve segmentation of cardiac ultrasound time series data, Medical Imaging 2018: Image Processing. Vol. 10574. International Society for Optics and Photonics, 2018. (* corresponding author)

Baichuan Yuan, Sathya R Chitturi, Geoffrey Iyer, Nuoyu Li, Xiaochuan Xu, Ruohan Zhan, Rafael Llerena, Jesse T Yen, Andrea L Bertozzi, Machine Learning for Cardiac Ultrasound Time Series Data, Medical Imaging 2017: Biomedical Applications in Molecular, Structural, and Functional Imaging. Vol. 10137. International Society for Optics and Photonics, 2017.

Eric L. Lai*, Daniel Moyer*, Baichuan Yuan*, Eric Fox, Blake Hunter, Andrea L. Bertozzi, P Jeffrey Brantingham, Topic Time Series Analysis of Microblogs, IMA Journal of Applied Mathematics, 81(3) pp. 409-431, 2016. (* equal contribution)

CHAPTER 1

Introduction

Spatiotemporal (ST) point processes, especially ST-Hawkes processes (also known as "self-exciting point processes"[1]) have been widely used to model and forecast clustered point-process data in the study of earthquakes [Oga98], crimes [MSB11], invasive species [BSM12], terrorist attacks [PW12], infectious diseases [Sch18], and financial markets [BMM15]. These models, which are characterized by a triggering density describing how the occurrence of one event may spark future events nearby, have real-world impacts on the crime rate in Los Angeles [MSM15].

Recently, digital devices such as smartphones and tablets generate a massive amount of spatiotemporal data on human activities, providing a wonderful opportunity for researchers to gain insight into human dynamics through our "digital footprints". A wide variety of human activities are now analyzed using such data, creating new disciplines such as computational social science and digital humanities [LPA09]. Examples of such activities include online check-ins in large cities [CML11], human mobility [BCG09] and currency flow [BHG06], online communications during Occupy Wall Street[CDF13], crime reports in Los Angeles county [KBB17], and many others. Spatiotemporal point processes [SBG14], as a class of generative models, can detect and explain lots of clustering effects from structural differences in space and time. In fig. 1.1, we visualize some examples of spatiotemporal clusters from the Twitter data in [LMY16]. To further incorporate accompanying information on each event such as the type of crime or the magnitude of an earthquake, multivariate point processes have been the subject of significant research in the areas of criminology [Moh14], finance [BMM15], neuroscience [CSS17], and text analysis [DFA15]. Applications

[1] We use these terms interchangeably in this thesis.

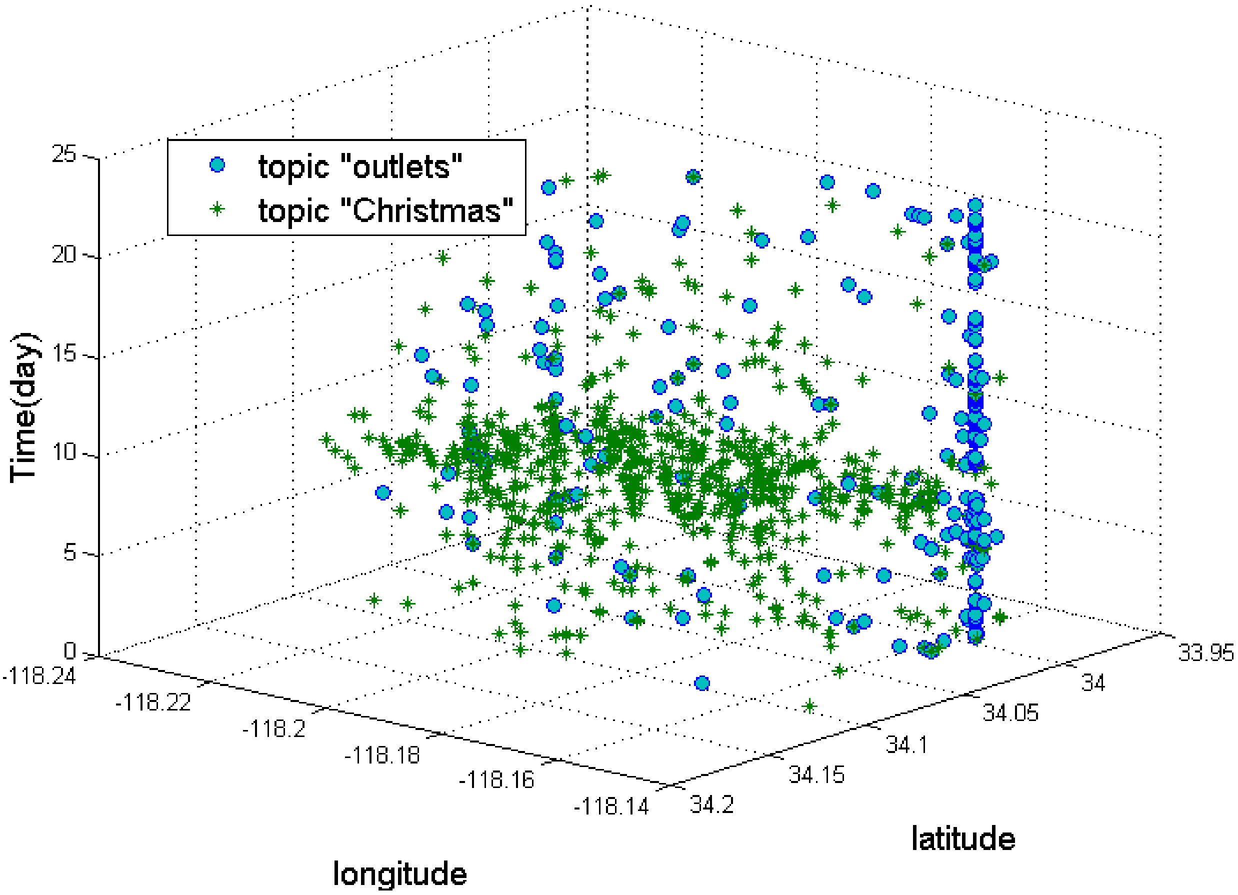


Figure 1.1: Spatiotemporal Twitter events in Los Angeles (LA) about the topic “outlets” and topic “Christmas”. The spatial and temporal clustering effects are clearly presented: events about “outlets” are spatially clustered around the location of the outlets and events about “Christmas” are temporally clustering around the Christmas day.

include network reconstruction [LA14, FSS16, HW16, YLB19, MRW18], causal inference [ABG17, EDD17, BYS18] and social-media cascade modeling [LMY16, FWR15]. The focus of this thesis is to provide a detailed review of current methods for multivariate ST point processes and their challenges when facing these new applications. Then we develop a set of approaches to bridge the gap between real-world applications and current methods.

1.1 Motivation

1.1.1 Social Networks

Network analysis is a powerful approach for representing and analyzing complex systems of interacting components [New18], and network-based methods can provide considerable insights into the structure and dynamics of complex spatiotemporal data [Bar18]. It has been valuable for studies of both digital human footprints and human mobility [BBG18]. To give one recent example, Noulas et al. [NSL12] studied geographic online social networks to illustrate similarities and heterogeneities in human mobility patterns.

Suppose that each node in a network represents an entity, and that the edges (which can be either undirected or directed, and can be either unweighted or weighted) represent spatiotemporal connections between pairs of entities. For instance, in a data set of check-ins on a social-media platform, one can model each user as a node, which has associated check-in time and locations. In this case, one can suppose that an edge exists between a pair of users if they follow each other on the platform. One can use edge weights to quantify the amount of "influence" between users, where a larger weight signifies a larger impact. In our investigation, we assume that the relationships between nodes are time-independent.[2] Here we illustrate this idea via a fictional example in fig. 1.2, where nodes are users in a social-media network and an edge from node A to node B presents the influence of user A on user B. We further visualize the impact between users when they share posts over time on social media. In some cases, the entities and relationships are both known, and one can investigate the structure and dynamics of the associated networks. However, in many situations, network data are incomplete — with potentially a large amount of missing data, in the form of missing entities, interactions, and/or metadata [SSB11] — and it may not be possible to directly observe the relationships between nodes [SNM11]. For example, social-media companies attempt to infer friendship relationships between their users to provide accurate friendship recommendations for online social networks.

[2] Depending on the relative time scales between spatiotemporal processes and network dynamics, it may be important to consider time-dependent edges [PG16, Hol15].

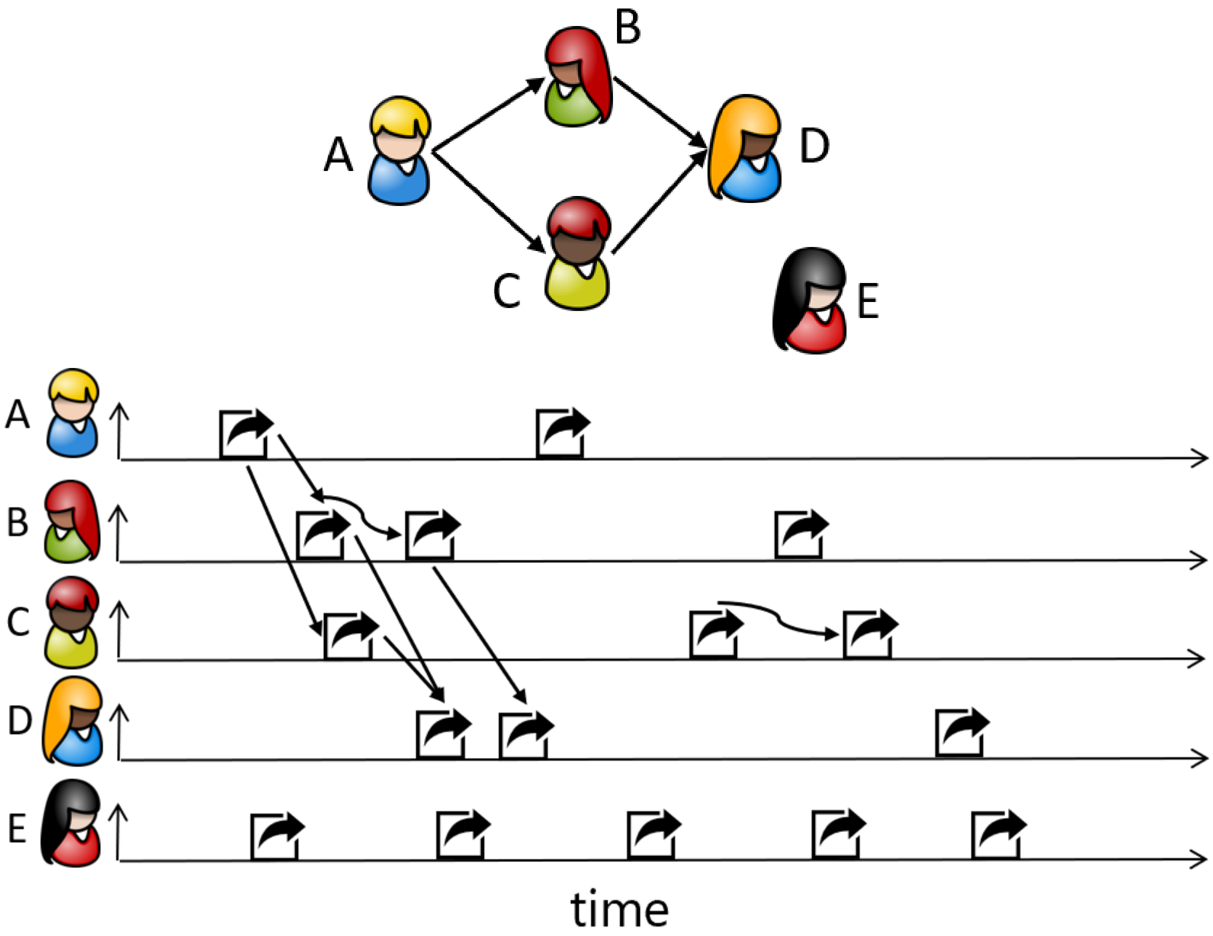


Figure 1.2: Top: a social network represents the influence between users in a fictional social media. Note that user E is not affected by any user. Bottom: Post-sharing timeline for each user. Here the line edge represents the triggering effect of one user's sharing on other users and the curved edge represents the self-excitation within each user. User E, not influenced by other users, displays an almost uniform distribution of sharing events.

In the last few years, there has been a considerable effort on inferring missing data (both the structure and weight) in networks. A basic approach for inferring relationships among entities is to calculate cross-correlations of their associated time series [Lau96]. Another approach is to use coefficients from a generalized linear model (GLM) [NW72], a generalization of linear regression that allows response variables to have a non-Gaussian error-distribution. Recently, researchers have begun to use point-process methods [SJ10] in network reconstruction. For example, Perry and Wolfe [PW13] modeled networks as multivariate point processes and then inferred covariate-based edges (both their existence and their weights). As a well-studied family of point-process models, Hawkes processes have been employed often for studying human dynamics [LA14, FSS16]. Hawkes-process models are characterized by mutual "triggering" among events [Oga88], as one event may increase the probability for subsequent events to occur. Such models can capture inhomogeneous inter-event times and causal (temporal) correlations, which are important considerations for human dynamics [KP15]. These properties illustrate the relevance of using Hawkes processes in social-network applications [KJK18]. It thus seems promising to employ such processes for network inference on dynamic human data, such as crime events and online social media. For example, Linderman and Adams [LA14] proposed a fully Bayesian Hawkes model that they reported to be more accurate at inferring missing edges for their data than GLMs, cross-correlations, and a simple self-exciting point process with an exponential kernel. However, the aforementioned temporal point-process models are not without limitations. For example, most of these models do not use spatial information, even when it plays a significant role in a system's dynamics. Furthermore, many studies assume an *a priori* model [LA14] or a specific parametrization [CGB14] for their point processes.

1.1.2 Crime Forecasting

Recent years have seen a surge of complete criminal records and related information collected by law enforcement. In Los Angeles, for example, there are nearly half-million criminal records collected by the Los Angeles Police Department (LAPD). The volume of data combined with the development of quantitative techniques has boosted crime forecasting, which helps to prevent crime and evaluate police intervention. Mohler et al. [MSM15] reported a 7.4% reduction in crime when the police department used their point-process model for daily patrol.

Current models on crime prediction usually focus on certain desired properties of predictive policing models. Crime hotspots describe crimes' spatial distribution using kernel density estimation or using spatial point processes. These models are easy to compute and can incorporate covariates such as demographics [WB12]. As a result, they are scalable to multivariate data. However, the standard model for crime hotspots is static in time and multiple timescales are usually not reflected [Moh14].

To model the dynamics of hotspots, self-exciting point-process models adapted from seismology [MSB11] assume near-repeats of crimes [HR12] and the decay of risk along time and spatial neighborhood. In fig. 1.3, we illustrate the near-repeat phenomenon via visualizing recent burglaries. The decay functions over time and space estimated via the point-process model in [YSB19] are showed in fig. 1.4. This model is extended as a marked point-process model to include covariates, such as crime type [Moh14] and spatial features [RG18]. Marked point-process models, especially multivariate models, are able to reveal the mutual-triggering effects between different crime types as showed in fig. 1.5. The inference of these models is based on maximizing the log-likelihood function using off-the-shelf optimization techniques such as BFGS, or expectation maximization (EM) algorithm [VS08]. In terms of algorithm complexity, each evaluation of the log-likelihood function is $O(N^2)$ and the overall complexity for EM is $O(N^3)$ where N is the number of crime events. This is not ideal when handling millions of crimes in the data set

Accurate spatiotemporal events forecasting is also one of the important tasks for artificial intelligence. Recent developments in the deep neural network provide multiple tools for crime forecasting. Kang et al. [KK17] utilized a convolutional neural network (CNN) to extract the features from historical crime data, and then used a support vector machine (SVM) to classify whether there will be a crime or not at the next time slot. CNN-based approaches are scalable for large data sets and have good generalization ability. The data at a certain timescale are represented by the spatial distribution histogram on the grid and a CNN is used

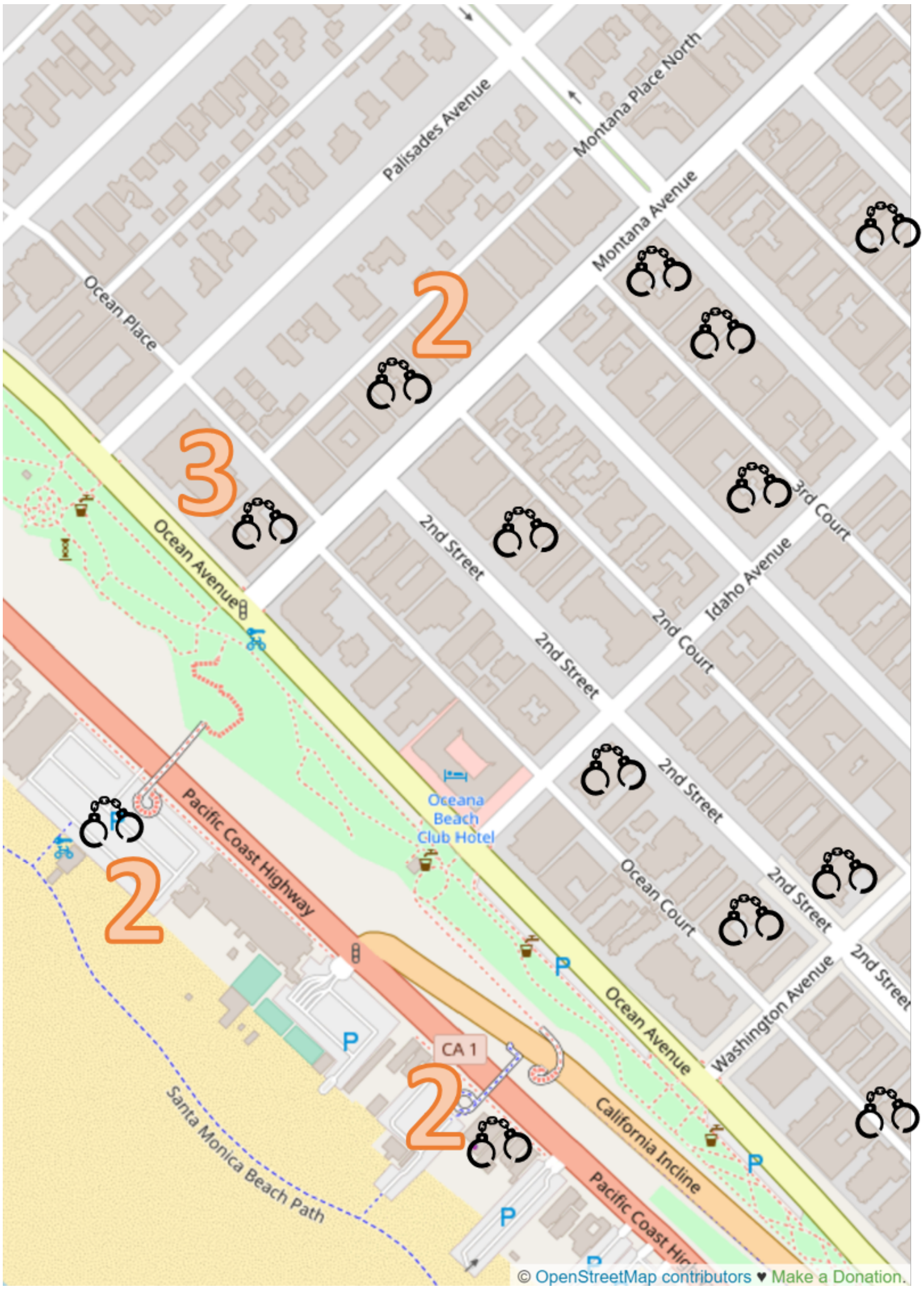


Figure 1.3: Recent burglary crimes from LAPD's Crime Mapping in part of Santa Monica from 08/14/2019 to 02/09/2020. Here the red number represents the count of repetitive crimes at the same location. The underlying map is from OpenStreetMap.

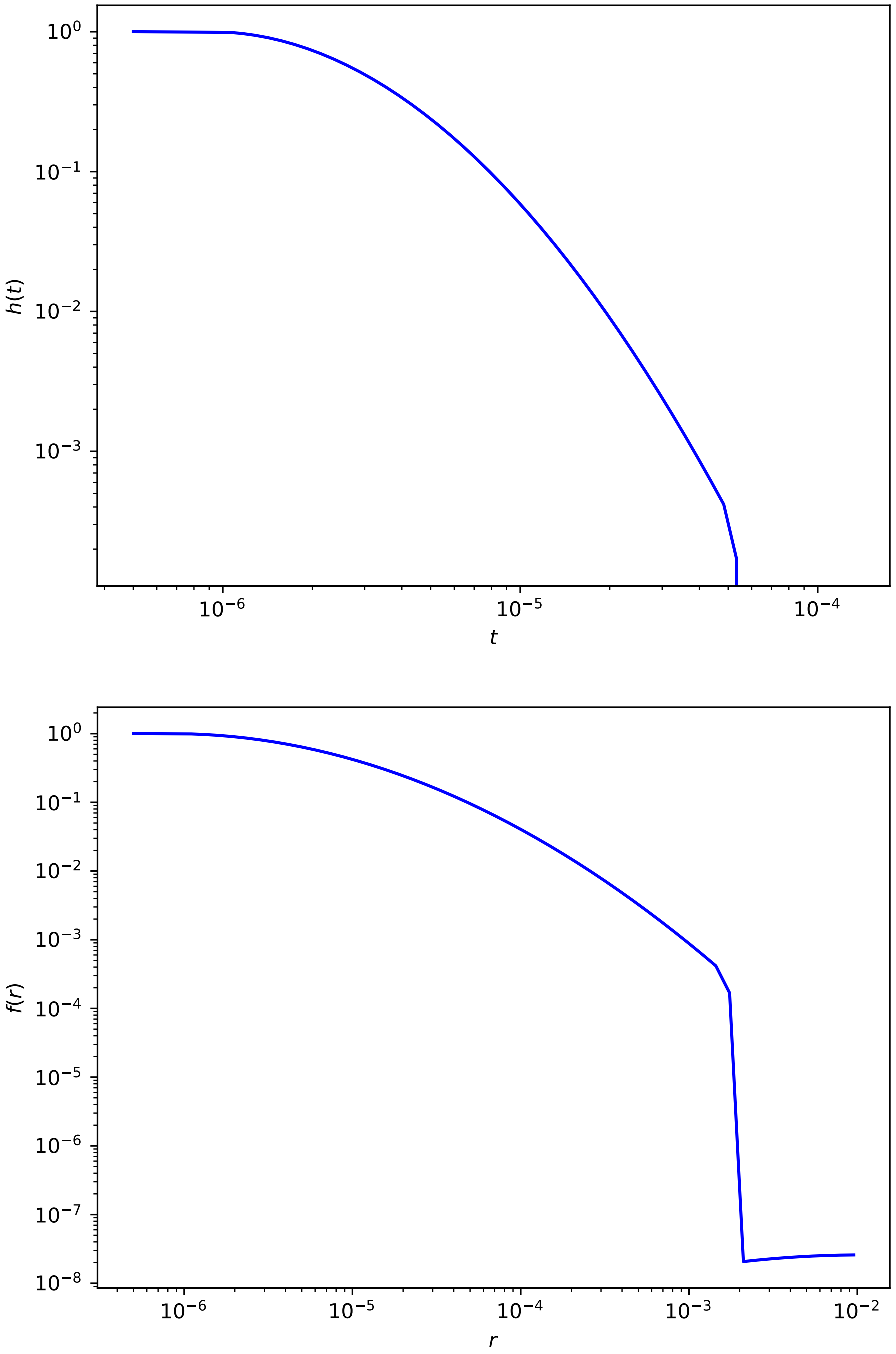


Figure 1.4: Temporal (top) and spatial (bottom) triggering effects' decay functions for all crimes in Los Angeles from 2009 to 2014. Here r is distance (in degrees) and t is time (in days). The decay functions are estimated using the multivariate spatiotemporal Hawkes process in [YSB19] with nonparametric kernels. Note that the drop in $f(r)$ when r is between 0.001 and 0.01 might be due to the artifact of our model such as the choice of the support.

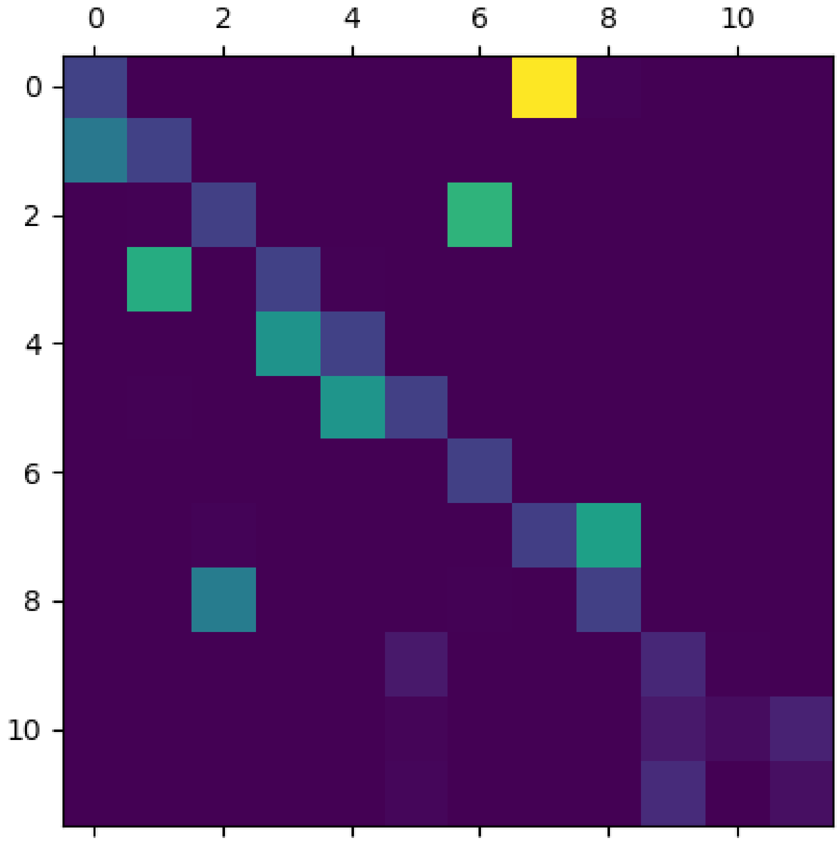


Figure 1.5: Mutual-triggering effects between different kinds of crimes estimated via the model in [YSB19]. Lighter color represents a stronger trigger effect. The crime types are (from 0 to 11) other, theft, grand theft auto, vandalism, burglary/theft from motor vehicle, robbery, burglary, aggravation, homicide, grand theft person, arson, and kidnap.

to predict the future histogram. This CNN-based approach is sub-optimal from two aspects. First, the geometry of a city is usually highly irregular, resulting in the city's configuration taking up only a small portion of its bounding box. This introduces unnecessary redundancy into the algorithm. Second, spatial sparsity can be exacerbated by the spatial grid structure. Directly applying a CNN to fit the extreme sparse data will lead to all zero weights due to the weight sharing of CNNs [WYB19]. This can be alleviated by using spatial super-resolution, with an increased computational cost. Moreover, this lattice-based data representation omits geographical information and spatial correlations within the data itself.

1.2 Directions for Improvement

To improve the effectiveness of ST point processes, a natural question to ask is what makes a good point-process model for applications above. Given the size of the data, scalable methods are essential for real-time forecasting and evaluation. Moreover, real-world data sets contain rich information aside from spatiotemporal stamps. For example, for criminal records, we want to use information such as gang involvement, a brief description of the crime, and intervention attempts. As a result, a multivariate representation of the events is useful in modeling and can utilize additional data. Finally, in forecasting problems, the ability to generalize the algorithm is ideal since we are more interested in events that we have not seen yet. In summary, a better point-process model should be able to handle large-scale multivariate data in real-time and achieve reliable results in new data. Current methods all have limits in some respects. In this thesis, we propose new approaches to achieve these three properties, including scalability, generalization ability as well as the use of the multivariate model. Multivariate models will be able to analyze millions of data in linear time $O(n)$ to achieve real-time prediction.

Our models relieve the computational burden of point-process models and make it possible to apply them on multiple large-scale problems such as network reconstruction, recommendation systems and predictive policing. For example, a real-time and accurate crime forecasting model will improve the design of police patrol. Applications of our model are not limited to these directions. Retaliation has been long featured in the discussion of gangs, rising almost to one of de facto definitive characteristics [Pap09]. During the collaboration with the City of Los Angeles Mayor's Office of Gang Reduction & Youth Development (GRYD), we discovered a promising application [BSY17] of multivariate point processes on the evaluation of GRYD's gang intervention program in terms of causal effects. In the case of missing data in crime records, our multivariate approach has been applied to crime network inference [YLB19] and gang retaliatory dynamics [BYS18].

CHAPTER 2

Background

2.1 A Brief Review of Point Processes

Given a complete, separable metric space, a *point process* S is a random measure that values in $\{0, 1, 2, \ldots\} \cup \{\infty\}$ [SBG14]. While the definitions and results below can be extended quite readily to other metric spaces, we will assume for simplicity that the metric space is a bounded interval $[0, T]$ in time or a bounded area $R \times [0, T]$ in space-time.

We first consider a *temporal point process*, which consists of a list $\{t_1, t_2, \ldots, t_N\}$ of N time points, with corresponding events $1, 2, \ldots, N$. Let $S[a, b)$ denote the number of points (i.e. events) that occur in a finite time interval $[a, b)$, with $a < b$. One typically models the behavior of a simple temporal point process (multiple events cannot occur at the same time) by specifying its conditional intensity function $\lambda(t)$, which represents the rate at which events are expected to occur around a particular time t, conditional on the prior history of the point process before time t. Specifically, when $H_t = \{t_i | t_i < t\}$ is the history of the process up to time t, one defines the *conditional intensity function*

$$\lambda(t) = \lim_{\Delta t \downarrow 0} \frac{\mathbb{E}[S[t, t + \Delta t)|H_t]}{\Delta t} \,.$$

One important point-process model is a *Poisson process*, in which the number of points in any time interval follows a Poisson distribution and numbers of points in disjoint sets are independent of each other. A Poisson process is called *homogeneous* if $\lambda(t) \equiv$ constant, and it is thus characterized by a constant rate at which events are expected to occur per unit time. It is called *inhomogeneous* if the conditional intensity function $\lambda(t)$ depends on the time t (e.g. $\lambda(t) = e^{-t}$). In both situations, numbers of points (i.e. events) in disjoint intervals are independent random variables.

We now discuss self-exciting point processes, which allow one to examine a notion of causality in point-process models. If we consider a list $\{t_1, t_2, \ldots, t_N\}$ of time stamps, we say that a point process is *self-exciting* if

$$\mathrm{Cov}\left[S(t_{k-1}, t_k), S(t_k, t_{k+1})\right] > 0\,, \quad \text{with} \quad t_{k-1} < t_k < t_{k+1}\,,$$

where k is a positive integer. In a self-exciting point process, if an event occurs, another event becomes more likely to occur locally in time.

A *univariate temporal Hawkes process*, which we express using the common cluster representation [HO74], has the following conditional intensity function:

$$\lambda(t) = \mu(t) + K \sum_{t_k < t} g(t - t_k)\,, \tag{2.1}$$

where the background rate $\mu(t) > 0$ can either be a constant or a time-dependent function that describes how the likelihood of some processes (crimes, e-mails, tweets, and so on) evolves in time. For example, violent crimes are more likely to occur at night than during the day, and business e-mails are less likely to be sent during the weekend than on a weekday. One can construe the rate $\mu(t)$ as a process that designates the likelihood of an event to occur, independent of the other events. The summation term in eq. (2.1) describes the self-excitation: past events increase the current conditional intensity. The function $g(t) \geq 0$ is called the *triggering density* or *triggering kernel* satisfying $\int_0^\infty g(v)dv = 1$, which describes the conductivity of events, and the productivity parameter K denotes the mean number of events that are triggered by an event, which is typically required to satisfy $0 \leq K < 1$ in order to ensure stationarity and subcriticality [Haw71]. One standard example is a Hawkes process with an exponential kernel $g(t) = \omega e^{-\omega t}$, where the constant decay rate ω for the triggering kernel controls how fast the rate $\lambda(t)$ returns to its baseline level $\mu(t)$ after an event occurs.

We fit the Hawkes process above to gang aggravated assaults and homicides in South Los Angeles from 2014–2015 in fig. 2.1 (A). Two cycles of gang violent crimes occur within a period of eighteen days. The conditional intensity λ reflects the instantaneous rate of gang crime. The background rate μ is the expected rate of gang crime in the absence of retaliation.

A crime causes λ to jump by an amount $K\omega$, increasing the risk of retaliation. The risk of retaliation following a single crime decays exponentially with a rate ω and a mean lifetime of $1/\omega$. We expect violence interruption deployed in the aftermath of a crime to cause the conditional intensity to fall and therefore future crimes to be less likely to occur than in the absence of intervention.

2.1.1 Multivariate Temporal Models

In a multivariate temporal point process, there are U different point processes $(S_u)_{u=1,\ldots,U}$; and the corresponding conditional intensity functions are $(\lambda_u(t))_{u=1,\ldots,U}$. We seek to infer the intensity functions from observed data $(t_j, u_j)_{j=1,\ldots,N}$ in a time window $[0, T]$, where t_j and u_j, respectively, are the time and point-process indices of event j. There are numerous applications of temporal multivariate point processes; they include financial markets [BMM15], real-time crime forecasting [WLZ18], and neuronal spike trains [BKM04].

Let's first consider two examples of multivariate processes that are not self-exciting. A trivial example of a multivariate point process is the multivariate Poisson process, in which each point process is a univariate Poisson process. Another example is the multivariate Cox process, which consists of doubly stochastic Poisson processes (so the conditional intensity itself is a stochastic process). Perry and Wolfe [PW13] used a Cox process to model e-mail interactions (edges) among a set of users (nodes).

Instead of modeling edges as Cox processes, Fox et al. [FSS16] used multivariate Hawkes processes to model people (nodes) communicating with each other via e-mail. Their conditional intensity function has an exponential kernel and a nonparametric background function $\mu_u(t) \geq 0$ for each person (process) u. It is written as

$$\lambda_u(t) = \mu_u(t) + \sum_{t_i < t} K_{u_i u} \omega e^{-\omega(t - t_i)}\,, \tag{2.2}$$

where $K_{uv} \geq 0$ is the expected number of events of person v that are triggered by one event of person u.

A (linear) multivariate temporal Hawkes process can be conveniently viewed as a sequence of temporal point processes indexed by $u = 1, ..., U$, where each subprocess N_u has conditional intensity

$$\lambda_u(t) = \mu_u + \sum_{t_k<t} K_{u_k,u}\, g_{u_k}(t - t_k)\,. \tag{2.3}$$

The idea behind this formula is that the triggering density g_{u_k} and productivity $K_{u_k,u}$ may depend on the index of the point t_k. Here μ_u is the background rate, indicating the rate at which points of mark u occur, absent any other prior events. For simplicity, one traditionally assumes a uniform background rate in time. $\boldsymbol{K} \in \mathbb{R}^{U\times U}$ is the triggering matrix, where $K_{u,v}$ is the expected number of events of index v that are triggered by one event of index u. This triggering effect, in this temporal-only case, is closely related to Granger causality [Gra69]. In fact, subprocess u does not Granger-cause subprocess v if and only if $K_{u,v} = 0$ [EDD17]. Similarly, for stationarity and subcriticality, $\boldsymbol{K}$ needs to satisfy $\|\boldsymbol{K}\| < 1$, where $\|\boldsymbol{K}\|$ is the spectral norm of $\boldsymbol{K}$. We show an example of fitting the gang crime data above to this multivariate model in fig. 2.1 (B). Gang crimes assigned to two different intervention conditions are modeled as two interacting point processes.

2.1.2 Spatial Point Processes

While the major theory of point processes centers around the temporal dynamics, spatial point process (SPP) models [Dig83] are established in forestry and seismology, focusing on the stationary and isotropic case. We define the (first-order) *intensity function* $\lambda(x)$, which is the expected rate of the accumulation of points around a particular spatial location x. We write

$$\lambda(x) = \lim_{|\Delta x|\downarrow 0} \frac{\mathbb{E}\left[S(\Delta x)\right]}{|\Delta x|}\,, \tag{2.4}$$

where Δx is a small ball in the metric space, e.g. the Euclidean space $\mathbb{R}^n$, with the centre x and measure $|\Delta x|$. The second-order intensity function is naturally defined as

$$\lambda_{(2)}(x, y) = \lim_{|\Delta x|,|\Delta y|\downarrow 0} \frac{\mathbb{E}\left[S(\Delta x)S(\Delta y)\right]}{|\Delta x||\Delta y|}\,, \tag{2.5}$$

measuring the chance of points co-occurring in both Δx and Δy. Normalizing this leads to the *pair-correlation function* $g(x, y) = \lambda_{(2)}(x, y)/\lambda(x)\lambda(y)$. Then $g(x, y) > 1$ indicates that points are more likely to form clusters than the simple Poisson process where $g(x, y) = 1$.

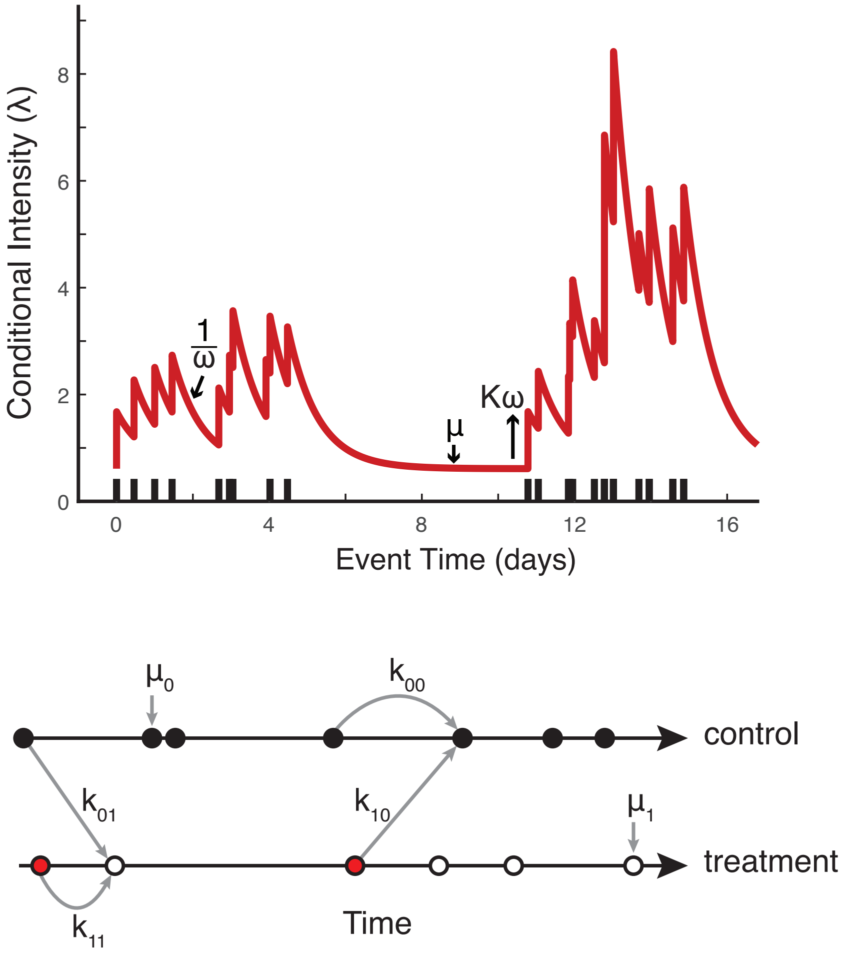


Figure 2.1: Self-exciting point-process models capture the dynamics of gang violent crime events. A temporal self-exciting point-process model $\lambda(t) = \mu + \sum_{t_i<t} Kg(t - t_i)$ with exponential kernel $g(t) = \omega e^{-\omega t}$ (Top). Non-retaliatory gang crimes assigned to each condition arise spontaneously at rate μ_j. Retaliations assigned to each condition may be triggered through separate pathways (Bottom). The parameter k_{ij} is an estimate of the average number of retaliations of type j triggered by a single crime of type i. The parameters k_{11} and k_{01} link previous crimes assigned to the treatment and baseline interventions, respectively, to retaliations subsequently assigned to the treatment intervention. The parameters k_{00} and k_{10} link previous crimes assigned to the baseline and treatment interventions, respectively, to retaliations subsequently assigned to the baseline intervention. If treatment interventions (red events) reduce the risk of gang retaliation, then we expect $k_{11} < k_{01}$ and $k_{10} < k_{00}$.

Common models in SPPs include the Poisson process with a non-stationary rate $\lambda(x)$, and the Cox process with a nonnegative-valued *intensity process* $\Lambda(x)$, which is also a stochastic process. Cox processes conditional on a realization of the intensity process $\Lambda(x) = \lambda(x)$ are Poisson processes with intensity $\lambda(x)$. To model the aggregated points patterns, Poisson cluster (Neyman–Scott) processes generate parent events from a Poisson process. Then each parent independently generates a random number of offspring. The relative positions of these offspring to the parent are distributed according to some p.d.f $K_\sigma(x)$ in space [Dig83]. Many point-process models, including most Cox processes, are in fact Poisson cluster processes. The duality between Cox processes and cluster processes is widely used to construct Cox process models. For example, the kernel-based intensity process $\Lambda(x) = \sum_{i=1}^{\infty} K_\sigma(x - x_i)$ with x_i from a Poisson process, is essentially a Poisson cluster process. The number of offspring is from a Poisson distribution with $\lambda = 1$ and the relative position distribution is $K_\sigma(x)$. Repulsive SPPs, on the other hand, model that nearby points of the process tend to repel each other. Higher-order intensities are often considered in this case, such as determinantal point processes.

Alternatively, if we are more interested in the realization intensity $\lambda(x)$ than the mechanical interpretation, the trans-Gaussian Cox process provides a tractable way to construct the Cox process using a nonlinear transformation on a Gaussian process. Popular choices for $\Lambda(x)$ include the log-Gaussian Cox process (LGCP) and the permanental process. Recent papers on Cox processes have been extensively focused on the cases that are modulated via the Gaussian random field, due to its capability in modeling the intensity and pair-correlations between subprocesses. We aim to develop a more explicit approach to model interactions for fast inference and the generalization ability for new subprocesses.

2.1.3 Spatiotemporal Point Processes

Many real-world data sets include not only timestamps but also accompanying spatial information, which can be particularly important for correctly inferring and understanding the dynamics associated with such data [Bar18]. In earthquakes, for example, most aftershocks usually occur geographically near the mainshock [Oga98]. In online social media, if two people often check-in at the same location at closely proximate times, there is more likely to be a connection between them than if such "joint check-ins" occur rarely [CML11]. These situations suggest that it is important to examine spatiotemporal point processes, rather than just temporal ones. Indeed, there are myriad applications of spatiotemporal Hawkes processes, including crime forecasting [MSB11], the detection of anomalous seismicity [Oga98], and inference of Twitter topics [LMY16].

We characterize a spatiotemporal point process $S(t, x, y)$ via its conditional intensity $\lambda(t, x, y)$, which is the expected rate of the accumulation of points around a particular spatiotemporal location. Given the history $\mathcal{H}_t$ of all points up to time t, we write

$$\lambda(t, x, y) = \lim_{\Delta t, \Delta x, \Delta y \downarrow 0} \left(\frac{\mathbb{E}\left[S\{(t, t + \Delta t) \times (x, x + \Delta x) \times (y, y + \Delta y)\} | \mathcal{H}_t\right]}{\Delta t \, \Delta x \, \Delta y} \right) .$$

For the purpose of modeling earthquakes, Ogata [Oga98] used a self-exciting point process with a conditional intensity of the form

$$\lambda(t, x, y) = \mu(x, y) + K \sum_{t > t_i} g(x - x_i, y - y_i, t - t_i) .$$

In this setting, if an earthquake occurs, aftershocks are more likely to occur locally in time and space. The choice of the triggering kernel $g(t, x, y)$ is inspired by physical properties of earthquakes. For example, Ogata [Oga98] used a modified Omori formula (a power law) [Oga88] to describe the frequency of aftershocks per unit time. In sociological applications, there is no direct theory to indicate appropriate choices for the kernel function. Some researchers have chosen specific kernels (e.g. exponential kernels) that are easy to compute. For example, Tita et al. [CGB14] used a spatiotemporal point process to infer missing information about event participants. They modeled interactions between event participants as a combination of a spatial Gaussian mixture model and a temporal Hawkes process with an exponential kernel. A key problem is how to justify kernel choices in specific applications. Another benefit of point-process modeling is that we can distinguish the clustering effects from the background or triggering via declustering. See section 3.3.4 for more details on declustering. We fit the spatiotemporal point process above to gang crimes in South Los Angeles in fig. 2.2 and show the declustering result.

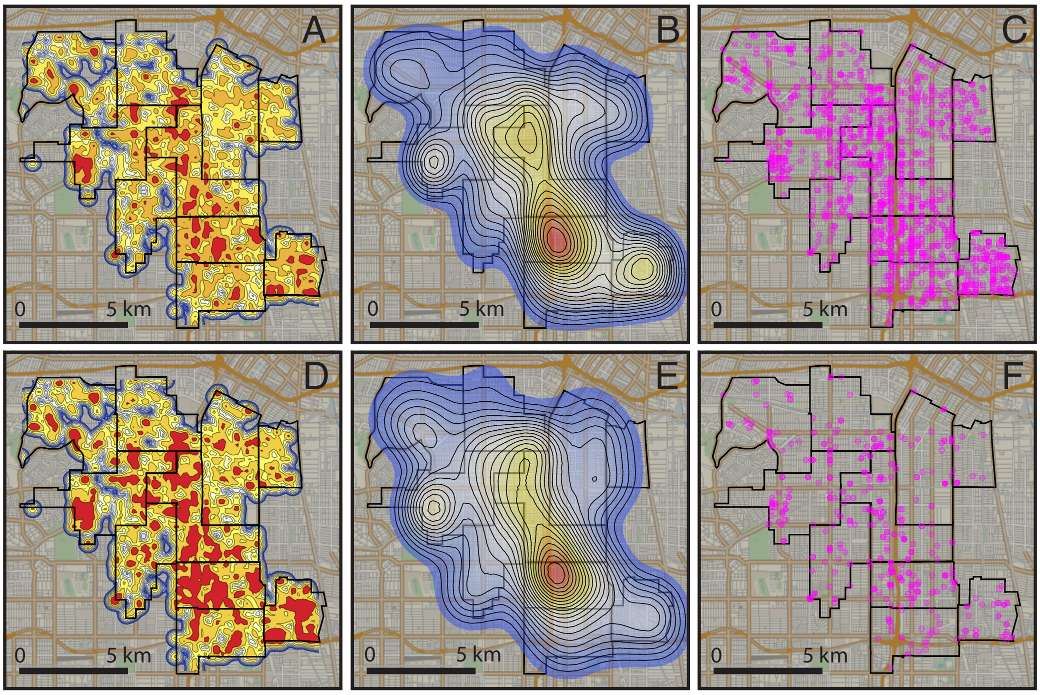


Figure 2.2: Spatiotemporal point process intensities of gang crimes in South Los Angeles. (A). The log of background intensity function μ for gang violent crimes mapped over space. (B) Contour plot of the density of background gang aggravated assaults and homicides determined by declustering. (C) Point locations of background gang aggravated assaults and homicides determined by declustering. (D) The log of spatiotemporal self-excitation of retaliation $\lambda - \mu$ mapped over space. (E) Contour plot of the density of retaliatory gang aggravated assaults and homicides determined by declustering. (F) Point locations of retaliatory gang aggravated assaults and homicides determined by declustering. Boundaries for ten GRYD IR Zones [BSY17] in South Los Angeles are outlined in black.

2.1.4 Model Inference

Inference methods for point processes are mainly based on the order statistics or likelihood function. The order statistics are often estimated nonparametrically, such as the kernel estimator [Dig85] of the intensity function. For the likelihood-based inference, we assume that one observes events $X = \{t_i\}_{i=1}^N$ of the underlying point process over the area R. The log-likelihood for the point process with intensity $\lambda(t)$ over a certain space R is

$$\log p(X|\Theta) = \sum_{i=1}^{N} \log(\lambda(t_i)) - \int_R \lambda(t)\,\mathrm{d}t\,. \tag{2.6}$$

The integration term is the log void probability and can be viewed as a normalization term for the likelihood.

In the multivariate temporal point process case, one can estimate the set of parameters Θ by minimizing the negative log-likelihood function

$$-\log(L(\Theta)) = -\sum_{k=1}^{N} \log(\lambda_{u_k}(t_k)) + \sum_{u=1}^{U} \int_0^T \lambda_u(t)\mathrm{d}t\,, \tag{2.7}$$

where $\log(x)$ denotes the natural logarithm of x. Recall that u_k is the point process associated with event k. There are several variants of the MLE for the multivariate Hawkes process. One is to add regularization terms to Equation eq. (2.7) to improve the accuracy of parameter estimation. Lewis and Mohler [LM11] used maximum-penalized likelihood estimation, which enforces some regularity on the model parameters, to infer Hawkes processes. Linderman et al. [LA14] added random-graph priors on $\mathbf{K}$ and developed a fully Bayesian multivariate Hawkes model. See [MRW18] for theoretical guarantees on inferring Hawkes processes with a regularizer. Another research direction is to speed up the parameter estimation of point-process models. For example, Hall et al. [HW16] tried to learn the triggering matrix $\mathbf{K}$ via an online learning framework for streaming data. In a very recent paper, Achab et al. [ABG17] developed a fast moment-matching method (instead of using a likelihood-based method) to estimate the matrix $\mathbf{K}$.

For Cox processes, the likelihood is the expectation over the Poisson likelihood above. It is difficult to directly integrate over the distribution of Λ. Monte Carlo methods [AMM09] are commonly used to approximate the expectation. To improve the scalability of the expensive sampling, many methods such as variational inference [LGO15], Laplace approximation [WR06] and reproducing kernel Hilbert spaces [FTS17] are proposed.

2.2 Point Processes on Real-world Problems

Multivariate point-process models have a wide range of applications. In network reconstruction, for example, one seeks to infer the relationships (i.e. edges) and the strengths of such relationships (i.e. edge weights) among a set of entities (i.e. nodes). When modeling the relationships in a network, it is more appropriate to use a multivariate point process than a univariate one. To utilize covariates in a criminal record, we need to extend the univariate self-exciting point-process model [MSB11] to the multivariate setting. For crime applications, each process could represent different gangs, crime types, zip code areas or police intervention attempts. Via defining subprocesses as different entities, the multivariate model can be applied to crime forecasting, gang network inference as well as causality estimation. Multivariate Hawkes processes are closely related to Granger causality (Granger, 1969). As a result, one can hypothesize that the mutual effect parameters K_{uv} can reflect real-world connections among different processes. For example, if each gang is a single point process, $K_{uv} > 0$ implies crimes in gang u will lead to crimes in gang v.

In a previous point-process model of crime [RG18], every crime increases a local crime risk that decays exponentially in time and diffuses as a Gaussian distribution in space. To increase the generalization ability, one can use a model-independent approach [ML08] to avoid the selection of a specific decay function (kernel) for different applications. In this setting, the kernel is assumed to be a stepwise constant function and the values will be learned from real-world data. EM-type algorithms are widely used in the inference of point-process models. However, this approach is far from satisfying [Sch18]. A specific limit is computation complexity. EM has a $O(N^3)$ time complexity and a $O(N^2)$ storage requirement, which is not acceptable for larger data sets. It is important to improve the scalability of the multivariate point-process model.

Table 2.1: Notation.

Notation	Definition or Descriptions
$S(x)$	counting measure on a metric space R
N_u	the number of events of subprocess u
$\lambda_u(x)$	intensity function of a subprocess u
$\Lambda_u(x)$	intensity process of a subprocess u
$K_\sigma(x)$	kernel function
U	the number of subprocesses on the space
$\mathbf{U}_b$	a set subprocesses with index in b
X	events set
X_u	observed events of subprocess u
X_b	all the observed events in a batch of subprocesses
x_i	embedding/location of the i_{th} event
Y_i^u	hidden variables indicate whether the subprocess u includes the i_{th} event
z_u	a hidden variable represents subprocess u
p_i^u	probability of the i_{th} event occurs in subprocess u
ϕ, θ	neural network parameters
$h_u(x)$	normalized density

Another application of multivariate models is the evaluation of a gang intervention program. Retaliation propels gang violence. Spontaneous attacks resulting from chance encounters between rivals, or situational interactions that challenge gang territory or reputation can trigger cycles of tit-for-tat reprisals. Yet it has been difficult to determine if interventions that seek to reduce the likelihood of retaliation translate into lower rates of gang crime. One can use a multivariate spatiotemporal point process to quantify the magnitude of retaliation arising from gang crimes given two distinct types of post-event interventions. The methods are well-suited to the analysis of real-world interventions where there is an interaction between outcomes. Our preliminary analysis [BSY17] of interventions in Los Angeles indicates that efforts to control rumors and engage impacted families, undertaken in the immediate aftermath of gang violent crimes, reduce the contagious spread of violence. These findings [BYS18, Wan18] have important implications for the design, implementation, and evaluation of gang violence prevention programs.

Finally, we want to clarify the difference between point-process models and time-series approaches. Point processes are generative models that are continuous in time. For time series analysis, however, it first discretizes the events into time bins and aggregates them together. We focus on the point-process models due to its convenient and natural representation of crime and social media events with precise spatiotemporal stamps. We also worked on specific problems in seismology [YTM19] and medical imaging [YCI17, DGQ18], which are more appropriate for time-series methods.

CHAPTER 3

Multivariate Hawkes Processes and Network Reconstruction

There is often latent network structure in spatial and temporal data, and the tools of network analysis can yield fascinating insights into such data. In this chapter, we propose a nonparametric and multivariate version of a spatiotemporal Hawkes process. Spatiotemporal Hawkes processes have been used previously to study numerous topics, including crime [MSB11], social media [LMY16], and earthquake forecasting [FSG16]. In our model, each node in a network is associated with a spatiotemporal Hawkes process. The nodes can "trigger" each other, so events that are associated with one node increase the probability that there will be events associated with the other nodes. We measure the extent of such mutual-triggering effects using a $U \times U$ "triggering matrix" $\mathbf{K}$, where U is the number of nodes. If one considers an exclusively temporal scenario, a point process u does not "cause" (in the Granger sense [Gra69]) a point process v if and only if $\mathbf{K}(u, v) = 0$ [EDD17]. Because triggering between point processes reflects an underlying connection, one can try to recover latent relationships in a network from $\mathbf{K}$. Such triggering should decrease with both distance and time according to some spatial and temporal kernels. In this work, instead of assuming exponential decay [FSS16] or some other distribution [LA14, CGB14], we adopt a nonparametric approach [ML08] to learn both spatial and temporal kernels from data using an expectation-maximization-type (EM-type) algorithm [VS08]. Recently, Chen et al. [CSS17] also studied Hawkes processes with a nonparametric approach, although they only considered exclusively temporal kernels.

This model helps fill a gap in the literature on incorporating spatial information into multivariate self-exciting point processes [Rei18a]. To our knowledge, it is the first method that uses a multivariate spatiotemporal Hawkes process with a nonparametric method to estimate a triggering kernel. Inspired by the successful employment of spatiotemporal univariate Hawkes processes in earthquake forecasting [FSG16, ML08] and predictive policing [MSM15], our work extends these ideas to multivariate Hawkes processes and uses these ideas in an application to network reconstruction. We illustrate our approach using both synthetic networks and networks that we construct from real-world data sets (a location-based social-media network, a narrative of crime events, and violent gang crimes). Our approach outperforms other recent point-process network-reconstruction methods [FSS16, LA14] on both synthetic and real-world data sets with spatial information. Additionally, our results illustrate the importance both of incorporating spatial information and of using nonparametric kernels. Although we assume that the relationships between nodes are time-independent, our model still recovers a causal structure among events in synthetic data sets. Based on this information, we build event-causality networks on data sets about violent crimes of gangs and examine gang-retaliation patterns using motif analysis.

This chapter proceeds as follows. In section 2.1, we review self-exciting point processes and recent point-process methods for network reconstruction. In section 3.1, we introduce our nonparametric spatiotemporal model and our approaches for model estimation and simulations. In section 3.3, we compare our model with others on both synthetic and real-world data sets. We construct our two examples of the latter from (1) a location-based social-media platform and (2) crime topics. We conclude in section 3.4.

3.1 Spatiotemporal Models for Network Reconstruction

Many network-reconstruction methods, such as the ones in [LA14, FSS16, CSS17], have used self-exciting point processes to infer time-independent relationships (i.e. edges) between entities (i.e. nodes) with corresponding (exclusively) temporal point processes. Entity (i.e. process) u is adjacent to entity v if $\mathbf{K}(u, v) > 0$, where one estimates the triggering matrix $\mathbf{K}$ from the data. Entity u is not adjacent to v if the former's point process does not cause the latter's point process in time (in the Granger sense [EDD17]). For many problems, it is desirable (or even crucial) to incorporate spatial information [Cre15, Bar18]. For example, spatial information is an important part of online fingerprints in human activity, and it has a significant impact on most other social networks. In crime modeling, for instance, there is a "near repeat" phenomenon in crime locations, indicating the necessity of including spatial information. Specifically, the spatial neighborhood of an initial burglary has a higher risk of repeat victimization than more-distant locations [SBB10]. In our work, we propose multivariate spatiotemporal Hawkes processes to infer relationships in networks and provide a novel approach for analyzing spatiotemporal dynamics.

It is also important to consider the assumptions on triggering kernels for Hawkes processes. In seismology, for example, researchers attempt to use an underlying physical model to help determine a good kernel. However, it is much more difficult to validate such models in social networks than for physical or even biological phenomena [PH17]. The content of social data is often unclear, and typically there is little understanding of the underlying mechanisms that produce them. With less direct knowledge of possible triggering kernels, it is helpful to employ a data-driven approach for kernel selection. Using a kernel with an inappropriate decay rate may lead to either underestimation or overestimation of the elements in the triggering matrix $\mathbf{K}$, which may also include false negatives or false positives in the inferred relationships between entities.

Therefore, we ultimately choose to use a nonparametric approach to learn triggering kernels in various applications to avoid *a priori* assumptions about a specific parametrization. Specifically, we use histogram estimators with EM-type algorithms to maximize the likelihood, as has been done in applications in seismology and crime modeling [ML08, VS08, LM11]. An alternative approach [CSS17] is a penalized regression scheme. With such a scheme, one can approximate a kernel as a sum of basis functions and minimize the squared-error loss of the intensity function with a group-lasso penalty. Although some of the goals of previous Hawkes-process models and our model are similar, there are many key differences. For example, [CSS17] focused on the theoretical development of Hawkes processes with inhibition; they did not consider spatial information. By contrast, the purpose of our model is to investigate spatiotemporal data sets from social media and crime with a self-exciting

Hawkes process. The theoretical guarantees of our model arise from the consistency of the Hawkes process with the non-negative triggering matrix and kernels (which is necessary for the cluster representation of such processes).

A multivariate spatiotemporal Hawkes process is a sequence $\{(t_i, x_i, y_i, u_i)\}_{i=1}^{N}$ with N events, where t_i and (x_i, y_i) are spatiotemporal stamps and u_i is the point-process index of event i. Each of the U nodes is a *marginal process*. The conditional intensity function for node u is

$$\lambda_u(t, x, y) = \mu_u(x, y) + \sum_{t>t_i} K_{u_i u} g(x - x_i, y - y_i, t - t_i)\,. \tag{3.1}$$

The above Hawkes process assumes that each node u has a background Poisson process that is constant in time but inhomogeneous in space with conditional intensity $\mu_u(x, y)$. There is also self-excitation, as past events increase the likelihood of subsequent events. We quantify the impact that events associated with node u_i have on subsequent events of node u_j with spatiotemporal kernels and the element $\mathbf{K}(u_i, u_j) = K_{u_i u_j}$ of the triggering matrix.

3.1.1 A Parametric Model

We first propose a multivariate Hawkes process with a specific parametric form. We use this model to generate spatiotemporal events on synthetic networks and provide a form of "ground truth" that we can use later.

The background rate μ_u and the triggering kernel g for Equation eq. (3.1) are given by

$$\begin{aligned} g(x, y, t) &= g_1(t) \times g_2(x, y) = \omega \exp\left(-\omega t\right) \times \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)\,, \\ \mu_u(x, y) &= \sum_{i=1}^{N} \frac{\beta_{u_i u}}{2\pi\eta^2 T} \times \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{2\eta^2}\right)\,. \end{aligned}$$

For simplicity, we use exponential decay in time [Oga88] and a Gaussian kernel in space [Moh14]. We let T denote the time window of a data set; $K_{u_i u}$ denote the mean number of the events in process u that are triggered by each event in process u_i; the quantity $\beta_{u_i u}$ denote the extent to which events in process u_i contribute to the background rate for events in process u; and σ and η, respectively, denote the standard deviations in the triggering kernel and background rate. The value of σ determines the spreading scale of the triggering in space.

3.1.2 A Nonparametric Model

With the conditional intensity in Equation eq. (3.1), we estimate the triggering kernel $g(x, y, t) = g_1(t) \times g_2(x, y)$ nonparametrically using histogram estimators [ML08]. We assume that g_2 is isotropic, which entails that $g_2(x, y) = g_2(r)$, where $r = \sqrt{x^2 + y^2}$. We let $h(r)$ be the spatial triggering kernel in the radial coordinate: $h(r) = 2\pi r g_2(r)$. We extend the background rate that was proposed in [FSG16] to the multivariate case and write

$$\mu_u(x, y) = \gamma_u \tau(x, y) = \frac{\gamma_u}{T} \sum_{i=1}^{N} \frac{p_{ii}}{2\pi d_i^2} \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{2d_i^2}\right), \tag{3.2}$$

where γ_u is the background intensity of process u and p_{ii} is the probability that event i is a background event (i.e. it is not triggered by any event). We compute d_i by determining the radius of the smallest disk centered at (x_i, y_i) that includes at least n_p other events and is at least as large as some small value ϵ (which represents the error in location).

Once we fit the model to spatiotemporal data, the triggering matrix $\mathbf{K}$ gives our inferences for the underlying relationships between entities. For two entities u and v, the matrix element $\mathbf{K}(u, v)$ indicates a mixture of temporal causality and spatial dependence between them. In inferring latent relationships in a network, we assume that entity u is not related to v if $\mathbf{K}(u, v) = 0$. We threshold the matrix $\mathbf{K}$ at a certain level: we set elements that are smaller than the threshold value to 0; and we either maintain the values of larger or equal elements to obtain a weighted network, or we set them to 1 to produce an unweighted network. We use $\tilde{\mathbf{K}}$ to denote the thresholded matrix $\mathbf{K}$. We interpret that there is no relation between two nodes u and v if $\tilde{\mathbf{K}}(u, v) = \tilde{\mathbf{K}}(v, u) = 0$.

3.2 Model Estimation

We use an EM-type algorithm [VS08] to estimate the parameters and kernel functions of our model. This EM-type algorithm gives an iterative method to find maximum-likelihood estimates of the parameters. We assume that the original model depends on unobservable latent variables. Suppose that we have data X and want to estimate parameters Θ. One can view the likelihood function $L(\Theta; X)$ as the marginal likelihood function of $L(\Theta; Y, X)$, where Y is a latent variable. We call $L(\Theta; Y, X)$ the "complete-data likelihood function" and $L(\Theta; X)$ the "incomplete-data likelihood function". Because both Y and $L(\Theta; Y, X)$ are random variables, we cannot estimate them directly. Therefore, we consider the following expectation function:

$$\begin{aligned} Q(\Theta, \Theta^{i-1}) &= \mathbb{E}\left[\log(L(\Theta; Y, X)) | X, \Theta^{i-1}\right] \\ &= \int \log(L(\Theta; Y, X)) f(Y|X, \Theta^{i-1}) \mathrm{d}Y \,, \end{aligned} \tag{3.3}$$

where $f(Y|X, \Theta^{i-1})$ is the probability density function of Y, given the data X and Θ^{i-1}. We update parameters by solving the following equation:

$$\hat{\Theta}^i = \arg\max_{\Theta} Q(\Theta, \Theta^{i-1}) \,.$$

3.2.1 Parametric Model

The log-likelihood for the parametric model defined in Equation eq. (3.1) in a spatial region R and time window $[0, T]$ is

$$\log(L(\Theta; X)) = \sum_{k=1}^{N} \log(\lambda_{u_k}(t_k)) - \sum_{u=1}^{U} \iint_R \int_0^T \lambda_u(t) \,\mathrm{d}t \,\mathrm{d}x \,\mathrm{d}y \,. \tag{3.4}$$

We define random variables Y_{ij} and Y_{ij}^b using the approach from [Moh14]. If event j triggers event i via the kernel g, then $Y_{ij} = 1$; otherwise, $Y_{ij} = 0$. The equality $Y_{ij}^b = 1$ indicates that event i is triggered by event j at a background rate of μ. We define two expectation matrices: $\mathbf{P}(i, j) = p_{ij} = \mathbb{E}[Y_{ij}]$ and $\mathbf{P}^b(i, j) = p_{ij}^b = \mathbb{E}[Y_{ij}^b]$. We convert the incomplete-data log-likelihood function in eq. (3.4) into the following complete-data log-likelihood function:

$$\begin{aligned} \log(L(\Theta; X, Y)) = &\sum_{j<i} Y_{ij} \log\left(K_{u_j u_i} g(t_i - t_j, x_i - x_j, y_i - y_j)\right) - \sum_{u=1}^{U} \sum_{i=1}^{N} \beta_{u_i u} \\ &- \sum_{u=1}^{U} \sum_{i=1}^{N} K_{u_i u} \left(1 - e^{-w(T - t_i)}\right) + \sum_{i=1}^{N} \sum_{j=1}^{N} Y_{ij}^b \log(\mu_{u_i}) \,. \end{aligned}$$

We then calculate the expectation function using eq. (3.3) to obtain

$$\begin{aligned}
Q(\Theta) =& \sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij}^{b} \log\left(\frac{\beta_{u_j u_i}}{2\pi\eta^2 T}\exp\left(-\frac{(x_i-x_j)^2+(y_i-y_j)^2}{2\eta^2}\right)\right) - \sum_{u=1}^{U}\sum_{i=1}^{N}\beta_{u_i u} \\
&+ \sum_{j<i} p_{ij} \log\left(\omega K_{u_j u_i} e^{-\omega(t_i-t_j)} \frac{1}{2\pi\sigma^2}\exp\left(-\frac{(x_i-x_j)^2+(y_i-y_j)^2}{2\sigma^2}\right)\right) \\
&- \sum_{u=1}^{U}\sum_{i=1}^{N} K_{u_i u}\left(1-e^{-w(T-t_i)}\right) .
\end{aligned}$$

We perform the maximization step of the EM-type algorithm (a projected gradient ascent) [LM11] directly by taking derivatives with respect to the parameters and setting them to 0. For the expectation step, we use the "optimal" parameter values from the prior maximization step to update the probabilities p_{ij} and p_{ij}^b. By (alternately) iterating the expectation and maximization steps, we obtain algorithm 1 for the parametric model. For initialization, we sample Θ^0, p_{ij}, and p_{ij}^b uniformly at random. Note, additionally that $p_{ij} = 0$ for $i > j$.

3.2.2 Nonparametric Model

The log-likelihood function of the nonparametric model is the same as that for the parametric model in eq. (3.4). We use a similar approach as before to derive an EM-type algorithm for the nonparametric model. The main differences are that (1) only Y_{ij} are latent variables and $Y_{ii} = 1$ signifies that event i is a background event, whereas $Y_{ji} = 1$ signifies that event i is triggered by event j; and (2) we assume that the triggering kernels $g_1(t)$ and $g_2(r)$ are piecewise constant functions. Note that the elements of the expectation matrix are $\mathbb{E}[Y_{ij}] = \mathbf{P}(i,j) = p_{ij}$. We discretize space and time into n_t^{bins} temporal bins and n_r^{bins} spatial bins, and the kernel takes a constant value in each spatiotemporal bin.

To formally present our EM-type algorithm (see algorithm 2), we borrow notation from [FSG16]. Let C_k denote the set of event pairs (i,j) for which $t_j - t_i$ belongs to the k^{th} temporal bin, D_k denote the set of event pairs (i,j) for which r_{ij} (the distance between nodes i and j) belongs to the k^{th} spatial bin, N_u denote the number of events that include node u, the parameter Δt_k denote the size of the k^{th} temporal bin, and Δr_k denote the size of the k^{th} spatial bin.

Algorithm 1 EM-type Algorithm for the Parametric Spatiotemporal Hawkes Model

```
Inputs: point process: $\{(u_i, t_i, x_i, y_i)\}_{i=1}^{N}$; initial guesses for parameters: $\Theta^{(0)} = \left(\{K_{uv}^{(0)}\}_{u,v=1}^{U}, \{\beta_{uv}^{(0)}\}_{u,v=1}^{U}, \sigma^{(0)}, \omega^{(0)}\right)$ and $\{p_{ij}^{(0)}\}_{i,j=1}^{N}$, $\{p_{ij}^{b,(0)}\}_{i,j=1}^{N}$; termination threshold: $\epsilon$.
Outputs: model parameters $\Theta = \left(\{K_{uv}\}_{u,v=1}^{U}, \{\beta_{uv}\}_{u,v=1}^{U}, \sigma, \omega\right)$.
Initialize $\delta = 1$ and $k = 0$.
while $\delta > \epsilon$ do
    Let $\eta^{2,(k)}$ and $\sigma^{2,(k)}$ be the value of $\eta^2$ and $\sigma^2$ at the $k^{\text{th}}$ iteration.
    Expectation step: for $i, j \in \{1, 2, \cdots, N\}$,
    $p_{ij}^{(k)} = \left(K_{u_j u_i} g\left(t_i - t_j, x_i - x_j, y_i - y_j\right)\right) / \lambda\left(x_i, y_i, t_i\right)$.
    $p_{ij}^{b,(k)} = \beta_{u_j u_i}^{(k)} \exp\left(-\frac{(x_j - x_i)^2 + (y_j - y_i)^2}{2\eta^{2,(k)}}\right) / \left(2\pi\eta^{2,(k)} T \lambda(x_i, y_i, t_i)\right)$.
    Maximization step: for $u, \hat{u} \in \{1, 2, \cdots, U\}$,
    $\omega^{(k+1)} = \frac{\sum_{j<i} p_{ij}^{(k)}}{\sum_{j<i} p_{ij}^{(k)}(t_i - t_j) + \sum_{u=1}^{U} \sum_{i=1}^{N} K_{u_i u}(T - t_i) e^{-\omega(T - t_i)}}$,
    Let $N_u$ denote the number of events in point process $u$; and let $i_l^u$, with $l \in \{1, \ldots, N_u\}$, index the events for process $u$.
    We update
    $K_{\hat{u}u}^{(k+1)} = \sum_{l=1}^{N_u} \sum_{t_{i_{\hat{l}}^{\hat{u}}} < t_{i_l^u}} p_{i_l^u i_{\hat{l}}^{\hat{u}}}^{(k)} \Big/ \sum_{l=1}^{N_{\hat{u}}} \left(1 - \exp\left[-w\left(T - t_{i_{\hat{l}}^{\hat{u}}}\right)\right]\right)$,
    $\beta_{\hat{u}u}^{(k+1)} = \sum_{l=1}^{N_u} \sum_{\hat{l}=1}^{N_{\hat{u}}} p_{i_l^u i_{\hat{l}}^{\hat{u}}}^{b,(k)} / N_{\hat{u}}$.
    $\sigma^{2,(k+1)} = \sum_{i,j=1}^{N} \left(p_{ij}^{b,(k)} + p_{ij}^{(k)}\right)\left((x_i - x_j)^2 + (y_i - y_j)^2\right) \Big/ \sum_{i,j=1}^{N} 2\left(p_{ij}^{b,(k)} + p_{ij}^{(k)}\right)$.
    $\eta^{2,(k+1)} = \sigma^{2,(k+1)}$.
    $\delta = \|\Theta^{(k)} - \Theta^{(k+1)}\|$.
    $k = k + 1$.
end while
```

Algorithm 2 EM-type Algorithm for the Nonparametric Spatiotemporal Hawkes Model

```
**Inputs**: point process: $\{(u_i, t_i, x_i, y_i)\}_{i=1}^N$; initial guesses of parameters: $\{K_{uv}^{(0)}\}_{u,v=1}^U$ and $\{p_{ij}^{(0)}\}_{i,j=1}^N$; termination threshold: $\epsilon$.
**Outputs**: model parameters: $\{K_{uv}\}_{u,v=1}^U$; triggering probability between events: $\{p_{ij}\}_{i,j=1}^N$; temporal triggering kernel: $g_1$; spatial triggering kernel: $g_2$.
Initialize $\delta = 1$ and $\eta = 0$.
**while** $\delta > \epsilon$ **do**
    Update $\tau^\eta(x, y)$ using eq. (3.2)
    $\gamma_u^{(\eta)} = \sum_{u_i=u} p_{ii}^{(\eta)} \big/ Z^{(\eta)}$, where $Z^{(\eta)}$ satisfies $\int_0^T \iint_R \tau^\eta(x, y) \mathrm{d}s\, \mathrm{d}t = Z^{(\eta)}$ on a bounded spatial domain $R$ for $u \in \{1, \dots, U\}$.
    $K_{uv}^{(\eta)} = \sum_{u_i=u} \sum_{u_j=v} p_{ij}^{(\eta)} \big/ N_u$ for $u, v \in \{1, \dots, U\}$.
    $g_1^{(\eta)}(t) = \sum_{i,j \in C_k} p_{ij}^{(\eta)} \Big/ [\Delta t_k \sum_{i<j} p_{ij}^{(\eta)}]$ for $t$ in the $k^{\text{th}}$ temporal bin.
    $h^{(\eta)}(r) = \sum_{i,j \in D_k} p_{ij}^{(\eta)} \Big/ [\Delta r_k \sum_{i<j} p_{ij}^{(\eta)}]$ for $r$ in the $k^{\text{th}}$ spatial bin. Set $g_2^{(\eta)}(r) = h^{(\eta)}(r) \big/ (2\pi r)$.
    $p_{ij}^{(\eta+1)} = K_{u_i u_j}^{(\eta)} g_1^{(\eta)}(t_j - t_i) g_2^{(\eta)}(r_{ij})$ for $i < j$ and $p_{jj}^{(\eta+1)} = \mu_{u_j}^{(\eta)}(x_j, y_j)$.
    Normalize $p_{ij}^{(\eta+1)}$ so that $\sum_{i=1}^N p_{ij}^{(\eta+1)} = 1$ for any $j$.
    $\delta = \max_{i,j} \|p_{ij}^{(\eta+1)} - p_{ij}^{(\eta)}\|$ and $\eta = \eta + 1$.
**end while**
```

3.2.3 Simulations

To generate synthetic data for model comparisons, we need to simulate self-exciting point processes with the conditional intensity in eq. (3.1) for each process u. We use the branching structures [ZOV04] of self-exciting point processes to develop algorithm 3 for our simulations.

3.3 Numerical Experiments and Results

We apply our algorithms to both synthetic and real-world data sets to demonstrate the usefulness of (1) incorporating spatial information and (2) our nonparametric approach. We consider a synthetic data set in section 3.3.1, a social-media data set from Gowalla in section 3.3.2, a crime-topic network data set in section 3.3.3, and a violent gang-crime data set in section 3.3.4. Using the first three of these data sets, we compare our nonparametric spatiotemporal Hawkes model ("Nonparametric Hawkes") with the Bayesian Hawkes model[1] from [LA14] ("Bayesian Hawkes"), the exclusively temporal Hawkes model with kernel $g(t) = \omega \exp(-\omega t)$ from [FSS16] ("Temporal Hawkes"), and the parametric spatiotemporal Hawkes model that we detailed in section 3.1.1 ("Parametric Hawkes"). We make comparisons by examining how well the following properties are recovered when we infer a triggering matrix: (1) symmetry and reciprocity; (2) existence of edges; and (3) community structure. We also demonstrate the ability of both our parametric and nonparametric algorithms to infer the triggering kernel g. Using the fourth data set (see section 3.3.4), we study a network of crime events using a violent gang-crime data set. We examine relations between crime events and repeated triggering patterns.

[1] Linderman and Adams [LA14] used a sparse log-Gaussian Cox process to model the background rate, a logistic-normal density for the temporal kernel, an Aldous–Hoover graph prior for the existence of entries in $\mathbf{K}$, and gamma prior for the weights of those entries. In all of our experiments, we use the default hyperparameters and priors that come with the published code at `https://github.com/slinderman/pyhawkes`.

Algorithm 3 Simulation of a Multivariate Spatiotemporal Hawkes Process

```
**Inputs**: time-window size: $T$; spatial region: $R \subset \mathbb{R}^2$; background rate: $\{\gamma_u\}_{u=1}^{U}$; triggering matrix: $\{K_{uv}\}_{u,v=1}^{U}$; temporal and spatial triggering kernels: $g_1(t)$, $g_2(x, y)$ .
**Output**: point process: $\mathbf{C} = \{(u_i, t_i, x_i, y_i)\}_{i=1}^{N}$ .
Initialize an empty set $\mathbf{C}$ and an empty stack $\mathbf{Q}$.
**Generate background events:**
  Draw $N_u^b$, the number of background events of type $u$, from a Poisson distribution with parameter $\lambda = \gamma_u T$ for each $u \leq U$.
  Add each background event $i \leq \sum_{u=1}^{U} N_u^b$, with its associated tuple $(x_i, y_i, t_i, u_i)$, to the set $\mathbf{C}$ and the stack $\mathbf{Q}$, where we draw $(x_i, y_i, t_i)$ from the uniform spatiotemporal distribution over the time interval $[0, T]$ and a bounded spatial region $R$.
**Generate triggered events:**
  **while** $\mathbf{Q}$ is not empty **do**
    Remove the most recently added element $(x_i, y_i, t_i, u_i)$ from the stack $\mathbf{Q}$.
    Draw $n_i$, the number of events triggered by event $i$, from a Poisson distribution with parameter $\lambda_i = \sum_{u'=1}^{U} K_{u_i u'}$.
    Generate events $(x_k, y_k, t_k, u_k)$ for each $k \leq n_i$ as follows:
      Sample $t_k$, $(x_k, y_k)$, and $u_k$ according to $g_1(t - t_i)$, $g_2(x - x_i, y - y_i)$, and $P(u_k = \tilde{u}) = \frac{K_{u_i \tilde{u}}}{\sum_{v=1}^{U} K_{u_i v}}$, respectively.
      Add $(x_k, y_k, t_k, u_k)$ to the set $\mathbf{C}$.
      **if** $t_k \leq T$ **then**
        Add the element $(x_k, y_k, t_k, u_k)$ to the stack $\mathbf{Q}$.
      **end if**
  **end while**
```

3.3.1 Synthetic Data

We first generate synthetic triggering matrices $\mathbf{K}$ using a weighted stochastic block model (WSBM) [AJC14, Pei19]. We assign a network's nodes to four sets (called "communities") and assign edges to adjacency-matrix blocks based on the set memberships of the nodes. Two of the communities consist of ten nodes each, and the other two communities consist of five nodes each. For each edge, we first draw a Bernoulli random variable to determine whether it exists, and we then draw an exponential random variable to determine the weight of the edge (if it exists). An edge between nodes from the same community exists with probability 0.68, and an edge between nodes from different communities exists with probability 0.2. The decay-rate parameter for the exponential random variable in these two situations is 0.1 and 0.01, respectively. By construction, our triggering matrices are symmetric.

The triggering matrices that we generate in this way are not guaranteed to satisfy the stability condition for Hawkes processes; this condition is that the largest-magnitude eigenvalue of $\mathbf{K}$ is smaller than one [DV07]. When this condition is satisfied, it is almost surely true that each event has finitely many subsequent events as "offspring". In our work, we discard any simulated adjacency matrices that do not satisfy the stability condition, and we generate a new one to replace it. (With our choices of the parameters, we discard about 65% of the generated adjacency matrices.)

With each triggering matrix $\mathbf{K}$, we use algorithm 3 to simulate a multivariate spatiotemporal Hawkes process with our parametric model in section 3.1.1 with $\omega = 0.6$, $\sigma^2 = 0.3$, $T = 250$, $S = [0,1] \times [0,1]$, and a homogeneous value $\gamma_u = 0.2$ for all nodes u. We then reconstruct the underlying networks and the triggering kernels from the simulated data. As a result, Parametric Hawkes serves as a "ground-truth model", and we expect it to have the best performance for synthetic data, given that we use the same model to produce the data.

Symmetry and Reciprocity

As we noted in section 3.3.1, our simulated triggering matrices are symmetric, but our reconstructed adjacency matrices generally are not symmetric. Without prior information about symmetry, measuring deviation from symmetry gives one way to evaluate the performance of our inference methods. We use various reciprocity measures to quantify such deviation.

We conduct two sets of experiments. In the first set, we fix a single synthetic triggering matrix and simulate ten multivariate spatiotemporal Hawkes point processes. We then estimate the triggering matrix $\mathbf{K}$ from each point process using various methods, which we thereby compare with each other. In the second set of experiments, instead of fixing a single triggering matrix, we generate ten different triggering matrices using the same WSBM model and parameters; and we simulate one point process for each triggering matrix.

There is no standard way of measuring reciprocity in a weighted network. In our calculations, we use diagnostics that were proposed in [SPR13] and [AMF12]. First, as in [SPR13], we compute the reciprocated edge weight $K_{uv}^{\leftrightarrow} = \min\{K_{uv}, K_{vu}\}$, and we then calculate a network-level reciprocity score R_1 as the ratio between the total reciprocated weight $W^{\leftrightarrow} = \sum_{u \neq v} K_{uv}^{\leftrightarrow}$ and the total weight $W = \sum_{u \neq v} K_{uv}$. That is, the "reciprocity" is $R_1 := W^{\leftrightarrow}/W$. Second, Akoglu et al. [AMF12] proposed three edge-level measures of reciprocity: (1) the "ratio" $R_{\text{ratio}} := \min\{K_{uv}, K_{vu}\}/\max\{K_{uv}, K_{vu}\}$; (2) "coherence" $R_{\text{coher}} = 2\sqrt{K_{uv}K_{vu}}/(K_{uv}+K_{vu})$; and (3) "entropy" $R_{\text{entropy}} := -r_{uv}\log_2(r_{uv}) - r_{vu}\log_2(r_{vu})$, where $r_{uv} = K_{uv}/(K_{uv} + K_{vu})$. These last three measures of reciprocity are measured at an edge level (as they are defined for a pair of nodes), whereas R_1 is a network-level measure. For each edge-level measure, we obtain a network-level measure by calculating the score for each pair of nodes and then taking a mean over all pairs of nodes. Each of the above quantities gives a score between 0 and 1, where a larger value indicates a stronger tendency for the nodes in a network to reciprocate. In a perfectly symmetric and reciprocal network, each of the four methods gives a value of 1.

In table 3.1, we report the mean reciprocity and the standard deviation over ten simulations with the same triggering matrix. In table 3.2, we report the mean results from ten different triggering matrices. Both spatiotemporal models give higher scores than the exclusively temporal models, which is what we expected, as the temporal models discard spatial information. According to these measures of success, the nonparametric model has the best performance, even over the ground-truth (parametric) model that generated the data.

Table 3.1: Reciprocity of the triggering matrices that we infer using different methods: a nonparametric spatiotemporal Hawkes model, a temporal Hawkes model, a parametric spatiotemporal Hawkes model, and a fully Bayesian Hawkes model. We report the mean and standard deviation (in parentheses) over ten simulations that use the same (ground-truth) triggering matrix.

	Nonparametric	Temporal	Parametric	Bayesian
R_1	0.59 (0.05)	0.29 (0.06)	0.54 (0.03)	0.36 (0.03)
Correlation	0.84 (0.05)	0.36 (0.16)	0.79 (0.05)	0.30 (0.14)
Ratio	0.55 (0.02)	0.37 (0.11)	0.58 (0.02)	0.32 (0.02)
Coherence	0.75 (0.01)	0.63 (0.03)	0.71 (0.02)	0.68 (0.02)
Entropy	0.71 (0.01)	0.59 (0.03)	0.68 (0.02)	0.60 (0.02)

Table 3.2: Reciprocity of the triggering matrices that we infer using different methods: a nonparametric spatiotemporal Hawkes model, a temporal Hawkes model, a parametric spatiotemporal Hawkes model, and a fully Bayesian Hawkes model. We report the mean and standard deviation (in parentheses) over ten simulations, each with a different (ground-truth) triggering matrix.

	Nonparametric	Temporal	Parametric	Bayesian
R_1	0.61 (0.12)	0.36 (0.12)	0.55 (0.10)	0.40 (0.05)
Correlation	0.81 (0.16)	0.48 (0.27)	0.76 (0.15)	0.23 (0.14)
Ratio	0.63 (0.04)	0.43 (0.06)	0.62 (0.03)	0.33 (0.03)
Coherence	0.78 (0.04)	0.62 (0.03)	0.72 (0.03)	0.70 (0.03)
Entropy	0.75 (0.05)	0.58 (0.03)	0.69 (0.03)	0.62 (0.04)

Edge Reconstruction

We also evaluate the reconstruction methods based on their ability to recover the existence of edges. This is particularly relevant if we want to know whether there is a connection between two entities. We will discuss this application in detail using the Gowalla data set (see section 3.3.2).

We consider an edge to exist if the corresponding weighted entry in an inferred triggering matrix exceeds a certain threshold. For different threshold levels, we compute the numbers of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) for a given ground-truth triggering matrix. We summarize our results in a receiver operating characteristic (ROC) plot (see fig. 3.1), in which we plot the true-positive rate (TPR) (where $\mathrm{TPR} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$) versus the false-positive rate (FPR) (where $\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP}+\mathrm{TN})$). A better inference of a triggering matrix gives a larger value of TPR for a fixed FPR.

Based on the ROC plot in fig. 3.1, we conclude that the spatiotemporal models — both the parametric and nonparametric Hawkes models that we proposed in section 3.1 — outperform the exclusively temporal ones. Therefore, incorporating spatial information improves the quality of our reconstructed unweighted (i.e. binary) networks, at least according to this measure of success. Unsurprisingly, the best results are from our parametric (ground-truth) model. The performance of our nonparametric model is very close to that of the parametric model, confirming its effectiveness at inferring the existence of edges.

Inferred Kernels

We report inferred kernels for our synthetic networks of the nonparametric spatiotemporal Hawkes model, parametric spatiotemporal Hawkes model, and temporal Hawkes model in fig. 3.2. Recall that the ground-truth kernels that we use to simulate point processes are $g_1(t) = \omega \exp(-\omega t)$ and $h(r) = 2\pi r g_2(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right)$, where $r^2 = x^2 + y^2$, $\omega = 0.6$, and $\sigma^2 = 0.3$. Let $\hat{g}_1$ and $\hat{h}$ denote the inferred temporal and spatial kernels, respectively.

We calculate the L_1 errors $\int |g_1(t) - \hat{g}_1(t)|\, \mathrm{d}t$ and $\int |h(r) - \hat{h}(r)|\, \mathrm{d}r$. We report these errors in table 3.3 and present visualizations of the inferred kernels in fig. 3.2. As expected, the

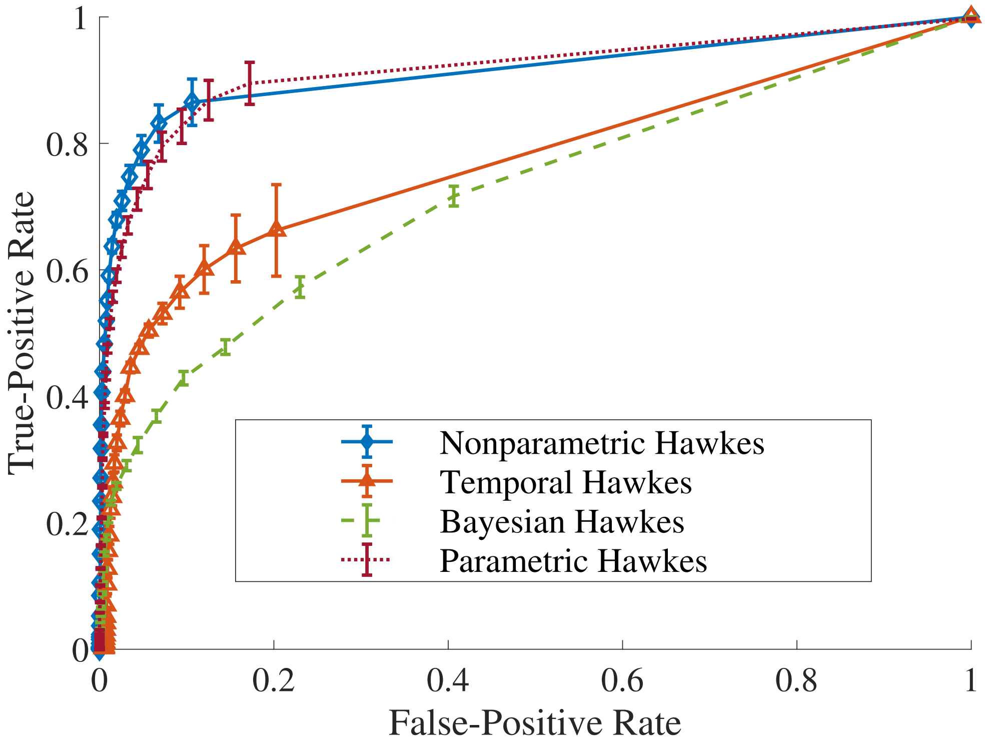


Figure 3.1: Model comparison using synthetic networks. We show the mean ROC curves with error bars (averaged over ten simulations, each with a different triggering matrix) on edge reconstruction. The ROC curve of a better reconstruction should be closer to 1 for a larger range of horizontal-axis values, such that it has a larger area under the curve (AUC), which is equal to the probability that a uniformly-randomly chosen existing edge in a ground-truth network has a larger weight than a uniformly-randomly chosen missing edge in the inferred network.

Table 3.3: The L_1 errors of the inferred spatial and temporal kernels. We simulate ten point processes with the same triggering matrix and triggering kernel. We report the mean and standard deviation (in parentheses) of the L_1 errors averaged over ten simulations that use the same triggering kernel and matrix. Note that the exclusively temporal model does not estimate a spatial kernel.

	Nonparametric	Temporal	Parametric
Temporal kernel	0.07 (0.02)	0.20 (0.06)	0.02 (0.02)
Spatial kernel	0.06 (0.02)	-	0.12 (0.02)

two spatiotemporal Hawkes models give more accurate kernel inference than the exclusively temporal model. The nonparametric Hawkes model does not use any information about the ground-truth kernels. Surprisingly, it is more accurate (in terms of the L_1 error) at inferring the spatial trigger kernel than the parametric model, whose kernel has the same parametric form as the ground-truth kernel.

Community-Structure Recovery

We also evaluate the quality of the inferred networks based on their community structure, in which dense sets of nodes in a network are connected sparsely to other dense sets of nodes [POM09, FH16]. Recall that we planted a four-community structure in the synthetic triggering matrices (see section 3.3.1). We apply the community-detection methods from [AJC14] (an inference method for a WSBM), [KDP12] (symmetric non-negative matrix factorization; NMF), and [JBJ19, NG04, New06, MRM10] (modularity maximization[2]). The WSBM that we infer for community detection is the same one that we use to construct the synthetic adjacency matrices (see section 3.3.1). To evaluate our inferred community structure, we use the square-root variant of *normalized mutual information* (NMI) [SG02] between the inferred community assignment and "ground-truth" community labels. Specifically, let S_1 and

[2] For modularity maximization, we use the implementation of a (locally greedy) Louvain-like method (called GenLouvain) from [JBJ19] with the default resolution-parameter value of 1 and the Newman–Girvan null model.

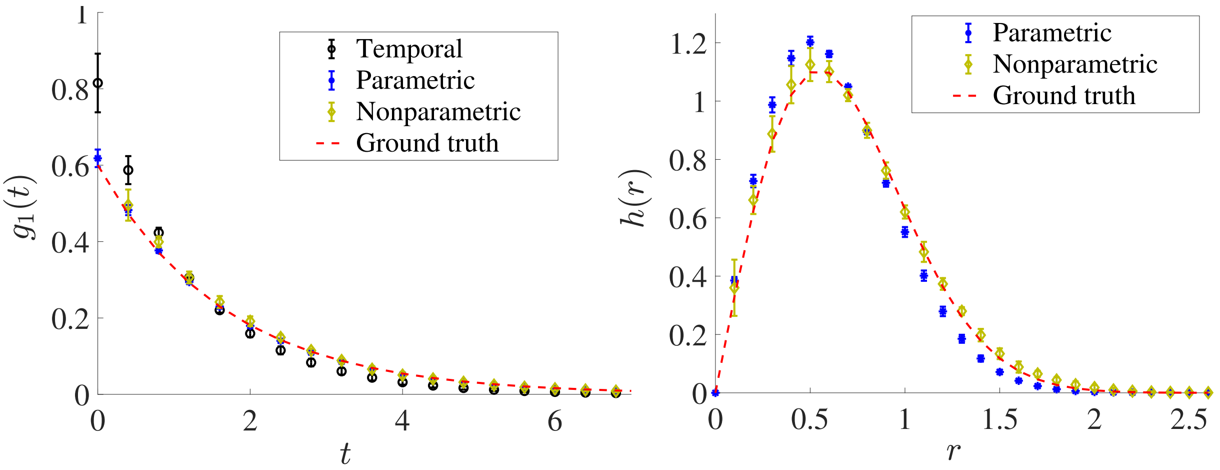


Figure 3.2: Model comparison using synthetic networks: Inferred (left) temporal and (right) spatial kernels using three different methods: Temporal Hawkes, Parametric Hawkes, and Nonparametric Hawkes. The dashed curves are (ground-truth) kernels that we used to generate the synthetic data.

S_2 be community assignments of the U nodes to C_1 and C_2 communities, respectively; and let $S_{\ell k}$, with $\ell \in \{1, 2\}$ and $k \in \{1, 2, \cdots, C_\ell\}$, denote the set of nodes in the k^{th} community in assignment S_ℓ. The NMI between S_1 and S_2 is

$$\mathrm{NMI}(S_1, S_2) = \frac{I(S_1, S_2)}{\sqrt{H(S_1)H(S_2)}} \in [0, 1]\,,$$

where $I(S_1, S_2) = \sum_{i=1}^{C_1} \sum_{j=1}^{C_2} \frac{|S_{1i} \cap S_{2j}|}{U} \log\left(\frac{|S_{1i} \cap S_{2j}|/U}{|S_{1i}||S_{2j}|/U^2}\right)$ (where $|J|$ denotes the cardinality of the set J) and the entropy is $H(S_\ell) = -\sum_{i=1}^{N_\ell} \frac{|S_{\ell i}|}{N} \log\left(\frac{|S_{\ell i}|}{N}\right)$ (with $\ell \in \{1, 2\}$). Intuitively, NMI measures the amount of information that is shared by two community assignments. If they are the same after permuting community labels, the NMI is equal to 1. A larger NMI score implies that the inferred community assignment shares more information with the ground-truth labels. See [TKM11] for a discussion of other approaches for comparing different community assignments in networks.

There are numerous approaches for detecting communities in networks [FH16, POM09, Pei19], and we use methods with readily-available code. As we show in table 3.4, all of these community-detection methods perform better when we infer triggering matrices using both spatial and temporal information than with exclusively temporal information. One can, of

Table 3.4: Normalized mutual information (NMI) between the outputs of different community-detection methods applied to the inferred networks (from four different types of Hawkes models) and the ground-truth community structure (averaged over ten simulations, each with a different triggering matrix).

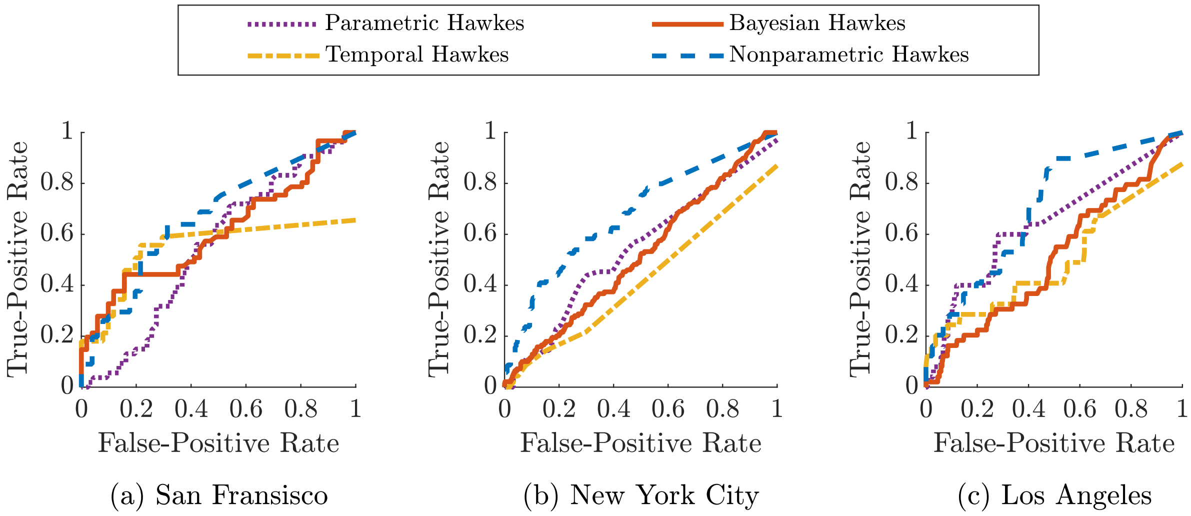
	Nonparametric	Temporal	Parametric	Bayesian
Weighted SBM	0.80	0.38	0.83	0.36
Symmetric NMF	0.62	0.31	0.66	0.19
Modularity Maximization	0.64	0.47	0.71	0.28

course, repeat our experiments using other methods.

3.3.2 Gowalla Friendship Network

Gowalla is a location-based social-media platform in which users share their locations by checking in. We use a Gowalla data set — collected in [CML11] using Gowalla's public API — of a "friendship" network with 196,591 users, 950,327 edges, and a total of 6,442,890 check-ins of these users between February 2009 and October 2010. The data set also includes the latitude and longitude coordinates and the time (with a precision of one second) of each check-in. Similar to a Facebook "friendship" network, the Gowalla friendship network is undirected. The mean number of friends per user is 9.7, the median is 3, and the maximum is $14,730$. We study several subnetworks in the Gowalla data set; see the end of this section for details. We view the spatiotemporal check-ins of Gowalla users within each subnetwork as events in a multivariate point process and infer relationships between these users.

We compare our Nonparametric Hawkes method with the Bayesian Hawkes and the exclusively Temporal Hawkes in terms of how well our inferred edges match the Gowalla friendships. Because a Gowalla "friendship" relationship is undirected in nature, we first symmetrize the inferred triggering matrix (by calculating $\tilde{\mathbf{K}} = \left(\mathbf{K} + \mathbf{K}^T\right)/2$) to obtain an undirected network. We then calculate FPRs and TPRs in the same fashion as in section 3.3.1 using $\tilde{\mathbf{K}}$'s associated "ground-truth" friendship network and generate the corresponding ROC curves. In the ROC curves of three different cities in fig. 3.3, we obtain the best



(a) San Fransisco (b) New York City (c) Los Angeles

Figure 3.3: ROC curves of four different Hawkes models for reconstructing three Gowalla friendship networks. We show results for Nonparametric Hawkes (blue dashed curves), Temporal Hawkes (yellow dash-dotted curves), Bayesian Hawkes (red solid curves), and Parametric Hawkes (dotted purple curves).

results when using our nonparametric model that incorporates spatial information. In the examined subnetworks, the mean AUCs are 0.4277 (with a standard deviation of 0.1042) for the Temporal Hawkes method; 0.5301 (with a standard deviation of 0.0585) for the Bayesian Hawkes method; 0.5816 (with a standard deviation of 0.0525) for the Parametric Hawkes method; and 0.6692 (with a standard deviation of 0.0421) for our Nonparametric Hawkes method.

Now we detail how we preprocess the Gowalla data that were collected and studied in [CML11]. We examine data from three cities: New York City, Los Angeles, and San Francisco. In fig. 3.4, we visualize the networks that we use in this paper.

New York City (NYC)

We study check-ins in New York City (NYC) during the period of April–October 2010. We use a bounding box (with a north latitude of 40.92, a south latitude of 40.48, an east longitude of -73.70, and a west longitude of -74.26)[3] to locate check-ins in NYC. We consider "active" users, who have at least 100 check-ins during the period. To alleviate our computational burden, we also only consider users who have at most 500 check-ins during the period to reduce the number of users and the total number of check-ins. Our inference process requires computing a triggering probability for each pair of events (i.e. check-ins); this results in a full upper-triangular matrix. The number of nonzero entries in this matrix scales with the square of the total number of events, so the memory requirement also scales quadratically with the number of events. We perform experiments only for cases in which the total number of events is at most $10,000$ to be able to store triggering probabilities for all pairs of events in 4-gigabyte memory. There are $5,801$ unique users with at least one check-in in NYC during the period, and there are $101,329 check-ins$ in total. After removing "inactive" users (i.e. those with strictly fewer than 100 check-ins) and overly active users (i.e. those with strictly more than 500 check-ins), we are left with 160 users and a total of $29,118$ check-ins. We also restrict ourselves to users in the largest connected component (LCC) of the network. This yields 46 users and $8,495$ check-ins, on which we apply our inference methodology.

Los Angeles (LA)

We apply the same procedure as in section 3.3.2 on the check-in data for Los Angeles (LA). The bounding box that we use for LA has a north latitude of 34.34, a south latitude of 33.70, an east longitude of -118.16, and a west longitude of -118.67. We restrict the area of LA to be the same as that of NYC, although LA's geographic area is much larger than that of NYC. After selecting only users in the LCC of the Gowalla network among active users (with at least 150 check-ins) but not overly active (with at most 1000 check-ins) users, we are left with 23 users and $6,203$ check-ins.

[3] We obtain latitude and longitude coordinates from `http://www.mapdevelopers.com/geocode_bounding_box.php`.

San Francisco (SF)

To look at a different type of example, we also examine the 1-ego network of the most popular user (who has 14 friends) in San Francisco (SF). (A 1-ego network [UKB11] of a node is an induced subgraph that includes a focal node — the ego — and its direct neighbors.) The bounding box that we use for SF has a north latitude of 37.93, a south latitude of 37.64, an east longitude of -122.28, and a west longitude of -123.17. In this 1-ego network, there are $9,887$ check-ins.

3.3.3 Crime-Topic Network

In a recent paper on crime classification, Kuang et al. [KBB17] performed topic modeling (see [KCP15] for a review) on short narrative (i.e. text) descriptions of all crimes, with spatial coordinates and timestamps (with a precision of one minute), that were reported to and officially recorded by the Los Angeles Police Department (LAPD) between 1 January 2009 and 19 July 2014. The premise in their work was that crime topics, sets of words that co-occur frequently in the same crime narrative, better reflect the ecological circumstances of crime than official crime classifications based on legal codes. Targeting discovery of up to twenty topics, they found six topics related to violent crime, eight topics related to property crime, and six topics that seem to be related to deception-based crime. This classifies the twenty crime topics into three types.

In the present case study, we extend this work by modeling the above data set as a crime-topic network. We associate each crime topic with a node, and we infer edges based on whether crime events of one topic trigger events of other topics. That is, we discover latent relationships between different crime topics based on associated crime events. Inspired by previous research on point-process models of crime events [MSB11], we model crime events of different topics via a multivariate point process and infer connections between the crime topics using our Nonparametric Hawkes method. To evaluate our approach, we compare the communities that we detect in the reconstructed network with the three crime classes in [KBB17].

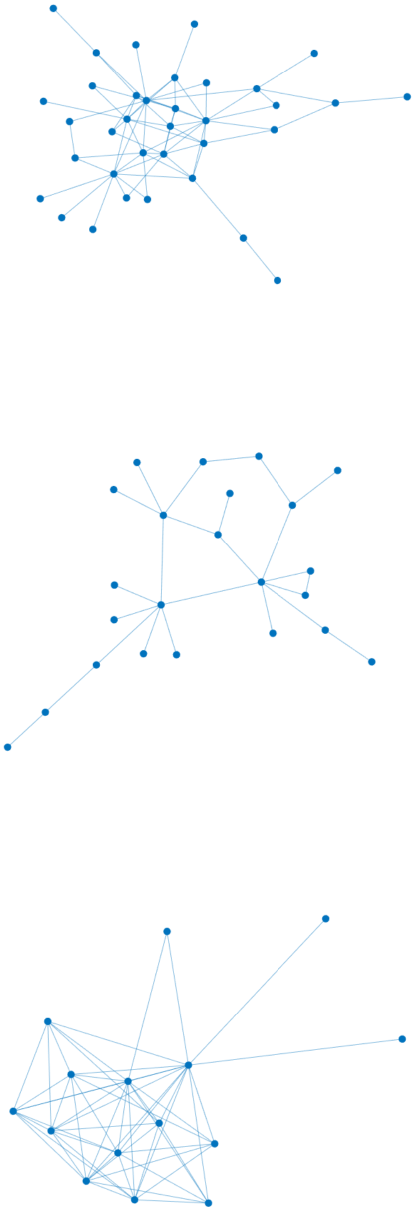


Figure 3.4: Three different friendships networks in the Gowalla data set. We compare different network-reconstruction methods for these networks.

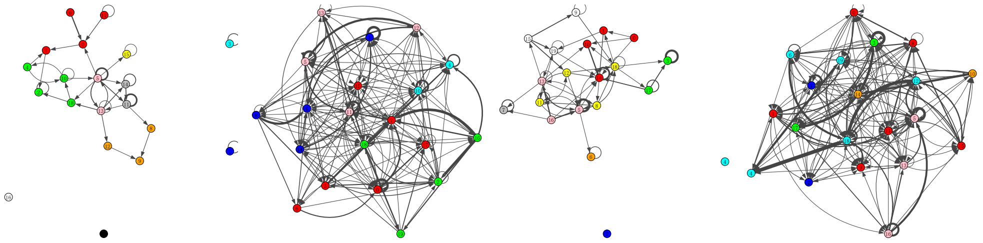


Figure 3.5: Crime-topic networks generated by the Nonparametric Hawkes and Temporal Hawkes methods colored by community assignments from modularity maximization (from left to right): Nonparametric Hawkes in Westwood, Temporal Hawkes in Westwood, Nonparametric Hawkes in Wingfoot, and Temporal Hawkes in Wingfoot.

Community Detection

We infer crime-topic networks directly from crime events within individual Los Angeles neighborhoods[4] using our Nonparametric Hawkes method, the Parametric Hawkes method, and the exclusively temporal Hawkes method. We investigate the 100 neighborhoods with the most reported crime events among all 296 neighborhoods of Los Angeles (LA). On average, there are $4,140$ crime events in the top-100 neighborhoods and $8,750$ such events in the top-10 neighborhoods. We then apply the community-detection methods that we mentioned in section 3.3.1 to the reconstructed networks; this assigns crime topics to communities. We quantify the difference between these community assignments and the crime-topic classifications from [KBB17] by calculating NMI. We also visualize the crime-topic networks of the Westwood and Wingfoot neighborhoods in fig. 3.5; they are located in West LA and South LA, respectively. From table 3.5, we see that using spatial information combined with a nonparametric kernel leads to the best mean NMI score among the methods that we examine.

[4] We use the Zillow neighborhood boundaries from `https://www.zillow.com`.

Table 3.5: Mean NMI (with one standard deviation reported in parentheses) between community assignments from several community-detection methods and the classifications from [KBB17] in the 100 neighborhoods in Los Angeles with the most recorded crime events between 1 January 2009 and 19 July 2014.

	Nonparametric	Temporal	Parametric	Bayesian
Symmetric NMF	0.25 (0.11)	0.12 (0.084)	0.084 (0.12)	0.12 (0.080)
Weighted SBM	0.24 (0.12)	0.085 (0.086)	0.078 (0.079)	0.076 (0.075)

3.3.4 Network of Crime Events

In the previous sections, we studied relationships between entities based on spatiotemporal events associated with them. To examine connections between events, we now define an event network, which is both weighted and directed, in which each event is a node and $\mathbf{A}$ denotes the adjacency matrix of this network. That is, $\mathbf{A}(i,j)$ is the probability that event j is triggered by event i, and $\mathbf{A}(i,i)$ is the probability that event i is a background event. From this definition, we see that $\mathbf{A}$ is equal to the expectation matrix $\mathbf{P}$ from section 3.2.2. The weight of an edge reflects a triggering effect between two events, and the direction points from the earlier event to the later one. (Removing weights yields a directed acyclic graph.) For example, we can build a crime-event network in which each node is a crime incident (i.e. an event), and we estimate edges between events using our nonparametric model.

Stochastic Declustering

With an event network, a natural question is whether one can differentiate between "true" background events and triggered events. Such differentiation using the probability p_{ii} is called *stochastic declustering* [ZOV02]. To determine whether event i is a background event, we compare p_{ii} with a uniformly random sample from the interval $(0,1)$. If p_{ii} is larger than the random number, we consider this event to be from the background; otherwise, we consider it to be triggered by other events.

We perform declustering experiments on synthetic data; we simulate ten synthetic point processes using a fixed triggering matrix that we generate from a WSBM. (See section 3.3.1 for details.) Recall from algorithm 3 that we retain causality information in the simulations (i.e. which events cause which others and which events are from the background), giving a notion of "ground truth" about the ancestors of each event. One way to measure the quality of declustering is by comparing the inferred branching ratio [SU09] with the one from the ground-truth data. The branching ratio is defined as $1 - N_b/N$, where N_b is the number of background events. However, the difference in branching ratios itself typically does not completely reflect reconstruction errors. For example, in an extreme case, stochastic declustering can erroneously misclassify some number of background events as triggered ones and the same number of triggered events erroneously as background ones, although the branching ratio is the same as the true branching ratio in this scenario. To resolve this problem, we view declustering as a binary classification problem that assigns events to be either background or triggered events. We use measurements such as recall and precision to evaluate our declustering results. Recall that "recall" is the ratio between the number of background events that are correctly recovered by the declustering methods (i.e. the true positives) to the total number of background events; and "precision" is the ratio between the number of true positives to the number of events that are labeled as background events by stochastic declustering. From the results in table 3.6, we see that the Temporal Hawkes method performs worse than the Nonparametric Hawkes and Parametric Hawkes methods. Our Nonparametric Hawkes method has the best recall and precision (with the smallest variations as well), and the Parametric Hawkes method has the smallest branching-ratio error.

Motif Analysis

Declustering methods can help differentiate between the background and triggered events in an event network. To further examine spatiotemporal dynamics, we consider causality information between events. Similar to a relational-event model [But08], one can obtain causality information from the matrix $\mathbf{P}$, because p_{ij} is the probability that event j is triggered by event i. We focus on repeated patterns to obtain information about the local causal

Table 3.6: Comparison of our stochastic declustering results for the Nonparametric Hawkes, Parametric Hawkes, and Temporal Hawkes methods using synthetic point-process data with networks from a WSBM (see section 3.3.1) and background labels from the simulation from algorithm 3. (We do not include results for Bayesian Hawkes, because it does not provide $\mathbf{P}$ directly.) We report the mean and the standard deviation (in parentheses) of the branching-ratio error, precision, and recall over ten simulations (which we do for ten point processes with the same triggering kernels and matrix). For each simulation, each calculation is the mean over 20 runs of stochastic declustering.

	Nonparametric	Parametric	Temporal
Branching-ratio error	0.039 (0.0050)	0.01 (0.011)	0.022 (0.019)
Recall	0.75 (0.0098)	0.65 (0.027)	0.60 (0.035)
Precision	0.70 (0.0082)	0.64 (0.0093)	0.59 (0.0086)

structure. Specifically, we examine network motifs [MSI02], which are recurrent (and often statistically significant) patterns in a network.

We find that motif analysis is insightful for studying gang-crime event networks. Gang crimes are often characterized by retaliations (triggered crime events) between rivalry gangs; this can lead to a series of tit-for-tat reciprocal crimes. To find significant gang retaliation patterns, we use a gang-crime data set (provided by the LAPD) from 2014–2015 with 4,158 events in Los Angeles. Using these data, we generate an event network with our Nonparametric Hawkes method. We then threshold the network, by keeping edges whose weight is at least 0.1 and then binarizing them, so that the edges are unweighted. We use the motif-detection method and code from [MSI02], including their null model.[5]

We find, for thresholds ranging from 0.5 to 0.001, that a three-node feedforward-loop motif [MA03] occurs more significantly than by chance (with z-scores that are larger than

[5] For each of our networks, we produce 100 "randomized" networks. To produce one such network, we use the default edge-swapping approach from [MSI02]. This entails making several random swaps equal to about 100–200 times the number of edges. For each node in a network, we require that the randomized network preserves its numbers of in-edges, out-edges, and bidirectional edges.

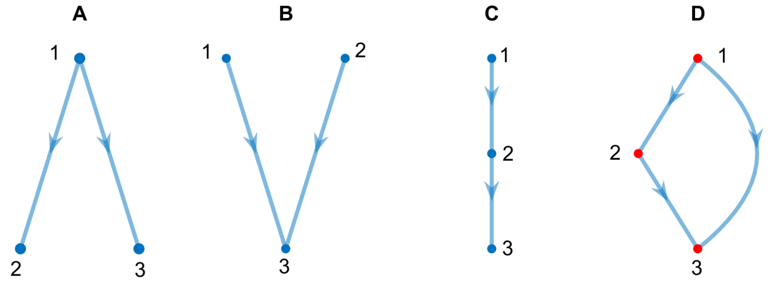


Figure 3.6: All possible three-node motifs for a network of events in the form of a directed acyclic graph (DAG). The DAG structure arises from the temporal information in the events. We highlight the nodes in the feedforward-loop motif (D) in red.

2) in both the city-wide data set and in the South LA[6] a subset (which consists of 1,912 events) of the data set. Davies and Marchione [DM15] found that the same three-node motif is significant in networks that they constructed (using different methods from ours both for network construction and motif detection) using data sets from maritime piracy and residential burglaries.

We focus on the South LA area because it is the center of a gang intervention program [BSY17]. Establishing which causal structures are statistically significant has important implications for countering gang violence, and a fast response to a gang crime may reduce the potential that it triggers a future retaliation. Knowing that feedforward-loop network motifs occur at rates that are larger than chance suggests that disrupting retaliation may require an assessment of trade-offs in how to allocate intervention resources. For example, in a simple triggering chain (see fig. 3.6 C), one can expect that intervention following an initial triggering event will have a direct effect on the second event and an indirect effect on the third event, although the effect on the third event may be attenuated by the intervening event. By contrast, we expect that intervention following the first event in the feedforward-loop motif (see fig. 3.6 D) will have a direct effect on the second event and both a direct and indirect effect on the third event. The third event may be more likely to be disrupted given

[6] We use the term "South LA" to designate a specific area of Los Angeles that is defined in a recent Gang Reduction and Youth Development (GRYD) report [BSY17].

the feedforward structure and intervention following the first event than would be the case with direct intervention following only the second event.

3.4 Conclusions and Discussion

In this chapter, we studied the role of spatial information and nonparametric techniques in network reconstruction. We used point-process models to infer latent networks from synthetic and real-world spatiotemporal data sets. We then applied tools from network analysis to examine the inferred networks.

As we have illustrated, it is very important to incorporate spatial information in network reconstruction. However, using such information effectively requires making a good choice of spatiotemporal triggering kernels. We achieved this using a nonparametric approach. Through experiments on synthetic data sets, we showed that our nonparametric Hawkes method is capable of doing a good job of successfully recovering spatial and temporal triggering kernels. Moreover, our approach can infer a network structure that better recovers — compared to other network-reconstruction methods that we studied — symmetry and reciprocity, edge reconstruction, and community structure. Through experiments on real-world data sets, we illustrated that our approach yields meaningful inferred networks, in the sense that they have large positive correlations with some metadata.

Naturally, our network-reconstruction method is not without limitations. It uses $O(U^2)$ parameters for U nodes. To avoid underfitting, it requires a large number of observed events. The computational complexity and memory requirement scale at least quadratically with the number of events, so the current EM-type algorithm is not ideal for analyzing large data sets. Therefore, in the next chapter, we develop a novel approach to improve our inference method for network reconstruction, especially for large data sets.

CHAPTER 4

Fast Estimation of Multivariate Hawkes Processes

Much of recent research on spatiotemporal (ST) point processes has been fueled by advances in the nonparametric estimation of Hawkes processes, and in particular by the landmark work of Marsan and Lengliné [ML08], who detailed a method for estimating the triggering kernel in a ST-Hawkes process by assuming the triggering density to be a step function and then estimating the step heights via maximum likelihood estimation (MLE). Such nonparametric estimation methods allow the triggering density to be estimated without assuming a particular parametric form which may be subject to misspecification or over-fitting, which can be very serious problems, especially in social science applications [YLB19]. Instead, the data drive the estimation of the triggering density, and this is especially attractive to use with the large data sets that are increasingly becoming available in applications. Unfortunately, however, a major limitation of current nonparametric estimation methods is their computational complexity and lack of speed, as existing methods are mainly based on maximum likelihood estimation (MLE) [Rei18b], or variants such as EM-type algorithms [VS08, ML08], which are typically non-convex problems without closed-form solutions. For applications to crimes or to social media, for instance, catalogs of millions of ST events are often the subject of study, and each calculation of the likelihood function with N events requires at least $O(N^2)$ time. In such situations, the estimation of the triggering density using existing methods can be infeasible. As a result, it is important to develop better alternatives to current MLE-based methods [Sch18].

Recent developments in the nonparametric estimation of the Hawkes process provide new insights for this problem, including an analytic method for computing the MLE of the triggering density in the special case where the adjacency matrix is invertible [SGH18], and generalized moments methods (GMM) for the estimation of the triggering matrix [ABG17]. However, several limitations prevent us from applying them directly to multivariate ST-Hawkes processes. The analytic MLE method in [SGH18] can only be applied to the univariate case while the GMM method for the temporal process cannot estimate the triggering kernels. In this chapter, we propose a new, highly computationally efficient, scalable nonparametric estimator for ST-Hawkes processes, based on a blend of these recent ideas with modern advances in the regularization and inversion of sparse matrices. There are two major ingredients. The first is the analytic derivation of the likelihood-based estimation, which directly computes the exact maximum likelihood estimation of the nonparametric triggering density. We develop it for the multivariate case and add regularization to improve stability and robustness. The second is the moment-based method for the background rate and triggering matrix estimation, which is extended here for the spatiotemporal case.

The contributions of this chapter are three-fold. First of all, we extend the analytic formula for the MLE of the step heights in the triggering density [SGH18] to the multivariate ST case and greatly improve the stability of the resulting estimator using regularization. We next extend the cumulant-based estimators of [ABG17] to the multivariate ST case and derive GMM estimators of the triggering matrix in this context. Finally, we combine the MLE and GMM estimators to obtain a scalable, consistent and efficient estimator and show that the proposed estimator has a linear computation complexity in the number of events N, allowing one to explore applications to large data sets with millions of events, in which our method outperforms current state-of-the-art methods in terms of both accuracy in network reconstruction and computation time.

The structure of this chapter is as follows. We first give a detailed review of inference methods for Hawkes processes in section 4.1. In section 4.2, we develop the proposed method and show consistency and computational complexity. The performance of this estimator is inspected using a variety of synthetic and real social-network data sets in section 4.4. Finally, we conclude and discuss important directions for future research in section 4.5.

4.1 Multivariate Hawkes Processes and Nonparametric Estimations

In this section, we review the definition of multivariate Hawkes processes and previous papers on inference methods, focusing especially on MLE and GMM. Recall that for a multivariate temporal Hawkes process, each subprocess N_u has conditional intensity

$$\lambda_u(t) = \mu_u + \sum_{t_k<t} K_{u_k u}\, g_{u_k}(t - t_k)\,, \tag{4.1}$$

and the N points of the entire process may conveniently be labeled (t_k, u_k), for $k = 1, ..., N$, where t_k indicates the time of point k, and u_k indicates the index dictating to which subprocess the point belongs.

In the nonparametric estimation of g, one typically assumes that each subprocess has the same piecewise-constant triggering density $g_{u_k}(t) = g(t)$ which controls how quickly the rate $\lambda_u(t)$ returns to its baseline level μ_u after an event occurs. One can estimate the parameters $\boldsymbol{\mu} = (\mu_u)_u$, $\boldsymbol{K}$, and the triggering densities g via MLE [Oga78] or minimize a regression loss [CSS17]. Here we focus on the MLE approach. The log-likelihood function of the intensity function eq. (4.1) becomes

$$l = \sum_{k=1}^{N} \log(\lambda_{u_k}(t_k)) - \sum_{u=1}^{U} \int_0^T \lambda_u dt\,. \tag{4.2}$$

One can directly maximize this function using off-the-shelf optimization methods or the EM-type algorithm proposed in [VS08]. See [YLB19] for details about the derivation of the EM-type algorithm for ST-Hawkes processes. Another MLE-based approach, based on their analytic derivation of MLE, is first proposed in [SGH18] for the univariate case ($U = 1$). They found that one can solve the MLE problem via solving linear equations in g and two additional linear equations for the background rate μ and productivity K. However, for the multivariate case, the coefficients of these equations depend on the triggering matrix $\boldsymbol{K}$ and it is no longer a linear system. Also, there is the problem of stability when the matrix of the linear system is singular or nearly singular. The inversion of the matrix is a major problem [SGH18] in its implementation in practice, and in section 4.3 we present the solution to this problem via regularization.

Another kind of estimation method [ABG17, BM16] is based on GMM using cumulants of Hawkes processes. Define $\mathbf{R} = (\mathbf{I} - \mathbf{K}^T)^{-1}$, where $\mathbf{I}$ is the identity matrix. As an alternative to the moments, the first, second and third cumulant of Hawkes process $\mathbf{\Lambda}$, $\boldsymbol{C}$ and $\mathbf{\Gamma}$ have the following relationships [ABG17] with $\boldsymbol{R}$

$$\mathbf{\Lambda}(i) = \Lambda^i = \sum_{m=1}^{U} R^{im} \mu_m \,, \tag{4.3}$$

$$\boldsymbol{C}(i,j) = C^{ij} = \sum_{m=1}^{d} \Lambda^m R^{im} R^{jm} \,, \tag{4.4}$$

$$\mathbf{\Gamma}(i,j,k) = \Gamma^{ijk} = \sum_{m=1}^{d} (R^{im} R^{jm} C^{km} + R^{im} C^{jm} R^{km} + C^{im} R^{jm} R^{km} - 2\Lambda^m R^{im} R^{jm} R^{km}) \,. \tag{4.5}$$

Here $R^{im} = \boldsymbol{R}(i,m)$. Although the definition and numerical estimations of the cumulants are different for the ST case, the above formulas still hold because the spatial information can be viewed as "marks" of the temporal point process.

The idea of GMM is to estimate the cumulants numerically from the data and then obtain the triggering matrix $\boldsymbol{K}^T = \mathbf{I} - \mathbf{R}^{-1}$ by minimizing the approximation error of the cumulants with some scaling coefficient κ (see more details in section 4.2.2)

$$L(\boldsymbol{R}) = (1-\kappa) \| \boldsymbol{R}^{\odot 2} \hat{\boldsymbol{C}}^T + 2(\boldsymbol{R} \odot (\hat{\boldsymbol{C}} - \boldsymbol{R}\hat{\boldsymbol{L}}))\boldsymbol{R}^T - \hat{\mathbf{\Gamma}}^{\boldsymbol{c}} \|_2^2 + \kappa \| \boldsymbol{R}\hat{\boldsymbol{L}}\boldsymbol{R}^T - \hat{\boldsymbol{C}} \|_2^2 \,. \tag{4.6}$$

Here $\odot$ is the Hadamard product and $\hat{\mathbf{\Gamma}}^{\boldsymbol{c}} = \hat{\mathbf{\Gamma}}(i,i,k)$. Given the estimated $\tilde{\boldsymbol{R}}$, we also have $\tilde{\boldsymbol{\mu}} = \tilde{\boldsymbol{R}}^{-1} \tilde{\mathbf{\Lambda}}$ from the cumulants eq. (4.3). This provides a fast estimation procedure for both $\boldsymbol{\mu}$ and $\boldsymbol{K}$. But it does not estimate the triggering density, which plays an important role in the dynamics of the point process. In applications such as stochastic declustering [ZOV02], it is necessary to estimate triggering densities from the data. Some other moment-based methods [BM16] can estimate both of them at the cost of high computation time.

4.2 Proposed Methods for Multivariate ST-Hawkes

In this section, we extend the previous discussion to the case of multivariate ST-Hawkes processes and derive a fast estimation method via extending and combining the two approaches (MLE and GMM) discussed above. We recommend interested readers to check [Rei18b, SBG14] which provide comprehensive reviews of ST point processes. The focus of our method is to reduce the computational burden of the inference as well as improve the model estimation accuracy. Our motivation is from the application of network reconstruction. Previous studies have shown the ability of Hawkes process models to uncover the underlying connections between nodes (such as social media users [YLB19], neurons [CSS17], email users [FSS16] and crime [LA14]). It is essential to develop a scalable method because one often encounters data sets with thousands of nodes (large U) and millions of associated ST events (very large N).

We consider a simple multivariate ST-Hawkes process with a spatially isotropic triggering density $g(x, y, t)$ – i.e. $g(x, y, t) = g(r, t), r = \sqrt{x^2 + y^2}$ (g is only a function of time and distance). We assume that $g(t)$ is the same for all subprocesses for simplicity. It can be easily extended to the general case via adding more variables and equations on g like eq. (4.14). For each subprocess $u = 1, ..., U$, the conditional intensity characterizing the multivariate ST-Hawkes process is assumed to have the form

$$\lambda_u(x, y) = \mu_u(x, y) + \sum_{t_k<t} K_{u_k u} g(d_k, t - t_k)\,, \tag{4.7}$$

where (t_k, x_k, y_k, u_k), for $k = 1, ..., N$, denotes the N observed events in $R \times [0, T]$ and $d_k = \sqrt{(x_k - x)^2 + (y_k - y)^2}$. Current MLE-based methods such as the EM-type algorithm [VS08, YLB19] are not well-suited for large-scale problems due to its $O(N^3)$ computational complexity [ABG17]. Also, in many applications, it is difficult to determine the appropriate triggering density $g(r, t)$. Our proposed method has a linear $O(N)$ complexity and learns triggering densities directly from data. Specifically, we estimate $g(r, t)$ nonparametrically from MLE and $\boldsymbol{K}$, $\boldsymbol{\mu}$ from GMM. This combined method gives a fast and complete estimation of the ST-Hawkes process.

4.2.1 ST Triggering Density Estimation

We extend the analytic method, first proposed in [SGH18] for the univariate temporal case, to the case of multivariate ST-Hawkes processes.

First, we review the derivation of analytic estimates of the triggering function for the multivariate temporal Hawkes process eq. (4.1). WE Assume that $\boldsymbol{\mu}$ and $\mathbf{K}$ are given or well-estimated by other means, and the only variables here to be estimated are the heights of the step function comprising the triggering density $g(t) = \sum_{m=1}^{N_t} g_m \mathbb{1}_{t\in(\tau_m,\tau_{m+1})}$ with N_t grids $U_m = \{t \mid t \in (\tau_m, \tau_{m+1})\}, m = 1, ..., N_t$ dividing the time window $[0, T]$. One seeks to obtain step heights of the triggering density via maximizing the log-likelihood function. The log-likelihood function (from eq. (4.2))

$$l = \sum_{k=1}^{N} \log(\lambda_{u_k}(t_k)) - \sum_{u=1}^{U}(\mu_u T + \sum_{m=1}^{N_t} g_m \delta_m \sum_{k=1}^{N} K_{u_k u}) \tag{4.8}$$

is concave with respect to $\{g_m\}_m$. We take the derivative with respect to g_m and set it to zero:

$$0 = \frac{\partial l}{\partial g_m} = \sum_{(t_j - t_i)\in U_m} \frac{K_{u_i u_j}}{\lambda_{u_j}(t_j)} - \sum_{u=1}^{U}\sum_{i=1}^{N} K_{u_i u}\delta_m \,, \tag{4.9}$$

where $\delta_m = \tau_{m+1} - \tau_m$. Using the notation $\boldsymbol{\lambda} = \{\lambda_{u_j}(t_j)\}_j$, $\boldsymbol{A}(k, j) = \sum_{t_j - t_i \in U_k} K_{u_i u_j}$, $\boldsymbol{\beta} = \{g_m\}_m$ and $\boldsymbol{b} = \{\sum_{u=1}^{U}\sum_{i=1}^{N} K_{u_i u}\delta_m\}_m$, we obtain a matrix representation of eq. (4.9) as

$$0 = \boldsymbol{A}(1/\boldsymbol{\lambda}) - b\,. \tag{4.10}$$

Here $1/\boldsymbol{\lambda}$ is the element-wise reciprocal. The solution of eq. (4.10) yields an estimate of $\boldsymbol{\lambda}$. Further, eq. (4.1) can be rewritten as

$$\boldsymbol{\lambda} = \boldsymbol{\mu} + \boldsymbol{A}^T\boldsymbol{\beta}\,. \tag{4.11}$$

Solving this equation using the estimation of $\boldsymbol{\lambda}$ from eq. (4.10) provides the maximum likelihood estimation of $\boldsymbol{\beta}$.

We now focus on the multivariate ST-Hawkes process with a piecewise-constant ST triggering density $g(r, t)$. We simply assume a uniform background rate $\mu_u(x, y) = \mu_u$. For each subprocess $u = 1, ..., U$, the conditional intensity satisfies

$$\lambda_u(x, y, t) = \mu_u + \sum_{t_k < t} K_{u_k u} \sum_{m=1}^{N_t} \sum_{n=1}^{N_r} g_{mn} \mathbb{1}_{t_k - t \in (\tau_m, \tau_{m+1})} \mathbb{1}_{d_k \in (r_n, r_{n+1})} \,. \tag{4.12}$$

Here $d_k = \sqrt{(x_k - x)^2 + (y_k - y)^2}$ and g is defined on a 2-D $N_r \times N_t$ grids with $V_n = \{d_k \mid d_k \in (r_n, r_{n+1})\}, n = 1, ..., N_r$ dividing the space R and $U_m = \{t_k - t \mid t_k - t \in (\tau_m, \tau_{m+1})\}, m = 1, ..., N_t$ dividing the time window $[0, T]$. The log-likelihood function of this intensity function is [Sch13]

$$\begin{aligned} l &= \sum_{k=1}^{N} \log(\lambda_{u_k}(x_k, y_k, t_k)) - \sum_{u=1}^{U} \iint_R \int_0^T \lambda_u(x, y, t) dt\, dx\, dy \\ &= \sum_{k=1}^{N} \log(\lambda_{u_k}(x_k, y_k, t_k)) - \sum_{u=1}^{U} (\mu_u |R| T + \sum_m \sum_n g_{mn} \delta_m \bigtriangleup_n \sum_{k=1}^{N} K_{u_k u}) \,, \end{aligned} \tag{4.13}$$

where $|R|$ is the area of R, $\delta_m = \tau_{m+1} - \tau_m$ and $\bigtriangleup_n = \pi(r_{n+1}^2 - r_n^2)$.

Assuming that $\boldsymbol{\mu}$ and $\boldsymbol{K}$ are given, the only variables here are $\{g_{mn}\}_{m,n}$. Maximizing the log likelihood function will give us the estimation of the triggering density g. Since eq. (4.13) is concave, we take the derivative of equation with respect to g_{mn} and set it to zero:

$$0 = \frac{\partial l}{\partial g_{mn}} = \sum_{(t_j - t_i) \in U_m, d_{ij} \in V_n} \frac{K_{u_i u_j}}{\lambda_{u_j}(x_j, y_j, t_j)} - \sum_{u=1}^{U} \sum_{i=1}^{N} K_{u_i u} \delta_m \triangle_n \,, \tag{4.14}$$

with $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. Similar to the temporal case, we define $\boldsymbol{\lambda} = \{\lambda_{u_j}(x_j, y_j, t_j)\}_j$, $\boldsymbol{A}(k(m, n), j) = \sum_{t_j - t_i \in U_m, d_{ij} \in V_n} K_{u_i u_j}$, $\boldsymbol{\beta} = (g_{mn})_{k(m,n)}$ and $\boldsymbol{b} = (\sum_{u=1}^{U} \sum_{i=1}^{N} K_{u_i u} \delta_m \triangle_n)_{k(m,n)}$ with the index $k(m, n) = N_r(m - 1) + n$. Then we obtain the matrix representation of eq. (4.14) and eq. (4.12) as

$$0 = \boldsymbol{A}(1/\boldsymbol{\lambda}) - \boldsymbol{b} \,, \tag{4.15}$$

$$\boldsymbol{\lambda} = \boldsymbol{\mu} + \boldsymbol{A}^{\boldsymbol{T}} \boldsymbol{\beta} \,. \tag{4.16}$$

Finally we can estimate $\boldsymbol{\beta}$ via solving the above linear equations separately.

4.2.2 Triggering Matrix Estimation

In previous sections, we estimate the triggering density with the assumption that both $\boldsymbol{\mu}$ and $\boldsymbol{K}$ are given. In the univariate case, one can remove this assumption by adding two additional linear equations [SGH18]. However, in the multivariate case, because matrix $\boldsymbol{A}$ is depend on the matrix $\boldsymbol{K}$, solving $\boldsymbol{\mu}$, $\boldsymbol{K}$ and g simultaneously is no longer a linear problem.

In order to solve this problem, we extend the cumulants eqs. (4.3), (4.4) and (4.19) to the ST case for a fast estimation of $\boldsymbol{\mu}$ and $\boldsymbol{K}$. For a ST-Hawkes process with U sub-processes, we define its first, second and third cumulant as [DV07]

$$\Lambda^i dtdxdy = \mathbb{E}(dN^i_{t,x,y})\,, \tag{4.17}$$

$$C^{ij} dtdxdy = \int_{\tau,a,b\in\mathbb{R}^3} (\mathbb{E}(dN^i_{t,x,y} dN^j_{t+\tau,x+a,y+b} - \mathbb{E}(dN^i_{t,x,y})\mathbb{E}(dN^j_{t+\tau,x+a,y+b})\,, \tag{4.18}$$

$$\begin{aligned}\Gamma^{ijk} dtdxdy = &\int_{\tau',a',b'\in\mathbb{R}^3}\int_{\tau,a,b\in\mathbb{R}^3} (\mathbb{E}(dN^i_{t,x,y} dN^j_{t+\tau,x+a,y+b} dN^k_{t+\tau',x+a',y+b'})\\ &+ 2\mathbb{E}(dN^i_{t,x,y})\mathbb{E}(dN^j_{t+\tau,x+a,y+b})\mathbb{E}(dN^k_{t+\tau',x+a',y+b'})\\ &- \mathbb{E}(dN^i_{t,x,y} dN^j_{t+\tau,x+a,y+b})\mathbb{E}(dN^k_{t+\tau',x+a',y+b'})\\ &- \mathbb{E}(dN^i_{t,x,y} dN^k_{t+\tau,x+a,y+b})\mathbb{E}(dN^j_{t+\tau',x+a',y+b'})\\ &- \mathbb{E}(dN^j_{t+\tau,x+a,y+b} dN^k_{t+\tau',x+a',y+b'})\mathbb{E}(dN^i_{t,x,y}))\,.\end{aligned} \tag{4.19}$$

Here $1 \le i, j, k \le U$ and τ, a and b are the variables of integration corresponding to t, x and y.

Cumulants can be numerically estimated from the ST events from each subprocess $Z^i = (t_k, x_k, y_k)_k, i = 1, ..., U$ on the ST bounded area $R \times [0, T]$. Here we simply assume that R is a rectangular with length X and width Y. We obtain the following estimation formulas for eqs. (4.17) to (4.19),

$$\hat{\Lambda}^i = \frac{1}{TXY}\sum_{\tau,a,b\in Z^i} = \frac{N^i_{T,X,Y}}{TXY}\,, \tag{4.20}$$

$$\hat{C}^{ij} = \frac{1}{TXY}\sum_{\tau,a,b\in Z^i} (N^j_{a+\tilde{X},b+\tilde{Y},\tau+H} - N^j_{a-\tilde{X},b-\tilde{Y},\tau-H} - 8\tilde{X}\tilde{Y}H\hat{\Lambda}^j)\,, \tag{4.21}$$

$$\begin{aligned}\hat{\Gamma}^{ijk} =& \frac{1}{TXY} \sum_{\tau,a,b\in Z^i} (N^j_{a+\tilde{X},b+\tilde{Y},\tau+H} - N^j_{a-\tilde{X},b-\tilde{Y},\tau-H} - 8\tilde{X}\tilde{Y}H\hat{\Lambda}^j)\times \\ & (N^k_{a+\tilde{X},b+\tilde{Y},\tau+H} - N^k_{a-\tilde{X},b-\tilde{Y},\tau-H} - 8\tilde{X}\tilde{Y}H\hat{\Lambda}^k) \\ & - \frac{\hat{\Lambda}^i}{TXY} \sum_{\tau',a',b'\in Z^k} \sum_{\tau,a,b\in Z^j} (2H - |\tau-\tau'|)^+(2\tilde{X} - |a-a'|)^+(2\tilde{Y} - |b-b'|)^+ \\ & + 64(H\tilde{X}\tilde{Y})^2\hat{\Lambda}^i\hat{\Lambda}^j\hat{\Lambda}^k\,, \end{aligned} \tag{4.22}$$

via numerical integration approximations of the cumulants on $[-\tilde{X},\tilde{X}]\times[-\tilde{Y},\tilde{Y}]\times[-H,H]$ assuming that the support of the triggering density is within this region (see the Appendix B.3 in [ABG17] for more details). One also needs to symmetrize the approximated cumulants via $(\hat{C^{ij}} + \hat{C^{ji}})/2$ and $(2\hat{\Gamma^{iji}} + \hat{\Gamma^{jii}})/3$ because cumulants satisfy $\Gamma^{iji} = \Gamma^{iij}$ and $C^{ij} = C^{ji}$. Finally, we can plug the approximated cumulants into eq. (4.6) to estimate $\boldsymbol{\mu}$ and $\boldsymbol{K}$. The error function in eq. (4.6) is a non-convex polynomial and similar to the loss function of a multilayer neural network. As a result, stochastic gradient descend (SGD) with acceleration (e.g. Adam [KB15] or AdaGrad [DHS11]) can be used to minimize the error function. The normalization term κ is $\kappa = \frac{\|\hat{\boldsymbol{\Gamma^c}}\|_2^2}{\|\hat{\boldsymbol{C}}\|_2^2+\|\hat{\boldsymbol{\Gamma^c}}\|_2^2}$ based on the theory of GMM [ABG17]. The ratio between the support of the triggering density and the ST bounded area $R\times[0,T]$ matters for the consistency of GMM [ABG17]. Usually for specific applications such as social-network reconstruction, $R\times[0,T]$ is much larger than the square of the support of the triggering density, which guarantees the consistency of GMM estimation.

4.2.3 Consistency Guarantee

The consistency of MLE [Oga78] or GMM estimates [ABG17] is guaranteed by general theoretical results. Here we show that our proposed method, as a combination of GMM and MLE, is also consistent.

First, in [Oga78], Ogata showed the MLE of the full vector of parameters is, under quite general conditions, consistent. Also, if only some of the parameters are to be estimated and others, such as in this instance $\boldsymbol{K}$ and $\boldsymbol{\mu}$, are known exactly, then again one may consider the parameter vector to be only those parameters being estimated, and again [Oga78] showed the estimated ones will be consistent. However, we are considering the case where $\boldsymbol{K}$ and $\boldsymbol{\mu}$ are not known but are estimated consistently via GMM, and then the other parameters are estimated by MLE. To the best of our knowledge, this case has not been studied previously, and the result does not immediately follow from the theorems in [Oga78]. We show that $\hat{\boldsymbol{\beta}}$ inherits the property of consistency from the MLE and GMM estimators, under the same assumptions as in [ABG17, Oga78].

Let Θ denote the full vector of parameters, including $\boldsymbol{K}$ and $\boldsymbol{\mu}$. Let Θ_0 denote the true value of Θ. Let $\boldsymbol{U}$ denote a neighborhood of Θ_0. Let $\boldsymbol{K'}$ and $\boldsymbol{\mu'}$ denote the GMM estimates of $\boldsymbol{K}$ and $\boldsymbol{\mu}$. Let $\Theta = (\boldsymbol{K}, \boldsymbol{\mu}, \boldsymbol{\beta})$, where $\boldsymbol{\beta}$ is the vector of other parameters estimated by MLE. Let $\hat{\boldsymbol{K}}$, $\hat{\boldsymbol{\mu}}$, and $\hat{\boldsymbol{\beta}}$ be the MLEs of these parameters.

Theorem 1. *Assuming the same regularity conditions used in the proofs of consistency of the MLE and GMM estimator in [ABG17] and [Oga78], the combined estimator $\hat{\boldsymbol{\beta}} \to \boldsymbol{\beta}$ in probability as $T \to \infty$.*

Proof. Let L denote the log-likelihood divided by T. Thus L depends on T but we will suppress this here. Let Θ_1 denote the supremum over $\boldsymbol{U}^c$ of L, which is the MLE outside of $\boldsymbol{U}$.

We are given that $(\boldsymbol{K'}, \mu') \to (\boldsymbol{K}, \mu)$ in probability as $T \to \infty$. Thus, $(\boldsymbol{K'}, \mu')$ are in $\boldsymbol{U}$ with probability going to 1 as $T \to \infty$.

We have $L(\Theta_1) \to \mathbb{E}(L(\Theta_1))$ and $L(\Theta_0) \to \mathbb{E}(L(\Theta_0))$, where this convergence is in probability as $T \to \infty$ and is uniform in Θ. This follows from the same logic as in the proof of Theorem 2 of [Oga78].

Similarly, following exactly as in the proof on p253 of [Oga78], we have $\mathbb{E}(L(\Theta_0)) > \mathbb{E}(L(\Theta_1))$ and for sufficiently large T, there exists $\epsilon > 0$ such that $|\mathbb{E}(L(\Theta)) - \sup_{\Theta \notin U} \mathbb{E}(L(\Theta))| > \epsilon/2$. This follows from the assumptions in [Oga78], particularly the assumption that λ is uniformly bounded away from 0.

Therefore, since $(\boldsymbol{K'}, \mu')$ are in $\boldsymbol{U}$ as $T \to \infty$, for sufficiently large T, we have $\mathbb{E}(L(\Theta_0))$ and therefore $L(\Theta_0)$ is also maximized within U with probability going to 1. More specifically, for any $\epsilon > 0$, there is $\delta > 0$ and sufficiently large T so that

$$\begin{aligned} P(\hat{\boldsymbol{\beta}} \notin \boldsymbol{U}) &= P\{\sup_{\boldsymbol{U}^c} L(\Theta) \geq \sup_{\boldsymbol{U}} L(\Theta)\} \\ &\leq P\{L(\Theta_1) \geq L(\Theta_0)\} \\ &\leq P\{L(\Theta_1) - \mathbb{E}(L(\Theta_1)) \geq \delta\} + P\{\mathbb{E}(L(\Theta_1)) - \mathbb{E}(L(\Theta_0)) > -2\delta\} \\ &\quad + P\{\mathbb{E}(L(\Theta_0)) - L(\Theta_0) \geq \delta\} \\ &\leq \epsilon/2 + 0 + \epsilon/2 \\ &= \epsilon\,. \end{aligned}$$

□

4.2.4 Computational Complexity

The state-of-the-art cumulants-based method (NPHC) [ABG17] has a complexity of $O(NU^2 + N_{\text{iter}}U^3)$, where N_{iter} is the number of iterations for SGD (around 200 for our applications). Our method has a similar complexity $O(NU^2 + N_{\text{iter}}U^3 + (N_rN_t)^3)$ as NPHC since the calculation time of spatiotemporal cumulants is just a constant multiple of temporal cumulants. The additional calculation for triggering density estimation is usually neglectable because N_r, N_t are small constants (we use 50 in experiments) and $\boldsymbol{A}$ is usually sparse. For an EM-type algorithm (EM) [LM11], the complexity is $O(N_{\text{iter}}N^3U^2)$ [ABG17]. With some clever implementation or in some special cases (e.g. temporal Hawkes process with an exponential triggering density), one can reduce this to $O(N^2)$ or better.

Our method outperforms EM when $N \gg U$. Moreover, in many cases, we find that our method is even faster than NPHC. This seems impossible since our method needs to process spatial data in addition to the timestamp. However, for ST data, many event pairs are close in time (within the support of the temporal triggering density) while spatially separated from each other (outside the support of the spatial triggering density). Temporal-only models such as NPHC will calculate these events pairs during the estimation of cumulants. This might cause false positives in causal inference. Our method, on the other hand, uses spatial information to exclude these events. It seems that, for a majority of data sets we examined, this effect is very significant and our method can be much faster than NPHC.

4.3 Regularization for Linear System

As noted in [SGH18], in many applications, the matrix $\boldsymbol{A}$ in eqs. (4.15) and (4.16) is often ill-conditioned or singular, even with a careful selection of the 2-D grids U_m and V_n . Further, even when it can be obtained, the direct inverse $\boldsymbol{A}^{-1}\boldsymbol{b}$ (or pseudo inverse $(\boldsymbol{A^T A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$) can give unstable results due to overfitting. To solve linear equations stably and robustly, we use regularization procedures to find meaningful approximate solutions. Regularization methods are widely used in mathematical inverse problems since many of the inverse problems are ill-posed. Here one often considers a linear case $Ku = f$, where K is a linear operator and the inverse K^{-1} is unbounded. In this case, it is necessary to change the original problem K to a more stable one with the help of regularization. The regularization can be both linear or nonlinear such as total-variation methods [LZO10].

More specifically, we propose the use of the Tikhonov regularization method [Neu98] with its analytic solution. For example, with the regularization, solving eq. (4.15) becomes this minimization problem

$$\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 + \|\boldsymbol{\Gamma}\boldsymbol{x}\|^2 , \tag{4.23}$$

for a Tikhonov matrix $\boldsymbol{\Gamma} = \alpha\boldsymbol{I}$. This is essentially the L_2 regularization, giving preference to solutions with smaller norms. L_1 regularization will typically give a sparse solution with many zero entities. It does not work here due to the fact that each element in $\boldsymbol{x} = 1/\boldsymbol{\lambda}$ is positive and nonzero. Further one could use other Tikhonov matrices to guarantee smoothness if the underlying vector is believed to be mostly continuous. Instead of this, for the estimation of the triggering density, we smooth g with the post-processing approach below.

We assume that the triggering density is separable in space and time [Oga98]. Note that this assumption is not essential as we can directly obtain the spatiotemporal kernel $g(r, t)$ from $\boldsymbol{\beta}$. It is mainly for comparison with previous methods [FSS16, YLB19] assuming this. As a result, we can decompose the triggering density $g(r, t)$ into the spatial triggering density $f(r)$ and temporal triggering density $h(t)$ (i.e. $g(r, t) = f(r)h(t)$). If we reshape the N_rN_t-

by-1 vector $\boldsymbol{\beta}$ as a N_r-by-N_t matrix $\boldsymbol{B}$, then estimating the spatial and temporal triggering density becomes the following unmixing problem (decomposing the triggering density into densities in space and time):

$$\min_{\boldsymbol{f}\geq 0, \boldsymbol{h}\geq 0} \|\boldsymbol{B} - \boldsymbol{f}\boldsymbol{h}\|^2 . \tag{4.24}$$

Here $\boldsymbol{B}$ is a nonnegative matrix based on the definition of $g(r,t)$ (triggering density function), $\boldsymbol{f}$ is a nonnegative N_r-by-1 vector and $\boldsymbol{h}$ is a nonnegative 1-by-N_t vector. This is, in fact, a rank-one nonnegative matrix factorization (NMF) [LS99] $\boldsymbol{B} = \boldsymbol{f}\boldsymbol{h}$ and we solve it using singular value decomposition (SVD). Finally, we use a Gaussian moving average filter to smooth $\boldsymbol{f}$ and $\boldsymbol{h}$ to obtain the estimation of piecewise-constant triggering densities. This is based on our assumption that g is smooth and it can reduce the variance of our estimation. Our numerical experiments show that the regularization procedure described above leads to a stable and robust estimation for synthetic and real-world data sets.

4.4 Numerical Examples

In this section, we compare our method, which is called ST-Hawkes cumulants (STHC) throughout this section, with other popular estimation methods for multivariate Hawkes processes on various data sets. We consider both simulation data and real-world social-network data. First, we simulate multiple synthetic data sets with different sizes, triggering matrices and triggering densities. These data sets with ground-truth information allow us to examine different methods in detail. Then for real-world applications, we further evaluate the performance of these methods on the task of network reconstruction for multiple location-based social-network check-in data sets. Moreover, our method directly estimates spatial and temporal triggering densities, which provides a useful tool for the study of ST dynamics among these check-in events. We conduct all of the experiments on a single machine with a NVIDIA 970 GPU (4 GB memory), 4-core Intel i7-6700K CPU (4.20 GHz), and 16 GB of RAM.

4.4.1 Synthetic Data

Our synthetic data sets are generated using algorithm 3 in chapter 3, which is based on the clustering representation of the Hawkes process. We simulate various ST-Hawkes processes and use them to evaluate our method (STHC), the state-of-the-art temporal cumulants method (NPHC) and the EM-type algorithm (EM). The details about the simulation and preprocessing are described at the end of this section. Here we define some error measurements used in this section.

- *Relative error* between the estimated triggering matrix $\hat{\boldsymbol{K}}$ and the ground-truth matrix $\boldsymbol{K}$:

$$\mathrm{RelErr}(\boldsymbol{K}, \hat{\boldsymbol{K}}) = \frac{1}{U^2} \sum_{u,v} \Big(\frac{|K_{uv} - \hat{K}_{uv}|}{|K_{uv}|} \mathbb{1}_{K_{uv} \neq 0} + |\hat{K}_{uv}| \mathbb{1}_{K_{uv} = 0} \Big) \, .$$

- *Mean squared error* (MSE) between the estimated triggering densities (temporal $\hat{h}(t)$, spatial $\hat{f}(r)$ and combined $\hat{g}(r,t)$) and the ground-truth triggering densities (temporal $h(t)$, spatial $f(r)$ and combined $g(r,t)$):

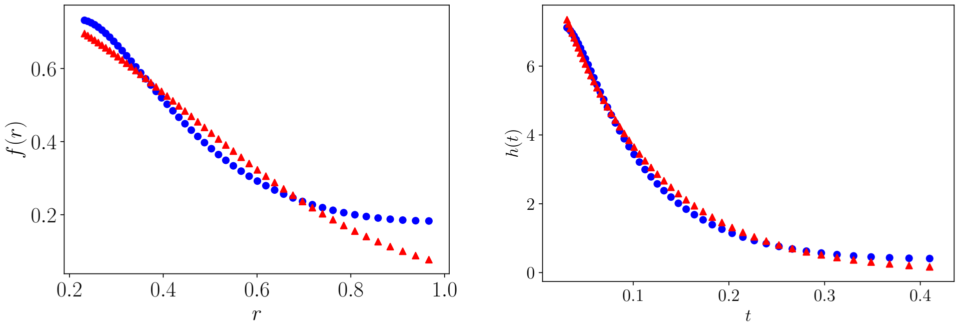
$$\mathrm{MSE}_r = \frac{1}{N_r} \sum_{i=1}^{N_r} (f_i - \hat{f}_i)^2, \;\; \mathrm{MSE}_t = \frac{1}{N_t} \sum_{i=1}^{N_t} (h_i - \hat{h}_i)^2, \;\; \mathrm{MSE}_\beta = \frac{1}{N_r N_t} \sum_{i=1}^{N_r} \sum_{j=1}^{N_t} (g_{ij} - \hat{g}_{ij})^2 \, .$$

Here $\hat{g}_{ij} = \boldsymbol{B}(i,j)$ is the discrete estimation of the triggering density on a 2-D grid of size 50×50 and g_{ij} is the ground-truth value of the triggering density on the grid. $\hat{h}_i = \hat{\boldsymbol{h}}(i)$ and $\hat{f}_i = \hat{\boldsymbol{f}}(i)$ are from the NMF decomposition of $\boldsymbol{B}$, and $h_i = \boldsymbol{h}(i)$ and $f_i = \boldsymbol{f}(i)$ are ground-truth values of the temporal and spatial triggering densities on the grid accordingly.

Triggering Density estimation

We first compare our methods with EM in terms of the triggering density estimation accuracy (NPHC does not estimate triggering densities). The simulation data with 2,587 events are from a ST-Hawkes process with $U = 1$, exponential triggering density in time and Gaussian in space. We get a good estimation of the triggering density $f(r)$ ($\mathrm{MSE}_r \approx 0.001662$), $h(t)$ ($\mathrm{MSE}_t \approx 0.02876$) in fig. 4.1 and the overall estimation for $\boldsymbol{\beta} = (g_{mn})_{k(m,n)}$ ($\mathrm{MSE}_\beta \approx$

Figure 4.1: The estimation results of STHC on $U = 1$ data. Ground-truth spatial triggering density $f(r)$ as red triangles and estimated triggering density as blue circles (left). Temporal triggering density $h(t)$ as red triangles and estimated triggering density as blue circles (right).



0.03400). This is a relatively small data set so that we can use EM for ST-Hawkes (ST-EM, see [YLB19]) estimation. For ST-EM, we get $f(r)$ ($\text{MSE}_r \approx 0.01485$), $h(t)$ ($\text{MSE}_t \approx 0.004058$) and $\boldsymbol{\beta}$ ($\text{MSE}_\beta \approx 0.2533$). Our method is faster (see table 4.1) and overall more accurate.

Triggering matrix

Then we evaluate the ability of our model to recover the triggering matrix $\boldsymbol{K}$. This is important for many applications such as network reconstruction and causal inference. On our existing architecture, the ST-EM method runs out of memory. Instead, we use EM and NPHC implementations in the tick package [BBG17] for the following comparisons.

We simulate a ST-Hawkes process with $U = 100$ and a symmetric $\boldsymbol{K}$ matrix (see fig. 4.2) because our network reconstruction data sets mainly have undirected social networks. We achieve a relative error of 0.1080. In the same setting, we get a relative error of 0.1626 for NPHC and 0.1459 for EM. The improvement in computation time (see table 4.1) is significant.

Figure 4.2: Ground-truth $\boldsymbol{K}$ matrix, STHC (the first row, from left to right), NPHC and EM estimation results (the second row, from left to right).

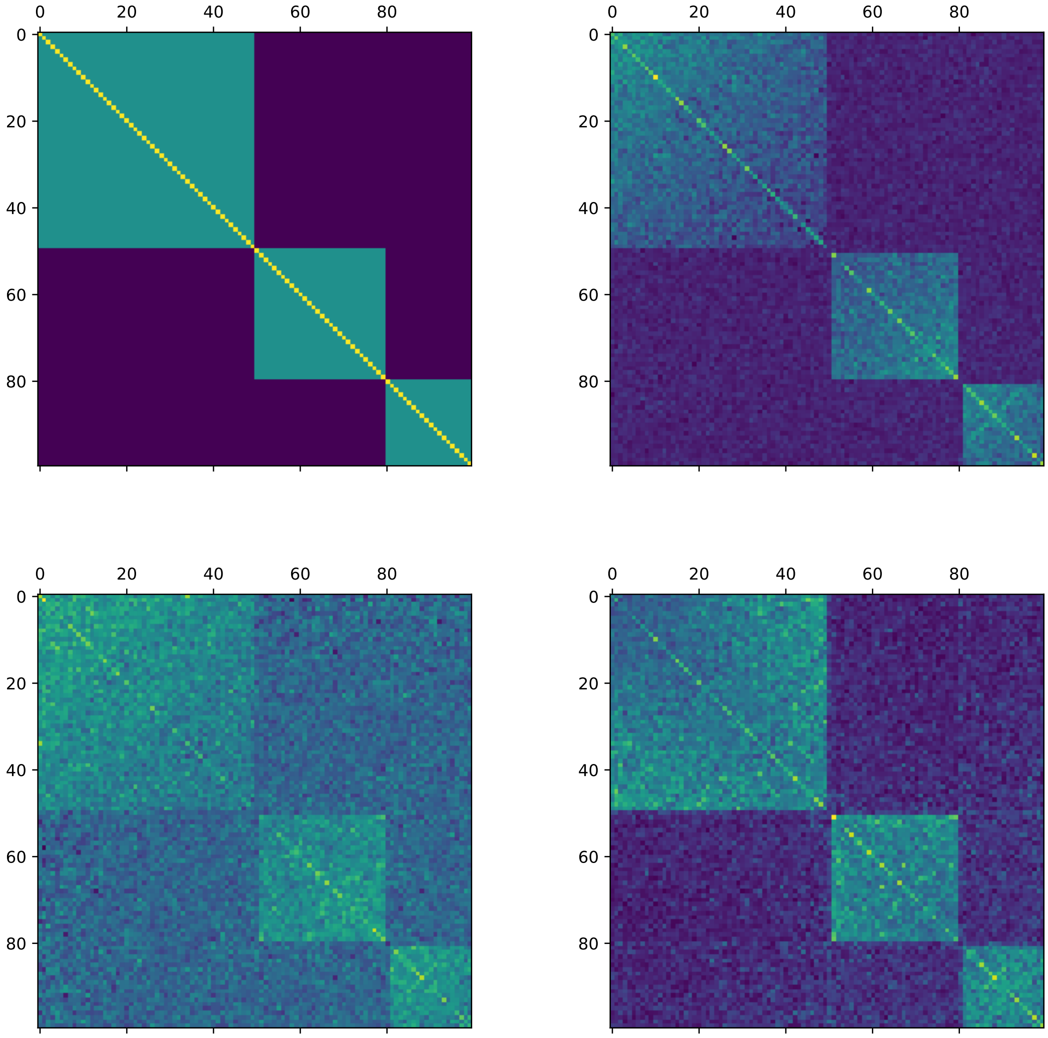
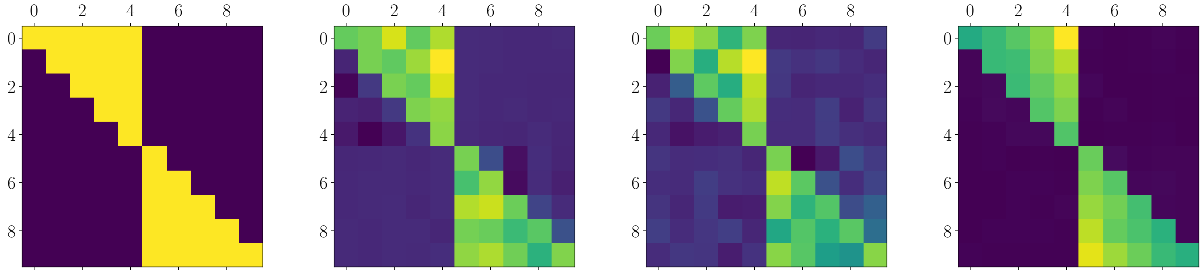


Table 4.1: The computation time for different methods on synthetic data sets. Here the time is in seconds.

	STHC	NPHC	EM
$U = 1$	0.165528	-	4.643132
$U = 10$	1.073085	1.093068	4.707377
$U = 100$	2.608996	4.174796	43.781988

Figure 4.3: Ground-truth $\boldsymbol{K}$ matrix, STHC, NPHC and EM estimation results (from left to right).



Combined estimation

Now we combine the two steps together and give a complete estimation of ST-Hawkes processes. We simulate a ST-Hawkes process with $U = 10$ and 179,176 events in total. From the results in fig. 4.3 and table 4.1, STHC gives very fast and also accurate estimations ($RelErr \approx 0.02901$) comparing to NPHC ($RelErr \approx 0.04899$) and EM ($RelErr \approx 0.03269$). We then threshold $\hat{\boldsymbol{K}}$ with $\epsilon = 0.01$ to remove noise. Using $\hat{\boldsymbol{K}}, \hat{\boldsymbol{\mu}}$, we get a good estimation of the triggering density $f(r)$ and $h(t)$ in fig. 4.4 with $\text{MSE}_r = 0.002381$, $\text{MSE}_t \approx 0.06664$ and $\text{MSE}_\beta \approx 0.1067$ while EM has a much worse MSE ($\text{MSE}_t \approx 0.9512$) since it does not consider spatial information.

Figure 4.4: The estimation results of STHC on $U = 10$ data. Ground-truth spatial triggering density $f(r)$ as red triangles and estimated triggering density as blue circles (left). Temporal triggering density $h(t)$ as red triangles and estimated triggering density as blue circles (right).

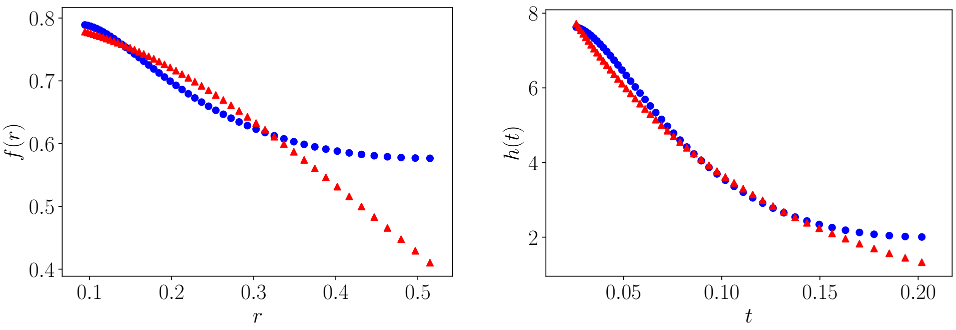


Table 4.2: Error measures for STHC on $U = 10$ data sets with different triggering densities.

	MSE_r	MSE_t	$\mathrm{RelErr}(\boldsymbol{K}, \hat{\boldsymbol{K}})$
Pareto in time	0.01244	0.0009966	0.02784
Uniform in time	0.01320	1.296×10^{-5}	0.09306
Power-law in space	0.0003904	0.04463	0.0409
Uniform in space	0.0006231	0.1294	0.04552

Combined estimation with different triggering densities

We modify the $U = 10$ data set above via replacing the ST triggering density with different functions. We first get accurate estimations of $\tilde{\boldsymbol{K}}$ and $\tilde{\boldsymbol{\mu}}$. Given $\tilde{\boldsymbol{K}}$ and $\tilde{\boldsymbol{\mu}}$, we then estimate the triggering density in space and time (See figs. 4.5 and 4.6). The results are summarized in table 4.2. Specifically, we consider Pareto triggering density in time, uniform triggering density in time, power-law triggering density in space and uniform triggering density in space. See the end of this section for more details on generating these synthetic data sets.

Here we provide more details about simulation data sets used in this section.

Figure 4.5: The estimation results of STHC on $U = 10$ data with a Pareto triggering density in time, a uniform triggering density in time (the first row, from left to right), a power-law triggering density in space and a uniform triggering density in space (the second row, from left to right). Ground-truth spatial triggering density $f(r)$ as red triangles and estimated triggering density as blue circles.

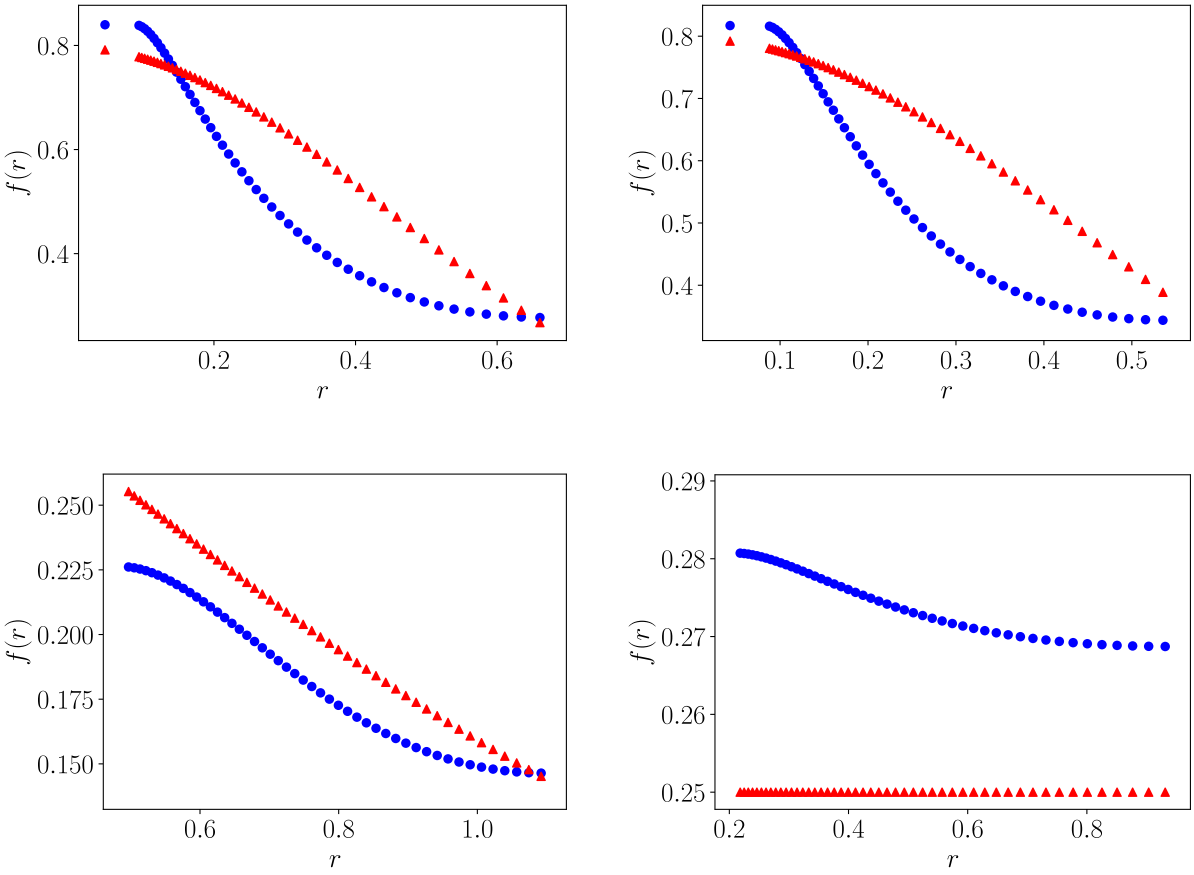
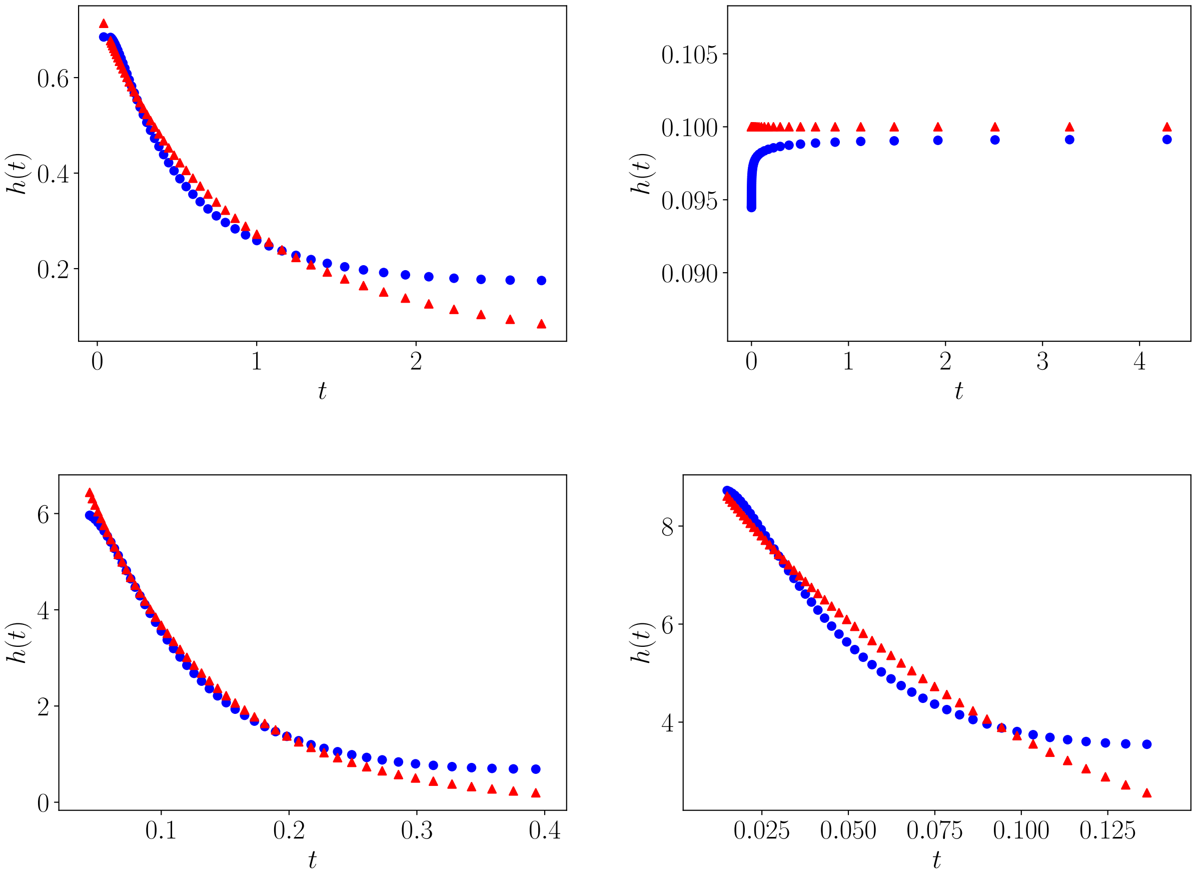


Figure 4.6: The estimation results of STHC on $U = 10$ data with a Pareto triggering density in time, a uniform triggering density in time (the first row, from left to right), a power-law triggering density in space and a uniform triggering density in space (the second row, from left to right). Ground-truth temporal triggering density $h(t)$ as red triangles and estimated triggering density as blue circles.



$U = 1$ Data

We simulate a univariate ST-Hawkes process with $K = 1/6$, $\mu = 0.01$, $T = 2.1 \times 10^5$, $X, Y \in (0, 10)$, $f(r) = \frac{1}{2\pi\sigma^2} \exp(-r^2/2\sigma^2)$ (with $\sigma^2 = 0.2$) and $h(t) = \omega \exp(-\omega t)$ (with $\omega = 10$). The regularization parameter $\alpha = 0.5$.

$U = 100$ Data

Using the same triggering densities, this data set has the following parameters: $U = 100$, the background rate $\boldsymbol{\mu} = (0.01, ..., 0.01)$. We choose $T = 10^5$, $X, Y \in (0, 10)$, $\sigma^2 = 0.2$ and $\omega = 10$ with 172,943 events. For the triggering matrix in fig. 4.2, each yellow, cyan and dark pixel represent the value 1/20, 1/40 and 0 separately.

$U = 10$ Data

With the same densities, the parameters are $U = 10$, $\boldsymbol{\mu} = (0.01, ..., 0.01)$, $T = 1e6$, $X, Y \in (0, 10)$, $\sigma^2 = 0.2$, $\omega = 10$ and $\boldsymbol{K}$ is shown in fig. 4.3. Here each yellow pixel is 1/6 and dark pixel is 0. The regularization parameter $\alpha = 0.55$.

$U = 10$ Data with a Pareto Triggering Density in Time

We keep the same parameters as the $U = 10$ above. The changes on densities are the temporal density $h(t) = (p - 1)c^{p-1}/(t + c)^p$ with $c = 2$ and $p = 2.5$ and the same spatial triggering density with $\sigma^2 = 0.1$. The regularization parameter $\alpha = 0.38$.

$U = 10$ Data with a Uniform Triggering Density in Time

Similar to the section above, we change the temporal densities to be uniform $h(t) = 0.1$ and the spatial triggering density with $\sigma^2 = 0.1$. The regularization parameter $\alpha = 0.4$. We threshold the estimated $\tilde{\boldsymbol{K}}$ with $\epsilon = 0.01$ to remove noise.

$U = 10$ Data with a Power-law Triggering Density in Space

Similarly, we use the power-law density $f(r) = \frac{1}{(r^2+1)^2}$ in space and the exponential triggering density in time with $\omega = 10$. The regularization parameter $\alpha = 0.28$. We threshold the estimated $\tilde{\boldsymbol{K}}$ with $\epsilon = 0.02$ to remove noise.

$U = 10$ Data with a Uniform Triggering Density in Space

Given the same parameters as above, we change the spatial density to $f(r) = 0.25$ and keep the exponential triggering density in time with $\omega = 10$. The regularization parameter $\alpha = 0.36$. We threshold the estimated $\tilde{\boldsymbol{K}}$ with $\epsilon = 0.01$ to remove noise.

4.4.2 Location-based Social-Network Reconstruction

In many situations, network data are incomplete and it may not be possible to directly observe the hidden relationships between nodes. Our task of network reconstruction is to uncover the ground-truth friendship network among social media users using only the information of each user's check-ins.

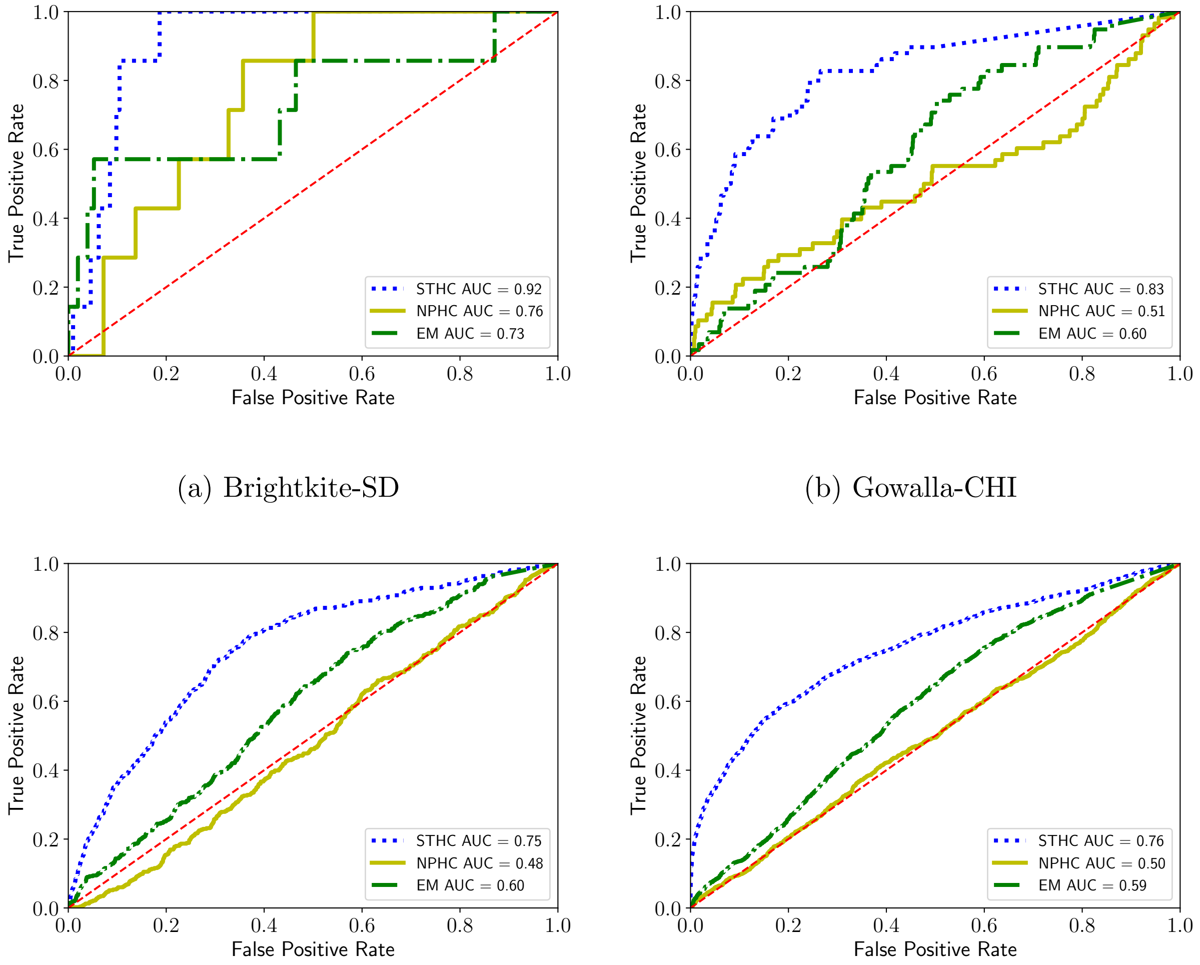
The Gowalla and Brightkite data sets, collected in [CML11], are both from location-based social-media websites in which users share their locations by checking in. Gowalla has a "friendship" network with 196,591 users, 950,327 edges, and a total of 6,442,890 check-ins of these users between February 2009 and October 2010. Brightkite's "friendship" network consists of 58,228 nodes and 214,078 edges, and a total of 4,491,143 check-ins throughout Apr. 2008 – Oct. 2010. Each check-in record includes the latitude and longitude coordinates, a user ID and the time (with a precision of one second). Similar to the Facebook "friendship" network, both the Gowalla and Brightkite friendship networks are undirected and unweighted. We study several subnetworks (Gowalla-SF, Brightkite-LA, Gowalla-CHI, and Brightkite-SD) within these data sets; see the end of this section for details.

We model the ST check-ins of each user within a subnetwork as events of one subprocess within a multivariate ST-Hawkes process. Then we infer relationships between these users (i.e. infer adjacency matrix) from the triggering matrix $\boldsymbol{K}$, which will uncover the macro-scale causality between users (each user is viewed as a subprocess). Our assumption here is that this causality information reflects actual friendship connections. We compare our method (STHC) with NPHC and EM in terms of how well the reconstructed networks match the friendship information from social media. With the prior information that friendship networks are undirected, we first symmetrize the inferred triggering matrix (via $\tilde{\boldsymbol{K}} = \left(\hat{\boldsymbol{K}} + \hat{\boldsymbol{K}}^T\right)/2$) to obtain the estimated weighted adjacency matrix. Then the network reconstruction becomes a binary classification problem with the probability $\propto \tilde{\boldsymbol{K}}$. Given the ground-truth binary adjacency matrix, we calculate the corresponding receiver operating characteristic (ROC) curves and the area under the curve (AUC) to evaluate the results.

The performances of different methods are examined on various subnetworks with different sizes. Our STHC method consistently outperforms other methods with more than 20% improvement in terms of the AUC in fig. 4.7. The improvement is mainly from the ability of our method to exclude false-positive connections. We show an example of network reconstruction results of Brightkite-SD in fig. 4.8. For the computation time (See table 4.3), STHC scales better than NPHC in all data sets, as explained in section 4.2.4. EM has the worst scaling due to its super-linear complexity. Finally, we estimate spatial and temporal triggering densities with $N_r = N_t = 50$ for these subnetworks and plot them in figs. 4.9 and 4.10. The spatial triggering densities for different subnetworks have similar shapes with a cut-off around 10^{-4}. This could come from the fact that the check-in location is usually fixed for a point of interest (POI, such as shop/cafe/gym). The triggering density also implies that the spatial triggering effects between users have a short radius, which will occur when they visit the same POI. These temporal triggering densities also share the same trend. The triggering effects only peak a few hours after the event time. This is also observed in other data sets, such as the insurgency activity in Iraq [LM11].

In the end, we describe the preprocessing procedure for Gowalla and Brightkite data sets. We focus on various local friendship subnetworks within different U.S. cities, including San Diego (SD), Chicago (CHI), Los Angeles (LA) and San Francisco (SF). They have diverse

Figure 4.7: ROC curves of different methods (STHC, NPHC and EM) on subnetworks in Gowalla and Brightkite data sets. The dashed line (red) is from random guess.



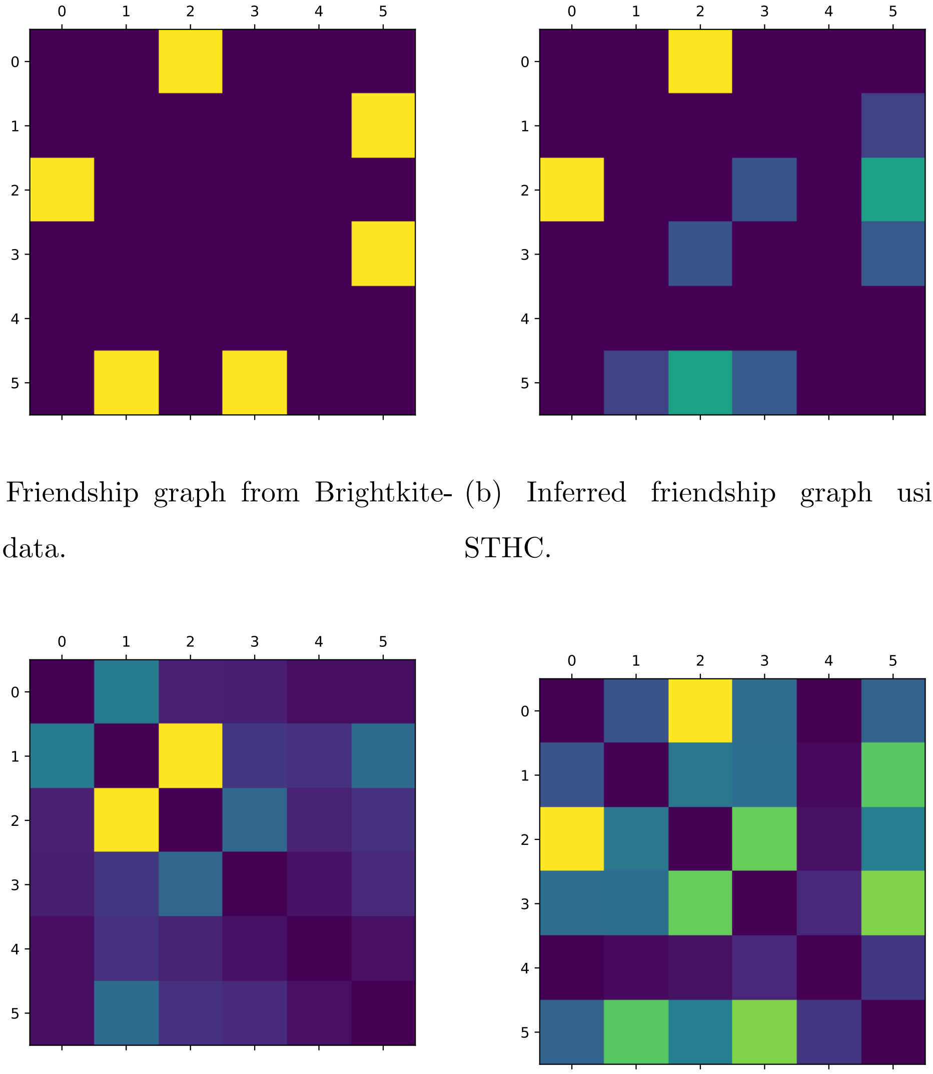
(a) Brightkite-SD

(b) Gowalla-CHI

(c) Brightkite-LA

(d) Gowalla-SF

Figure 4.8: Friendship network reconstruction using different methods on Brightkite-SD. Here we zoom in to show a subgraph within the Brightkite-SD network.



(a) Friendship graph from Brightkite-SD data.

(b) Inferred friendship graph using STHC.

(c) Inferred friendship graph using NPHC.

(d) Inferred friendship graph using EM.

Figure 4.9: Estimated spatial triggering densities for Brightkite-SD, Gowalla-CHI (the first row, from left to right), Brightkite-LA, and Gowalla-SF (the second row, from left to right). The plot is in the log-log scale and we normalize the triggering density for easy comparison. Note that the drop in $f(r)$ when r is between 0.0001 and 0.001 might be due to the artifact of our model such as the choice of the support for triggering densities.

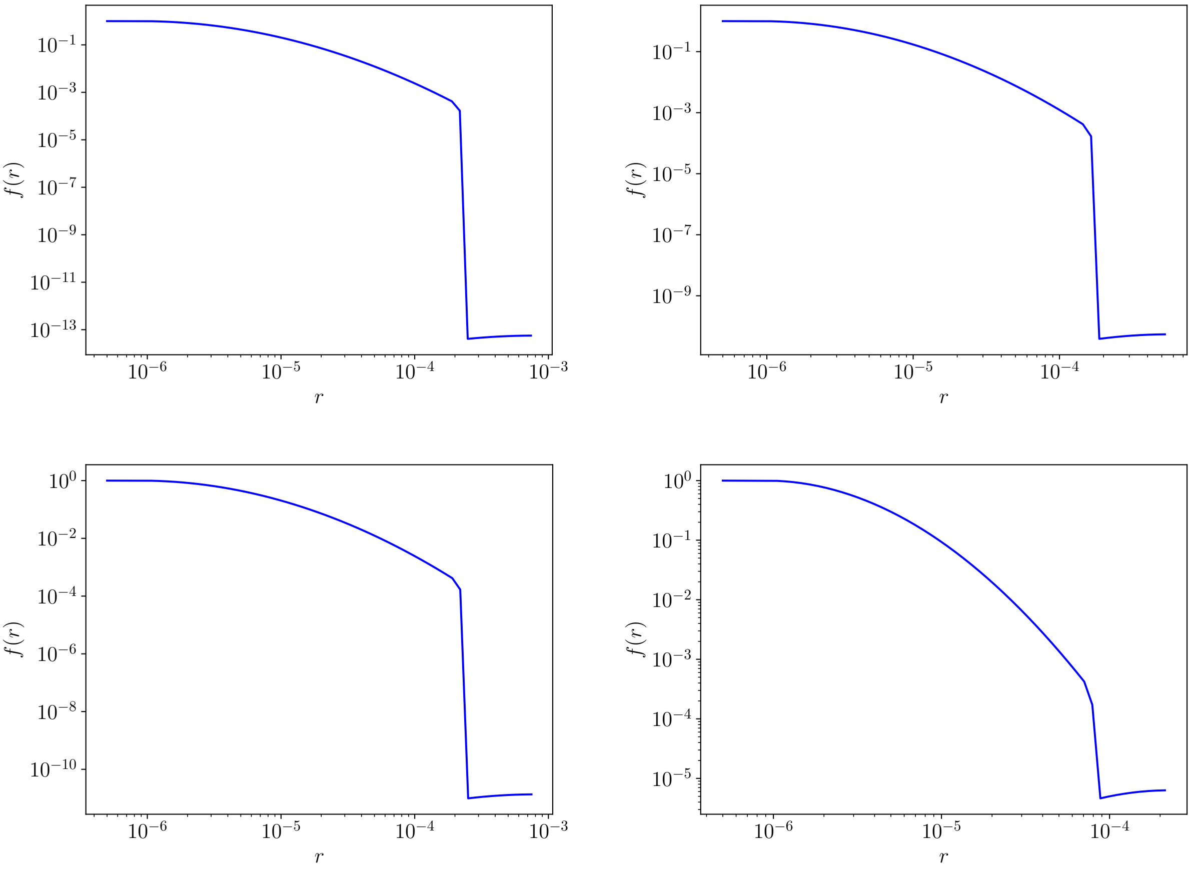
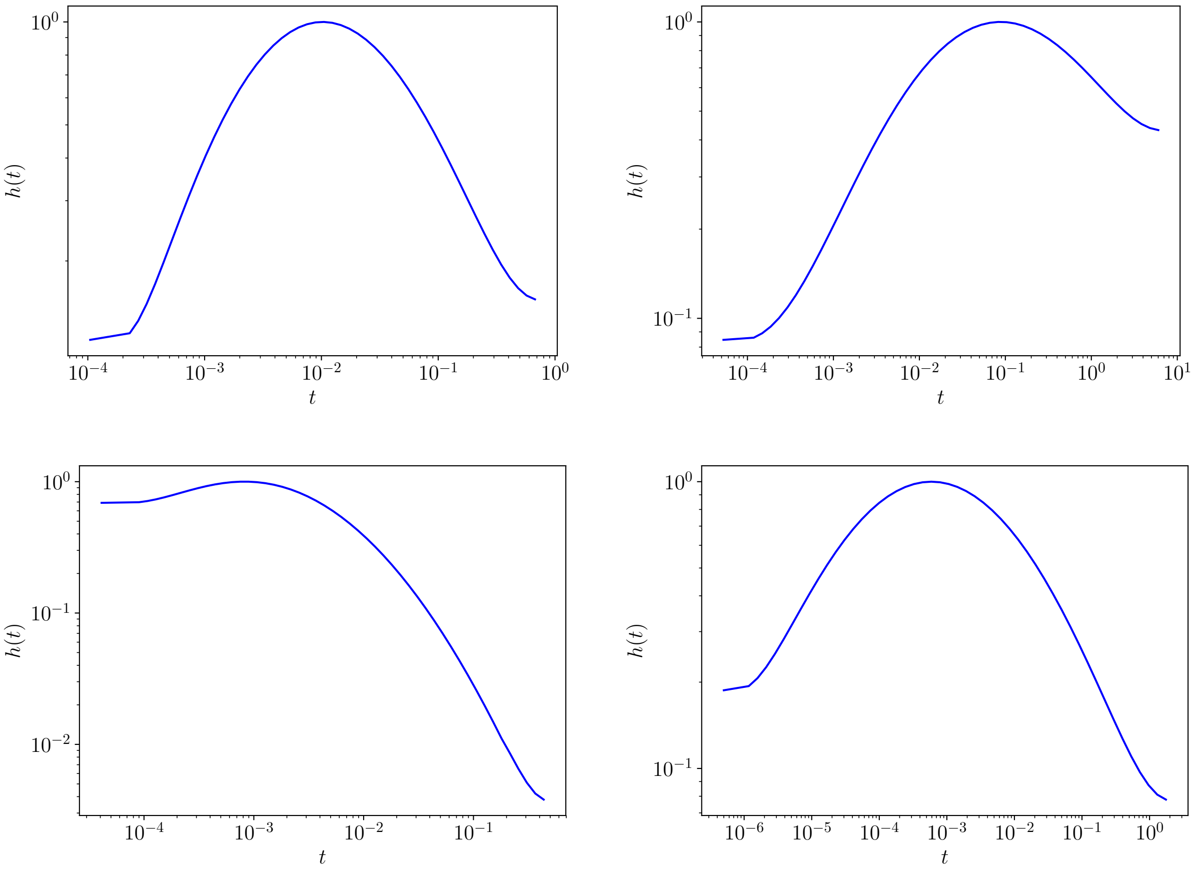


Table 4.3: The computation time for different methods on Gowalla and Brightkite data sets. Here the time is in seconds.

	STHC	NPHC	EM
Brightkite-SD	0.271304	2.035561	2.252009
Gowalla-CHI	2.978064	3.869652	15.474624
Brightkite-LA	3.976395	7.001311	36.357789
Gowalla-SF	40.754037	76.514422	180.918273

Figure 4.10: Estimated temporal triggering densities for Brightkite-SD, Gowalla-CHI (the first row, from left to right), Brightkite-LA, and Gowalla-SF (the second row, from left to right). The plot is in the log-log scale and we normalize the triggering density for easy comparison.



network sizes and ST patterns within the same period.

Brightkite-SD

We study check-ins in SD for the Brightkite data set. We use a bounding box (with a north latitude of 33.1142, a south latitude of 32.5348, an east longitude of −116.9058, and a west longitude of −117.2824)[1] to locate check-ins in SD. We consider "active" users, who have at least 301 check-ins during the period. This gives us a small subnetwork with 25 "active" users and a total of 13,760 check-ins in SD.

Gowalla-CHI

We apply the same procedure as in Brightkite-SD on the Gowalla check-in data for CHI. The bounding box for CHI has a north latitude of 42.0229, a south latitude of 41.6446, an east longitude of −87.5245, and a west longitude of −87.9395. After selecting only active users (with at least 101 check-ins) users, we have a medium-sized subnetwork with 96 users and 27,326 check-ins.

Brightkite-LA

We apply the same procedure as in Brightkite-SD on the Brightkite check-in data in LA. The bounding box for LA has a north latitude of 34.34, a south latitude of 33.70, an east longitude of −118.16, and a west longitude of −118.67. After selecting only active users (with at least 151 check-ins) users, we have a medium-sized subnetwork with 168 users and 89,127 check-ins.

Gowalla-SF

We apply the same procedure as in Brightkite-SD on the Gowalla check-in data in SF. The bounding box for SF has a north latitude of 37.93, a south latitude of 37.64, an east longitude

[1] We obtain latitude and longitude coordinates from `https://www.flickr.com/places/info`.

of -122.28, and a west longitude of -123.17. After selecting only active users (with at least 66 check-ins) users, we have a large subnetwork with 515 users and 102,673 check-ins.

4.5 Conclusion

We present a novel inference approach of ST-Hawkes processes; it is the most efficient and accurate method in comparison to other popular estimation methods. Moreover, this approach is successfully applied to network reconstruction problems and leads to promising applications for the inference of causal relationships and social interactions.

A point that should be stressed is that we make a few model assumptions to simplify the estimation procedure. To recapitulate (see section 4.2.1 for details), we assume a constant background rate in space and no boundary effect for events outside the area that we studied. For a more general spatial background (inhomogeneous) distribution, one can approximate it using a piece-wise constant function in space by dividing events into spatial grids. Essentially, for each grid, we still have a uniform background for estimation and then combine them. For applications on large areas with an inhomogeneous background, we expect a piece-wise constant or covariate-based background rate to achieve even better results [SGH18]; and incorporating boundary effects helps remove bias in the estimation of the background rate and triggering densities [Rei18b].

Finally, while we are focusing on the general case of multivariate ST-Hawkes processes, the current method can be very useful for the estimation of univariate models. The regularization improves the stability and robustness of the analytic method in [SGH18]. This makes it possible to apply univariate models to the study of large data sets in areas such as seismology, epidemiology, and criminology.

CHAPTER 5

Variational Autoencoders for Highly Multivariate Point Processes

As we discussed in the introduction and background chapters, multivariate point processes are widely used to model events of multiple types occurring in a nonempty compact connected metric space. This chapter focuses on the special case of multivariate spatial point processes (SPP), which can uncover hidden connections between subprocesses based on the correlations of their spatial point patterns. Often we encounter missing data problems, where some subprocesses are not fully observable. The underlying connections could further contribute to the prediction of these subprocesses over the unobserved areas. [MAA18] has shown the effectiveness of this joint model for Gaussian processes with heterotopic data. Multi-output models in [LHR15] such as coregionalization and cokriging can outperform independent predictions. However, there is limited literature on the statistical methodology of the highly multivariate spatial point processes, according to the very recent paper [CCC19].

Inference for multivariate spatial point processes is still a challenging problem [TDR15], especially with a large number of subprocesses. For popular Gaussian processes-based approaches [WR06], the multivariate intensity often consists of independent and multi-output Gaussian processes. The complexity of models and the curse of dimensionality hinder this approach for highly multivariate data, such as friendship networks and recommender systems with millions of users. In these problems, we only partially observe the events (e.g. users interact with items and/or locations) for each user, which is viewed as a subprocess. It is necessary to jointly infer the preference of each user based on their hidden correlations. For example, a common approach in recommender systems, collaborative filtering [HLZ17], predicts the item interests of each user with the help of the collection of item preferences for a large number of users.

To address these problems, we propose a multivariate spatial point-process model with a nonparametric intensity. We extend the well-known kernel estimator in [Dig85] to the multivariate case. This generalization is achieved through the introduction of hidden variables inspired by stochastic declustering [ZOV02]. The latent variables naturally lead to a variational Bayesian inference approach, which is different from the frequentist point estimation in the kernel estimator. To reduce the complexity in the highly multivariate case, we consider an alternative set of hidden variables that are designed to work well as latent variables for a variational autoencoder (VAE) [KW14]. This amortized inference [GG14] approach leads to fast inference once the model is fully trained. Further, we show the equivalence for these two different settings of hidden variables using the properties of spatial point processes. This efficient approach makes it possible to apply multivariate spatial point processes in many areas, including location-based social networks and recommender systems with many users. Moreover, the nonparametric method for analyzing spatial point data patterns is not related to specific parametric families of models, which only requires the intensity to be well-defined.

Our approach is not a direct replacement for current inference methods on few-variate spatial point processes [JGM15]. In contrast to the classical methodology, VAE requires a large number of training data. The highly multivariate data that are widely available in social networks and recommender systems can be ideal applications for our approach. In fact, it can be shown that our model is a generalization of a state-of-the-art VAE-based collaborative filtering model [LKH18]. Our model nonparametrically fits the underlying intensity function. Compared with the multinomial distribution used in [LKH18], this leads to not only a smoother intensity over space but also better predictions in terms of ranking-based losses. Compared to a univariate model, such as trans-Gaussian Cox processes [WR06], our multivariate model enhances the predictive ability on missing or unobserved areas, which is consistent with the results of heterogeneous multi-output Gaussian processes [MAA18].

The contributions of this chapter are three-fold. We first build a novel multivariate spatial point-process model and find a direct connection with the VAE-based collaborative filtering through detailed theoretical analysis. Secondly, this connection introduces amortized inference for an efficient multivariate point process estimation. Finally, point processes generalize the discrete distribution used in [LKH18] and lead to a better modeling of spatial heterogeneity. We validate these benefits through experiments with multiple multivariate data sets, showing improvement over classic SPP methods and potentials on collaborative filtering applications.

5.1 Variational Autoencoders

As a stochastic variational inference algorithm, VAE [KW14] is maximizing the evidence lower bound (ELBO) of the log-likelihood function

$$\log p(X|\Theta) \geq \mathbb{E}_{q_\phi(z|X)}[log(p_\theta(X|z)] - KL(q_\phi(z|X)|p(z)) \,. \tag{5.1}$$

The hidden variables z have a simple multivariate Gaussian prior $p(z) = \mathcal{N}(z; 0, I)$. The true posterior, which is often intractable as in the Cox process, is approximated via a multivariate Gaussian $q_\phi(z|X) = \mathcal{N}(z; \mu_\phi(X), \sigma_\phi(X))$. The KL divergence term in the ELBO can be calculated analytically. VAE uses a multilayer perceptron (MLP) to learn the mean and variance of the approximated posterior directly from the data. The most related work here is a recent VAE-based model for collaborative filtering (VAE-CF) [LKH18]. They assume that each user is a multinomial distribution over items with the log-likelihood $\log p_\theta(X_u|z_u) = \sum_{i=1}^{N} X_{iu} \log \pi_i(z_u)$ for each user u. Here X_u is the observed data of user clicking items, $\pi_i(z_u)$ is the probability that user u clicks the item i and X_{iu} is an indicator function on whether the user u clicked the item i.

5.2 Multivariate Spatial Point Processes

Here we consider a multivariate case of the SPP introduced in chapter 2, with U interdependent univariate point processes in the sample space R. The intensity function is measured in a similar way as the univariate case via $\lambda_u(x) = \lim_{|\Delta x| \downarrow 0} \left(\mathbb{E}\left[S_u(\Delta x)\right] / |\Delta x| \right)$, where $S_u(\Delta x)$ is the number of events within a set Δx for the subprocess u.

5.2.1 A Nonparametric Model

The observed data of multivariate SPP include the location of N_u events $X_u = \{x_i^u\}_{i=1}^{N_u}$ associated with each subprocess u. For each u, the observed event locations follow a Poisson process with spatial intensity $\lambda_u(x)$, which is a realization of the random intensity $\Lambda_u(x)$. Using the nonparametric kernel estimator, the intensity of the subprocess u is estimated by

$$\lambda_u(x) = \sum_{i=1}^{N_u} K_\sigma(x - x_i^u)\,. \tag{5.2}$$

Here $K_\sigma(x)$ is a kernel function and we usually adopt the radial basis function kernel (RBF) where $K_\sigma(x) = \exp(-\|x\|^2/2\sigma^2)$. We ignore the end-correction [Dig85] in this work.

In real-world applications, however, one often encounters the missing data problem, where we cannot directly observe points in certain areas for some subprocesses. Instead, we seek to infer the hidden data from other fully observed subprocesses. Note that $N = \sum_{u=1}^{U} N_u$ is the total number of events. We introduce hidden variables Y_i^u for each event $x_i = 1, ..., N$ and subprocess $u = 1, ..., U$, where $Y_i^u = 1$ if the subprocess u includes event x_i and $Y_i^u = 0$ otherwise. $\mathbb{E}Y_i^u = p_i^u$ is the probability that event x_i is from the subprocess u. Then the intensity process for our multivariate SPP model is

$$\Lambda_u(x) = \sum_{i=1}^{N} Y_i^u K_\sigma(x - x_i)\,, \tag{5.3}$$

for each subprocess u. This model generalizes the kernel density-based intensity to the missing data case. Similarly to the original method, it can be applied to estimate the intensity for both cluster processes such as Cox processes and repulsive ones like determinantal PPs. In order to incorporate prior information and model the data uncertainty, we adopt a variational inference approach for the hidden variables.

5.2.2 Variational Inference

A major drawback of current inference methods for SPP is the introduction of a large number of parameters in the highly multivariate case. For our model, we use an amortized inference approach called VAE [KW14], to avoid the computational complexity of directly estimating the posterior for each subprocess u.

The generative process of our model can be described as follows: For each subprocess u, it has a K-dimensional hidden variable z_u with a multivariate normal prior $z_u \sim \mathcal{N}(0, I_K)$. Here we use a low-dimensional representation and then a nonlinear mapping $f_\theta(z_u) = \{p_i^u\}_{i=1}^N$ transforms z_u so that it has the same dimension as the number of events N. Finally, spatial points of the subprocess u are sampled according to the intensity $\lambda_u(x) = \sum_{i=1}^N p_i^u K_\sigma(x - x_i)$. We approximate $p(z|X)$, which is the intractable posterior distribution of z, with a multivariate Gaussian $q_\phi(z|X) \sim \mathcal{N}(\mu_\phi(X), \sigma_\phi(X))$. As in [LKH18], we use MLPs to learn the nonlinear function $f_\theta(z)$ with parameters θ and the mean and variance with parameters ϕ. The variational bound of our multivariate Cox process model is then

$$\log p(X_u|\Theta) \geq \mathbb{E}_{q_\phi(z_u|X_u)}[log(p_\theta(X_u|z_u)] - KL(q_\phi(z_u|X_u)|p(z_u)) = \mathbf{L}\,. \tag{5.4}$$

The first term in $\mathbf{L}$ is essentially a complete likelihood function. For each subprocess u, it has the following (expected) intensity function

$$\mathbb{E}_{q_\phi(z_u|X_u)}\Lambda_u(x) = \sum_{i=1}^{N_u} p_i^u K_\sigma(x - x_i^u) \tag{5.5}$$

and a Poisson process log-likelihood function from eqs. (2.6) and (5.5))

$$\mathbb{E}_{q_\phi(z_u|X_u)} \log p_\theta(X_u|z_u) = \sum_{i=1}^{N_u} \log\left(\sum_{i=1}^{N_u} p_i^u K_\sigma(x - x_i^u)\right) - \int_R \sum_{i=1}^{N_u} p_i^u K_\sigma(x - x_i^u)dx\,. \tag{5.6}$$

For applications without explicit spatial information, we embed each event into a latent space as a vector. First, we obtain a similarity graph for all events. Then the embedding x_i of i_{th} event in this graph is obtained via graph neural networks (GNNs) such as GraphSAGE [HYL17]. See fig. 5.1 for an illustration of our framework.

5.2.3 Alternative Model

Recall that the hidden variables Y_i^u describe whether the event x_i is from the subprocess u. By definition, we have $\sum_{u=1}^U Y_i^u = 1$ and $\sum_{u=1}^U p_i^u = 1$ for any i. During the training process, it is difficult to normalize the probability p_i^u over all subprocesses (have to use the full data). Moreover, this constraint leads to $g_{uv} < 1$ for $u \neq v$, implying mutual-inhibition behaviors between subprocesses. Instead, we consider an alternative model where p_i^u is the

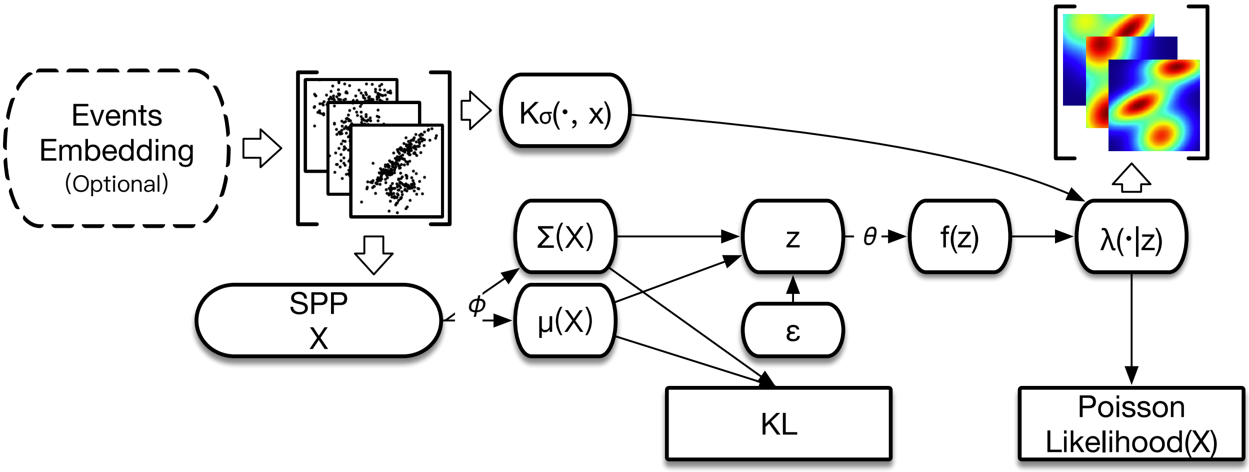


Figure 5.1: Visual illustration of our spatial point-process model via VAE. The spatial events or embeddings X are fed into a neural network with parameter ϕ to get the mean and variance for the approximate posterior $q_\phi(z|X)$, which is then used to calculate the KL divergence term. We can generate samples from the posterior using the reparameterization trick with ϵ from a standard normal distribution. Then the sample z is fed into a neural network with parameter θ to obtain $f(z)$, which combining with the events X and kernel K_σ yields the intensity function λ. Finally, we get the loss function from the KL term and the likelihood function from λ.

probability that the subprocess u generates an event x_i. Then we have $\sum_{i=1}^{N_u} p_i^u = 1$ for each u. During the training, the total number of events N_u is not viewed as a hidden variable for each subprocess. Thus the alternative model essentially normalizes λ_u by a constant. With the reparameterization trick in [KW14], we sample the log-likelihood function using all events within a mini user batch and compute the gradient. This approach incorporates all information about the user so that negative sampling is not needed. See algorithm 4 for our training procedure. For the model prediction, the normalized intensity of a new subprocess can be efficiently calculated in $O(N)$ using the approximated posterior $q_\phi(z|X_{\text{new}})$ and nonlinear function $f_\theta(z)$ with parameters θ, ϕ inferred from data. We can further reduce the computational challenge [LKH18] for large N due to $f_\theta(z)$ by discretizing the space.

Now we show the equivalence of our multivariate model and the alternative one. There are two probabilities to consider. The first one is the conditional probability of observing events X_u in the subprocess u with an intensity function $\lambda_u(x)$, given that there are N_u events within the metric space R. The second one is the probability of sampling X_u of size N_u from the normalized density $h_u(x) = \lambda_u(x)/\int_R \lambda_u(s)ds$. For general SPPs data, we have

Theorem 2. *A spatial point process on a measurable set $R \subset \mathbb{R}^n$ with an intensity function $\lambda_u(x)$ is equivalent to N_u independent and identically distributed (i.i.d) samples within R with a probability density function (p.d.f) $h_u(x) = \lambda_u(x)/\int_R \lambda_u(s)ds$, given we know $N_u = \int \lambda_u(s)ds$, which is the number of points within R for the point-process model.*

Proof. We define our model as a point process on R with the intensity function $\lambda_u(x)$.

The alternative model consists of N_u *i.i.d* samples within R with the *p.d.f* $h_u(x)$, given that we know that $N_u = \int \lambda_u(s)ds$ is the number of points within the point-process model.

(1) Our model has the following probability generating functional:

$$G(v) = \exp\left(-\int_{\mathbf{R}^d} [1 - v(x)]\, \Lambda(\mathrm{d}x)\right). \tag{5.7}$$

(2) Given N_u, we have that

$$p(x_1, ..., x_{N_u}|N_u) = \prod_{i=1}^{N_u} h_u(x_i). \tag{5.8}$$

(3) Our alternative model (a counting random variable $N(x)$ with locations according to $h_u(x)$) has the following characteristic functional

$$G_c(v) = \sum_{n=0}^{\infty} p(N(R) = n)\mathbb{E}\left[\exp(\int_R \log(v(s))N(ds))|N(R) = n\right]. \tag{5.9}$$

Using (2), we can evaluate this conditional probability

$$\mathbb{E}\left[\exp(\int_R \log(v(s))N(ds))|N(R) = n\right] = \left(\frac{\int_R \lambda_u(s)v(s)ds}{\lambda_u(s)ds}\right)^n. \tag{5.10}$$

Using (2) again and because the point process observation probability is

$$p(\omega) = p(N(R) = N_u)p(x_1, ..., x_{N_u}|N_u) = \frac{1}{n!}\left[\prod_{i=1}^{n} \lambda_u(x_i)\right] \exp\left(-\int_R \lambda(x)dx\right), \tag{5.11}$$

we have

$$G_c(v) = \exp\left(-\int_R \lambda(x)dx\right)\left(1+\sum_{n=1}^{\infty}\frac{1}{n!}(\int_R \lambda(s)v(s)ds)^n\right) = \exp\left(\int_R \lambda(s)(v(s)-1)ds\right) . \tag{5.12}$$

The theorem follows from $G_c(v) = G(v)$ as the probability generating functional completely determines the probability structure of the point process. □

We show that VAE-CF is a special case of our alternative model in the following corollary.

Corollary 2.1.

$$\mathbb{E}_{q_\phi(z_u|X_u)} \log p(X_u|z_u) = \sum_{i=1}^{N_u} \log(h_u(x_i^u)) + C . \tag{5.13}$$

Proof. Define $\lambda_u(x) = \mathbb{E}_{q_\phi(z_u|X_u)}\Lambda_u(x)$. Then we have

$$\begin{aligned}\mathbb{E}_{q_\phi(z_u|X_u)} \log p(X_u|z_u) &= \log\left(p(N(R)=N_u)p(x_1^u,...,x_{N_u}^u|N_u)\right) &(5.14)\\ &= \sum_{i=1}^{N_u} \log\left(h_u(x_i^u)\right) + \log(p(N(R)=N_u)) \ , &(5.15)\end{aligned}$$

where

$$\log(p(N(R)=N_u)) = N_u \log(\int_R \lambda(x)dx) - \log(N_u!) - \int_R \lambda(x)dx \tag{5.16}$$

is a constant given N_u. □

According to this corollary, we can replace the log-likelihood function eq. (5.6) in the ELBO with

$$\mathbb{E}_{q_\phi(z_u|X_u)} \log p_\theta(X_u|z_u) = \sum_{i=1}^{N_u} \log(h_u(x_i^u)) + C . \tag{5.17}$$

Here C is related to the log-likelihood on the number of events N_u, which is a constant because $\int_R \lambda_u(s)ds = N_u$ is observed. One drawback of this approach is that, for the prediction of actual missing data, we cannot infer the number of missing points. Instead, our VAE-based model generates the normalized intensity predicting the possible locations for the missing events. We use the alternative definition of p_i^u from now on.

This result shows that VAE-CF is a special case of this multivariate SPP model over a discrete space X of events. In fact, VAE-CF is the alternative model with a delta function as the kernel ($h_u(x_i) = \lambda_u(x_i)/\int_X \lambda_u(s)ds = p_i^u$), which is equivalent to the SPP model according to the theorem. To better model the spatial heterogeneity of events, one can replace the delta function with other kernels or use more advanced SPP intensities. We simply use a RBF kernel here, resulting in $h_u(x) = \sum_{i=1}^{N} p_i^u \exp(\|x - x_i\|^2/2\sigma^2)$.

Algorithm 4 Training VAE SPP with stochastic gradient descent.

```
Training subprocesses $u \in \mathbf{U_T}$ with their point locations $X_u$
**Inputs**: Parameters $\theta$ and $\phi$.Initialize $\theta$ and $\phi$ randomly.
**while** not converged **do**
  Sample a subprocesses batch $\mathbf{U_b}$ from $\mathbf{U_T}$ and their points $X_b = \bigcup_{u \in \mathbf{U_b}} X_u$.
  **for** $u \in \mathbf{U_b}$ **do**
    Sample $z_u \sim \mathcal{N}(\mu_\phi(X_u), \sigma_\phi(X_u))$ with reparameterization trick.
    Compute $f_\theta(z_u) = \{p_i^u\}_{x_i \in X_b}$.
    **for** $x \in X_u$ **do**
      Compute sampled normalized intensity $h_u(x) \approx \sum_{x_i \in X_b} p_i^u K_\sigma(x - x_i)$..
    **end for**
    Compute noisy gradients of the ELBO $\mathbf{L}$ w.r.t $\theta$ and $\phi$
  **end for**
  Average noisy gradients over batch.
  Update $\theta$ and $\phi$ with the Adam optimizer [KB15].
**end while**
```

One benefit of this alternative model is its resulted consistency. The nonparametric kernel estimation for the point process intensity is unbiased. To see this, for any measurable set R, we take the expectation of the estimated intensity $\lambda(x)$ over the Poisson point process distribution

$$\mathbb{E}\int_R \lambda(x)dx = \int_R \mathbb{E}\sum_{i=1}^{N} K_\sigma(x - x_i)dx = \int_R\int_R K_\sigma(x - y)\rho(y)dydx = \int_R \rho(y)dy\,, \qquad (5.18)$$

where $\rho(y)$ is the true intensity function. Then $\mathbb{E}\lambda(x) = \rho(x)$ under mild conditions, e.g. a spatially continuous assumption on ρ. But it is inconsistent due to the non-vanishing variance without normalization. For our alternative model, the normalized intensity function

$h_u(x)$ is still unbiased. And according to the standard theory of the multivariate kernel density estimation (KDE), the consistency of $h_u(x)$ is also guaranteed. Another benefit of using this alternative form can be seen from the cross pair-correlation function. For the alternative model, we remove the undesirable restriction of negative correlations between all users ($g_{uv} < 1$ for $u \neq v$) and can incorporate more diverse relationships between users. To see this, we first consider the *auto* and *cross pair-correlation function*

$$g_{u,v}(x, y) = \frac{\mathbb{E}[\Lambda_u(x)\Lambda_v(x)]}{\mathbb{E}\Lambda_u(x)\mathbb{E}\Lambda_v(x)} \,. \tag{5.19}$$

For our original model, it is straightforward to prove that $g_{uu} > 1$ and $g_{uv} < 1, u \neq v$ (see below). The auto pair-correlation functions show that our model is more clustered than the simple Poisson process.

Now we show that $g_{u,v}(x, y) > 1$ for $u = v$ and $g_{u,v}(x, y) < 1$ for $u \neq v$ for our original model. We have

$$\mathbb{E}\Lambda_u(x)\Lambda_v(x) = \mathbb{E}\left[\left(\sum_{i=1}^{N} Y_i^u K_h(x - x_i)\right)\left(\sum_{j=1}^{N} Y_j^v K_h(y - x_j)\right)\right] \tag{5.20}$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{N} \mathbb{E}[Y_i^u Y_j^v] K_h(x - x_i) K_h(y - x_j) \,. \tag{5.21}$$

Similarly,

$$\mathbb{E}\Lambda_u(x)\mathbb{E}\Lambda_v(x) = \left(\mathbb{E}\sum_{i=1}^{N} Y_i^u K_h(x - x_i)\right)\left(\mathbb{E}\sum_{j=1}^{N} Y_j^v K_h(y - x_j)\right) \tag{5.22}$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{N} p_i^u p_j^v K_h(x - x_i) K_h(y - x_j) \,. \tag{5.23}$$

Note that $\sum_{u=1}^{U} Y_i^u = 1$ and $\sum_{u=1}^{U} p_i^u = 1$. When $i \neq j$, we have $\mathbb{E}[Y_i^u Y_j^v] = p_i^u p_j^v$ for any u, v. When $i = j$, $\mathbb{E}[Y_i^u Y_i^u] = p_i^u > (p_i^u)^2$ for $u = v$ and $\mathbb{E}[Y_i^u Y_i^v] = 0 < p_i^u p_i^v$. Then it is easy to see $g_{u,v}(x, y) > 1$ for $u = v$ and $g_{u,v}(x, y) < 1$ for $u \neq v$.

5.3 Experiments

We compare our model (with the RBF kernel, VAE-SPP) with both VAE-CF [LKH18] and univariate spatial point-process models using a standard KDE [Dig85] or TGCP [WR06] as intensity functions. We adopt the experiment setting in VAE-CF. We split the data into training, validation and testing sets. For the multivariate model, the training data is used to learn the parameters θ and ϕ. For KDE and TGCP models, we omit the training data because different subprocesses are assumed to be independent and also because of the computational complexity of fitting a highly multivariate TGCP. We assume that only 80% of the events in the validation and test sets are observed. The remaining 20% are viewed as missing data to be inferred by different models. Hyperparameters are selected on the validation data as in [LKH18]. Finally, we compare the prediction performance of different models on the missing data given the partially-observed events. We use standard ranking losses such as NDCG@K and Recall@K defined below.

The ranking performance is evaluated through recall at K (Recall@K) and normalized discounted cumulative gain at K (NDCG@K). In our VAE-SPP model, the predicted rank of the held-out items I_u for each user u is obtained from sorting the intensity function $\lambda_u(x)$. Here we keep the definition in [LKH18]. Recall@K is defined as

$$\text{Recall@K} = \frac{\sum_{i=1}^{K} \mathbb{1}(r_i \in I_u)}{|I_u|} \,. \tag{5.24}$$

NDCG@K is calculated by normalizing discounted cumulative gain (DCG@K) with ideal DCG@K (IDCG@K). The definition are as follows:

$$\text{DCG@K} \quad = \sum_{i=1}^{K} \frac{2^{\mathbb{1}(r_i \in I_u)} - 1}{\log_2(i+1)} \,, \quad \text{NDCG@K} \quad = \frac{\text{DCG@K}}{\text{IDCG@K}} \,, \tag{5.25}$$

where $\mathbb{1}$ is the indicator function and r_i is the i_{th} item among held-out items; IDCG@K is the ideal DCG@K when the ranked list is perfectly ranked.

5.3.1 Multivariate SPP on Spatial Data

Synthetic Data Sets We simulate two different data sets using multiexponential and multisine models. For the multiexponential data set, we simulate 5,000 Poisson processes with $\lambda_k(x) = a_k e^{-b_k x}$, $k = 1, ..., 5{,}000$, $x \in [0, 30]$ as training data. Here a_k and b_k are uniformly sampled between $[5, 10]$ and $[0.1, 0.2]$ separately. We generate 500 validation and 500 test subprocesses in the same way with parameters sampled from a_k and b_k. The multisine data set is generated via replacing the intensity function with $\lambda_k(x) = \max(a_k \sin(b_k x), 5)$ and sampling a_k and b_k uniformly between $[5, 10]$ and $[1, 2]$ separately. Each realization of the spatial point process is discretized using a uniform grid over x with grid spacing 0.01.

Table 5.1: Testing results on the simulation data sets. Both the mean and standard derivation (in parentheses) are percentages.

	Multiexp			Multisine		
Name	NDCG@100	Recall@50	Recall@100	NDCG@100	Recall@50	Recall@100
VAE-CF	6.78(0.28)	7.25(0.40)	14.5(0.52)	3.30(0.15)	2.49(0.13)	4.64(0.18)
VAE-SPP	**7.11**(0.31)	**7.34**(0.40)	**14.9**(0.54)	3.53(0.15)	**2.58**(0.13)	**4.90**(0.18)
KDE	5.27(0.15)	5.85(0.12)	11.8(0.17)	3.23(0.15)	2.29(0.12)	4.55(0.27)
TGCP	3.11(0.14)	3.32(0.11)	6.44(0.11)	**3.77**(0.14)	1.88(0.11)	3.92(0.17)

Location-based Social Network. We consider the Gowalla data set [CML11] in New York City (NYC) and California (CA). We use a bounding box of −124.4096, 32.5343, −114.1308, 42.0095 for CA and −74.0479, 40.6829, −73.9067, 40.8820 for NYC (both from flickr[1]). Each user with at least 20 events (check-ins) is viewed as a subprocess. There are 673,183 events and 6,728 users for Gowalla-CA. We uniformly select 500 users as the validation set and 500 users as the testing set. We use the remaining users for training. For Gowalla NYC, there are 86,703 events from 1,171 users. We set the size of both validation and testing sets to 100. For the spatial tessellation, we use uniform grids (32 × 32 for NYC and 64 × 64 for CA). Both our model and VAE-CF can work without grids. We further compare the performance of our model with VAE-CF by viewing each location as an item.

In table 5.1, we summarize the performance of both multivariate and univariate models on the simulation data sets. It is clear that multivariate models outperform univariate ones. Moreover, testing on multivariate models takes less time because it only evaluates the posterior probability and intensity function. This illustrates the power of multivariate models using amortized inference. Within the multivariate models, our continuous model further

[1] The bounding boxes are from `https://www.flickr.com/places/info/`.

Table 5.2: Testing results on the Gowalla data sets with uniform grids. Both the mean and standard derivation (in parentheses) are percentages.

	CA			NYC		
Name	NDCG@100	Recall@50	Recall@100	NDCG@100	Recall@50	Recall@100
VAE-CF	41.8(1.5)	64.8(2.0)	70.0(2.0)	43.6(2.3)	73.9(2.9)	**86.2**(2.2)
VAE-SPP	**42.3**(1.5)	**65.2**(2.0)	**70.2**(1.9)	**44.8**(2.4)	**74.5**(2.9)	**86.2**(2.2)
KDE	34.5(1.5)	59.2(2.0)	64.0(2.0)	41.2(1.5)	69.9(2.0)	83.6(2.0)
TCGP	31.8(1.3)	56.5(2.0)	60.9(2.0)	37.3(2.3)	59.9(3.3)	75.9(2.8)

Table 5.3: Testing results on the Gowalla data sets without discretization. Both the mean and standard derivation (in parentheses) are percentages.

	CA			NYC		
Name	NDCG@100	Recall@20	Recall@100	NDCG@100	Recall@20	Recall@100
VAE-CF	21.3(0.77)	16.6(0.74)	32.8(0.97)	16.0(1.7)	13.2(1.7)	26.3(2.4)
VAE-SPP	**21.6**(0.77)	**17.0**(0.80)	**33.5**(0.76)	**16.1**(1.7)	**13.7**(1.8)	**27.1**(2.5)

improves upon the discrete VAE-CF. This is due to the fact that these simulation intensities are continuous over R. For real-world applications, the results on the location-based social network prediction and recommender systems with and without grids are presented in tables 5.2 and 5.3. We observe the same pattern in both NYC and CA. We stop using univariate models from now on due to their inferior performances, especially for collaborative filtering applications. Moreover, our model improves discrete VAE-CF regardless of the choice of spatial grids. For visualization purposes, in fig. 5.2, we plot a user's check-in locations in Gowalla-NYC and intensities estimated via different methods. Comparing with VAE-CF, our model generates a continuous intensity. The univariate models overfit the training data and lead to inferior predictions of the missing data.

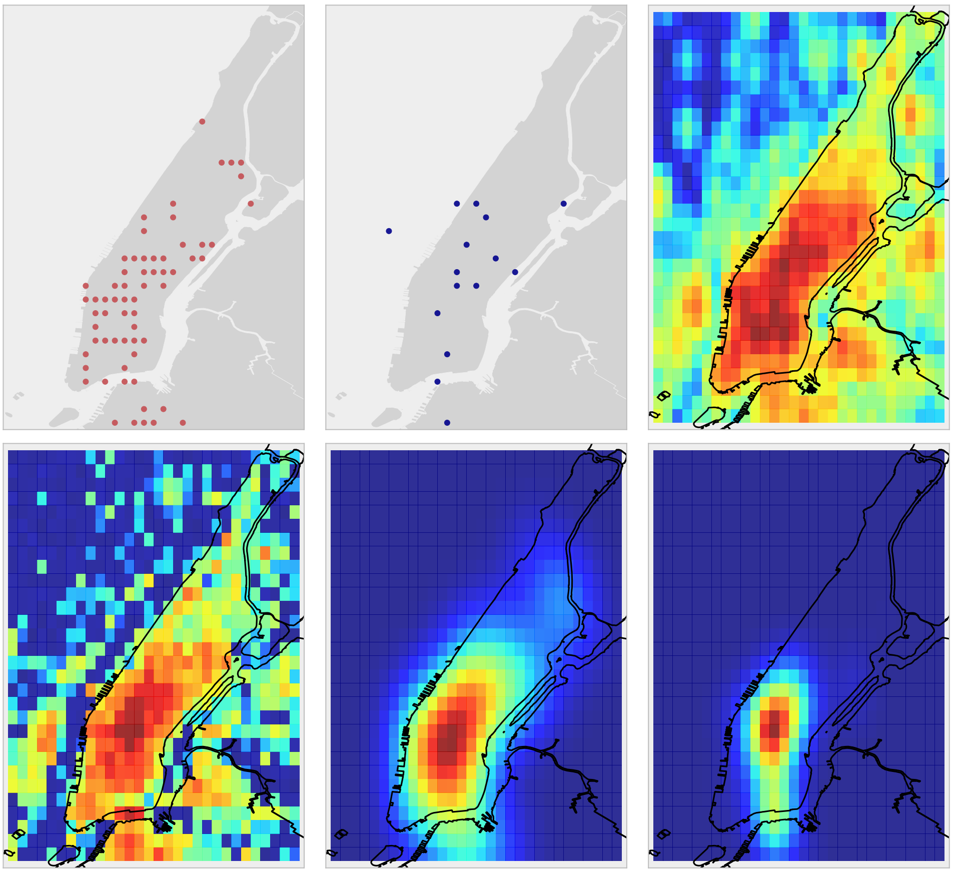


Figure 5.2: Estimated density functions for a Gowalla user in NYC (log scale). The first row from left to right: observed check-in locations (in red), held-out check-in locations (in blue, as missing data) and the estimated intensity from VAE-SPP. The second row from left to right: the estimated intensity (or density) from VAE-CF, KDE and TGCP.

5.3.2 Multivariate SPP with a Latent Space

The MovieLens data sets (ML-100K and ML-1M) include the movie (item) rating by users and we binarize the rating with a threshold of 4. In the spatial point process setting, we view each user as a subprocess over the latent space of item embeddings. Here the item embedding is generated via a GNN. This framework is a natural generalization of the multimodal distribution over items. The item-item graph is constructed based on item-item similarities. We use the Jaccard distance to measure the similarities between items, which are further viewed as the sampling probabilities for GNN. The cosine similarity can also be used here. We choose the Jaccard index because it usually leads to better performance when we measure similarities between users' behavior in the application within e-commerce platforms such as Alibaba. Currently, we only consider 1-hop connections. Both GNN and VAE are trained jointly, which is more expensive than VAE-CF but leads to better performance compared to separate training (see section 5.3.4 for more details). For movie recommendation tasks, we compare the discrete VAE-CF to our joint model with GNN. The results in table 5.4 show again the improvement of our model over the baseline.

Table 5.4: Testing results on the MovieLens data sets.

	ML100K			ML1M		
Name	NDCG@100	Recall@20	Recall@100	NDCG@100	Recall@20	Recall@100
VAE-CF	40.8(2.8)	**32.3**(2.8)	57.6(3.3)	41.6(0.76)	33.1(0.81)	56.8(0.88)
VAE-SPP	**41.5**(2.9)	31.3(2.7)	**59.0**(3.5)	**42.3**(0.77)	**33.9**(0.82)	**57.6**(0.88)

5.3.3 Hyperparameters

We implement our models in Tensorflow based on VAE-CF[2]. We keep the same MLP architecture and hyperparameters for both of them. We use β-VAE as suggested. The only additional hyperparameter for our model is the σ^2 in the kernel function, which is determined using a grid search on the validation set. We conducted experiments on a single GTX 1080 TI 11GB GPU.

For simulation data, we train both models for 200 epochs using Adam optimizer with $\beta = 0.2$ and $l_r = 5 \times 10^{-5}$. We use mini-batches of size 20. Our architectures consist of a one layer MLP with $K = 50$. For VAE-SPP, $\sigma^2 = 0.001$. For Gowalla-NYC data, we train both models for 200 epochs using Adam optimizer with $\beta = 0.2$ and $l_r = 5 \times 10^{-4}$. We use mini-batches of size 20. Our architectures consist of a one layer MLP with $K = 50$. For VAE-SPP, $\sigma^2 = 1 \times 10^{-5}$. For Gowalla-LA data, we train both models for 200 epochs using Adam optimizer with $\beta = 0.2$ and $l_r = 1 \times 10^{-3}$. We use mini-batches of size 20. Our architectures consist of a one layer MLP with $K = 50$. For VAE-SPP, $\sigma^2 = 0.001$. For

[2] The code is available at `https://github.com/dawenl/vae_cf`.

ML-1M data, we train both models for 100 epochs using Adam optimizer with $\beta = 0.2$ and $l_r = 1 \times 10^{-3}$. We use mini-batches of size 5. Our architectures consist of a one layer MLP with $K = 200$. For VAE-SPP, $\sigma^2 = 1 \times 10^{-5}$. For ML-100K data, we train both models for 100 epochs using Adam optimizer with $\beta = 0.2$ and $l_r = 1 \times 10^{-3}$. We use mini-batches of size 5. Our architectures consist of a one-layer MLP with $K = 200$. For VAE-SPP, $\sigma^2 = 1 \times 10^{-5}$. The one-layer GNN in ML data is trained using GraphSAGE, for which the embedding dimension is 32 and the number of the neighborhood is 10 for items and 5 for users. The graph consists of the edges between users and items as well as the edges between items based on their Jaccard similarity.

We use PYTHON statsmodel for the KDE and GPy for TGCP. The bandwidth h for KDE is selected automatically in the statsmodel package. The hyperparameters for TGCP are determined with a grid search on the validation set. For simulation data sets, we set an RBF kernel with variance 1, lengthscale 0.1 for TGCP. For Gowalla-CA data sets, we set a Matern32 kernel with variance 0.001, lengthscale 0.1 for TGCP. For Gowalla-NYC data sets, we set a Matern32 kernel with variance 0.0001, lengthscale 0.01 for TGCP.

5.3.4 Additional Experiments

On the training of VAE and GNN, we tried different training settings (separately or jointly) and choose to train them jointly. We also tested the point estimate version of the VAE-CF called DAE-CF (Mult-DAE in Liang et al., (2018), with the same setting), which can improve the result under certain metrics. One can easily extend our work to a DAE-SPP to obtain a point estimation for the SPP intensity.

Table 5.5: Testing results on MovieLens-100K. These methods share the same network and are trained with 100 epochs. The test data are used to evaluate the model with the best performance during the validation. VAE-SPP-Separate means that GNN is trained separately with VAE-SPP.

	NDCG@100	Recall@20	Recall@100
VAE-CF	40.88	32.32	57.63
DAE-CF	40.98	29.29	58.80
VAE-SPP	41.50	31.34	58.99
VAE-SPP-Separate	41.43	31.15	58.82

We also did experiments on the MLPs for VAE. For Movie Lens 1M, the larger network in VAE-CF leads to a 40.3 NDCG@100 for VAE-CF and 41.9 for VAE-SPP. As a result, we use the smaller one instead.

5.4 Conclusions

In this chapter, we introduce a novel spatial point-process model for efficient inference on the highly multivariate case. Through amortized inference, our model makes it possible to investigate correlations between a myriad of point patterns based on a large number of training data, and the theoretical analysis builds the connection between our model and VAE-CF. There are many promising directions for future work including the extension for spatiotemporal PPs [MSB11] and using features as covariates. Another interesting application is to handle real-world recommender systems via improving the joint training efficiency and comparing thoroughly with simpler algorithms as in [DCJ19].

CHAPTER 6

Novel Applications of Our Methods

In this chapter, we discuss two specific applications of point-process models on real-world problems. The first part is a direct application of the model in chapter 3 on the opioid overdose data. Via clustering different overdose types from the drug structure, we use the multivariate spatiotemporal Hawkes process to model overdoses' dynamics and predict the future outbreaks. The second part is a combination of point-process models with a deep learning approach for time series applications, especially focusing on the real-time crime forecasting.

6.1 Drug Movers' Distance-based Hawkes Processes for Overdose Spike Early Warning

Opioid addictions and overdoses have increased across the U.S. and internationally over the past decade. In urban environments, overdoses cluster in space and time, with 50% of overdoses occurring in less than 5% of the city and dozens of calls for emergency medical services being made within a 48-hour period [CMR19]. In this work, we introduce a system for early detection of opioid overdose clusters based upon the toxicology report of an initial event. We first use drug SMILES, one hot encoded molecular substructures, to generate a bag of drug vectors corresponding to each overdose (overdoses are often characterized by multiple drugs taken at the same time). We then use spectral clustering to generate overdose categories and estimate multivariate spatiotemporal Hawkes processes for the space-time intensity of overdoses following an initial event. As the productivity parameter of the process depends on the overdose category, this allows us to estimate the magnitude of an overdose

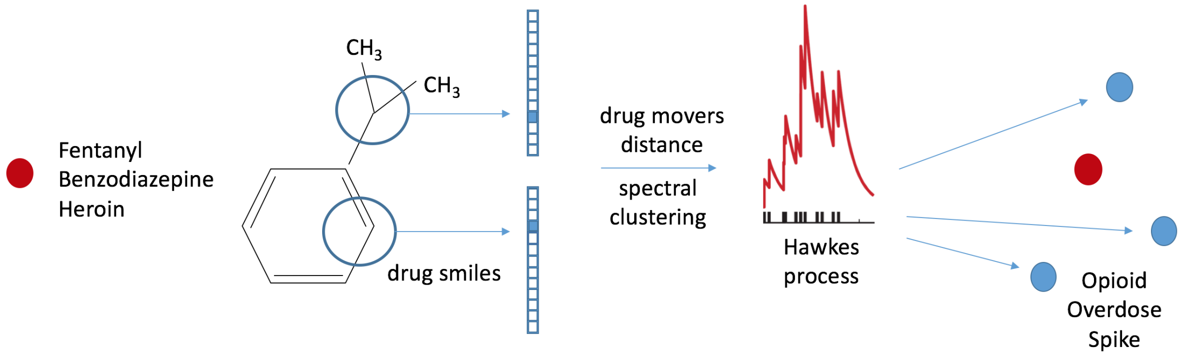


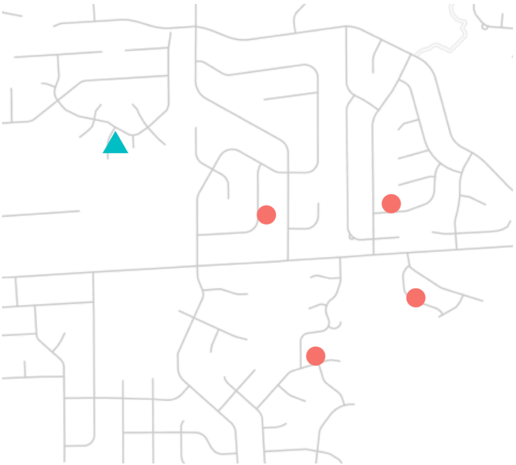
Figure 6.1: Overview of the system for early warning of opioid spikes. The initial overdose toxicology report shows fentanyl, benzodiazepine, and heroin present. Each drug is vectorized using SMILES and the event belongs to an overdose category using spectral clustering based on earth mover's distance of the drug vectors ("drug movers' distance"). The increase in the intensity of the Hawkes process is determined by the category and it allows for the prediction of an opioid overdose spike, triggered in the branching process by the initial overdose.

spike based on the substances present (e.g. fentanyl leads to more subsequent overdoses compared to Oxycontin). In fig. 6.1, we visualize the workflow of our model for an initial overdose event.

We validate the model using opioid overdose deaths in Indianapolis and show that the model outperforms several Hawkes-process models based on Dirichlet processes. See [CYL19] for the details about the results. Here in fig. 6.2 we show an example of triggered events followed from an initial spike. Our system could be used in combination with drug test strips to alert drug-using populations of risky batches on the market or to more efficiently allocate naloxone to users and health/social workers.

6.2 Graph-based Deep Modeling and Real-time Forecasting of Sparse Spatiotemporal Data

In crime forecasting, we develop a generic framework to model unstructured crime data. We combine the multivariate model with a deep neural network (DNN) to improve the



Initial Event
Alcohols, Ethanol, Benzodiazepine, Fentanyl

Triggered Events
1. Benzodiazepine, 6_MAM, Heroin_from_combo, Morphine, Codeine, Hydrocodone, Hydromorphone

2. Alcohols, Ethanol, Benzodiazepine, 6_MAM, Heroin_from_combo, Morphine, Codeine, Oxycodone, Oxymorphone

3. THC, Carboxy_THC, Oxycodone, Oxymorphone, Hydromorphone

4. Benzodiazepine, 6_MAM, Heroin_from_combo, Cocaine, Morphine, Codeine, Fentanyl, Oxycodone, Oxymorphone

Figure 6.2: An illustration for an initial event and its triggered events in one of the categories (i.e. one of the Hawkes processes). The initial overdose event marked in triangle symbol consists of four drug substances and it triggered four neighboring events consisting of different drug substances.

generalization ability. Compared to previous ad hoc spatial partitioning, we build a ST weighted graph (STWG) from the multivariate model to represent the data. This STWG carries spatial cohesion and temporal evolution of the data in different spatial regions over time. STWG is inferred in real-time via our linear and fast algorithm. For crime forecasting, we associate each graph node with a spatiotemporal point process of crime intensity in a zip code region, where each zip code is a node of the graph. The inferred STWG incorporates the macroscale evolution of the crime over space and time and is much more flexible than the lattice representation. To perform micro-scale forecasting of the ST data, we build a scalable graph-structured RNN (GSRNN) on the inferred graph based on the structural-RNN (SRNN) architecture [JZS16]. SRNN is built by arranging RNNs in a feed-forward manner: first, assign a cascaded long short-term memory (LSTM) to fit the time series on each node of the graph. Simultaneously, we associate each edge of the graph with a cascaded LSTM that receives the output from neighboring nodes. Then feed the tensors learned by these edge LSTMs to their terminal nodes. This arrangement of edge and node LSTMs gives a native feed-forward structure that is different from the classical multilayer perceptron. A

Figure 6.3: Flow chart of the algorithm. The spatiotemporal events are fed into a multivariate Hawkes process to obtain the graph of zip codes. The graph is then used in GSRNN with two LSTM networks and a fully connected layer.

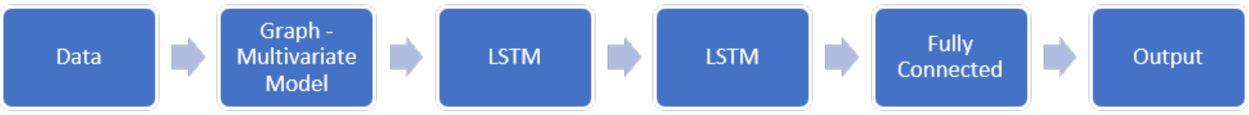
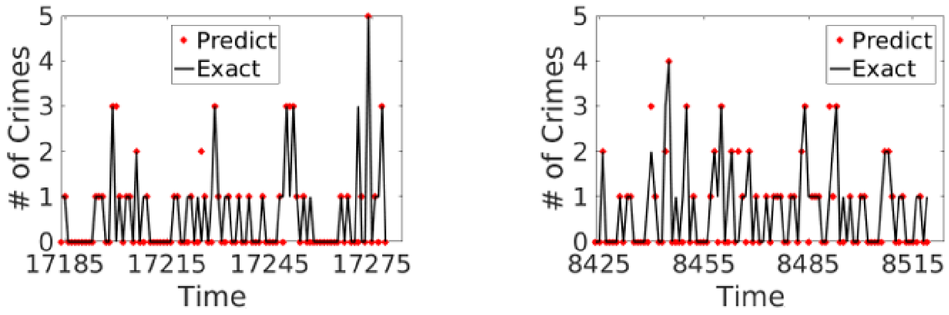


Figure 6.4: Inferred versus exact hourly crime rates for Chicago (left) and Los Angeles (right) data. The unit of time is in hours.



neuron is the basic building block of the latter, while SRNN is built with LSTMs as basic units. The STWG representation together with the feed-forward arranged LSTMs builds the framework for ST data forecasting. The flowchart of our framework is shown in fig. 6.3.

This approach is applied to hourly crime predictions for events from Chicago and Los Angeles. The results (see [WLZ18] for more details) are very accurate in real-time prediction, outperforming the state-of-the-art methods for spatiotemporal forecasting. Here we show examples of the model predictions in fig. 6.4. There are a few difficulties that require future attention. The first question is how to incorporate more spatial covariates in the multivariate model and DNN. This can be solved by adjusting the current DNN structure. Moreover, our multivariate model represents crime connections as a static graph. A temporal graph that can model the changing mutual influence between neighboring nodes could be incorporated in our framework.

CHAPTER 7

Conclusions

Scalability, generalization, and multivariate modeling play important roles in spatiotemporal data modeling. This thesis focuses on developing such methods for the spatiotemporal point processes, providing both nonparametric and deep learning models for real-world applications.

In chapter 3, we bridged the gap of current research by introducing multivariate spatiotemporal Hawkes processes and apply them to many network-related problems in crime and social media. We first derive the EM-type inference method for the Hawkes process and a cluster-based method for simulation. As a generative model, our model can generate new data similar to existing data for forecasting. The model benefits from using nonparametric modeling and spatial information. As a result, we see our method performs competitively in applications such as network reconstruction and declustering for both synthetic and real-world data. We point out its computational drawbacks. Specifically, we note that the computational complexity of the EM algorithm is at least quadratic due to the calculation of log-likelihood.

In chapter 4, we focus on improving the scalability for the point-process model. We improve its computational capability to linear time, making it possible to apply it to millions of events. The key idea involves deriving an analytic formula for the likelihood estimator combined with the moment-based method for productivity and background rate estimation. This approach ameliorates the scale-limitation of our model, leading to substantial improvements of point-process models on social network reconstruction. However, large-scale data often contains complicated dynamics over space and time that the simple linear Hawkes process may not have enough model capacity to capture. This could hinder the spatiotemporal model's generalization ability under certain circumstances.

In chapter 5, we propose the use of deep generative models such as variational autoencoders to improve its generalization ability when there are complex dynamics within point process data. The key theorem here proposes an intensity-free inference for point process data, which simply uses a density estimator for the estimation of point-process models. With the rich literature in density estimation, we adopt the concept of amortized inference and propose an efficient model for multivariate spatial point processes using variational autoencoders. Our approach greatly outperforms the classic point process methods when generalizing on unseen data. The improved performance allows us to apply point-process models to recommendation systems. We then show how our approach relates to another popular collaborative filtering approach.

In addition to creating more general, scalable and multivariate spatiotemporal point-process models, we also applied these models to real-world problems. In chapter 6, we demonstrate the use of the multivariate model on predicting opioid overdoses and the combination of fast network reconstruction with graph-structured recurrent neural networks for real-time crime forecasting. Besides these applications, we claim our models have potential uses for other fields. For example, our model is directly related the network reconstruction and causal inference. Another application is recommendation systems, where point-process models can be used to model human dynamics.

There are many interesting directions to extend our methods. For example, the current regularization method can be extended to a more general case to utilize the smoothness properties of triggering densities. Another direction considers extending the density estimation-based method in chapter 5 to more point-process models, which could provide a better alternative to deep point-process models comparing to the conditional intensity function. Our approaches are not limited to the Euclidean distance (for the spatial variables) that has been used commonly in seismology [Oga98] and crime applications [MSB11]. In other words, although the spatial triggering kernel is a function of Euclidean distance, one can potentially use embeddings, such as those from graph neural networks in chapter 5, or any notion of "distance" between two entities. For example, in a network, one can measure a distance between two entities based on the length of the shortest path between them. In a recent paper, Green et al. [GHP17] proposed a social-contagion model in which they assumed, using a parametric form, that the strength of triggering in a Hawkes-process model depends on the shortest-path distance. With our nonparametric approach in chapter 3, we can nonparametrically estimate such dependence. Another example considers point processes where each event is associated with textual information. For instance, in a Twitter data set, one can consider each tweet (a time-stamped body of text) as an event in a point process. One can measure a distance between two tweets based on their text. It will be interesting to apply our nonparametric approach to these and other applications.

REFERENCES

[ABG17] Massil Achab, Emmanuel Bacry, Stéphane Gaïffas, Iacopo Mastromatteo, and Jean-François Muzy. "Uncovering causality from multivariate Hawkes integrated cumulants." *The Journal of Machine Learning Research*, **18**(1):6998–7025, 2017.

[AJC14] Christopher Aicher, Abigail Z Jacobs, and Aaron Clauset. "Learning latent block structure in weighted networks." *Journal of Complex Networks*, **3**(2):221–248, 2014.

[AMF12] Leman Akoglu, Pedro OS Vaz de Melo, and Christos Faloutsos. "Quantifying reciprocity in large weighted communication networks." In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 85–96. Springer, 2012.

[AMM09] Ryan Prescott Adams, Iain Murray, and David JC MacKay. "Tractable nonparametric Bayesian inference in Poisson processes with Gaussian process intensities." In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 9–16. ACM, 2009.

[Bar18] Marc Barthelemy. *Morphogenesis of spatial networks.* Springer International Publishing, Cham, 2018.

[BBG17] Emmanuel Bacry, Martin Bompaire, Stéphane Gaïffas, and Soren Poulsen. "Tick: a Python library for statistical learning, with a particular emphasis on time-dependent modelling." *arXiv preprint arXiv:1707.03003*, 2017.

[BBG18] Hugo Barbosa, Marc Barthelemy, Gourab Ghoshal, Charlotte R James, Maxime Lenormand, Thomas Louail, Ronaldo Menezes, José J Ramasco, Filippo Simini, and Marcello Tomasini. "Human mobility: Models and applications." *Physics Reports*, **734**:1–74, 2018.

[BCG09] Duygu Balcan, Vittoria Colizza, Bruno Gonçalves, Hao Hu, José J Ramasco, and Alessandro Vespignani. "Multiscale mobility networks and the spatial spreading of infectious diseases." *Proceedings of the National Academy of Sciences of the United States of America*, **106**(51):21484–21489, 2009.

[BHG06] Dirk Brockmann, Lars Hufnagel, and Theo Geisel. "The scaling laws of human travel." *Nature*, **439**(7075):462–465, 2006.

[BKM04] Emery N Brown, Robert E Kass, and Partha P Mitra. "Multiple neural spike train data analysis: state-of-the-art and future challenges." *Nature neuroscience*, **7**(5):456, 2004.

[BM16] Emmanuel Bacry and Jean-François Muzy. "First-and second-order statistics characterization of Hawkes processes and non-parametric estimation." *IEEE Transactions on Information Theory*, **62**(4):2184–2202, 2016.

[BMM15] Emmanuel Bacry, Iacopo Mastromatteo, and Jean-François Muzy. "Hawkes processes in finance." *Market Microstructure and Liquidity*, **1**(01):1550005, 2015.

[BSM12] Earvin Balderama, Frederic P Schoenberg, Erin Murray, and Philip W Rundel. "Application of branching models in the study of invasive species." *Journal of the American Statistical Association*, **107**(498):467–476, 2012.

[BSY17] P Jeffery Brantingham, Nick Sundback, Baichuan Yuan, and Kristine Chan. "GRYD intervention incident response & gang crime 2017 evaluation report.", 2017.

[But08] Carter T Butts. "A relational event framework for social action." *Sociological Methodology*, **38**(1):155–200, 2008.

[BYS18] P Jeffrey Brantingham, Baichuan Yuan, Nick Sundback, Frederick P Schoenberg, Andrea L Bertozzi, Joshua Gordon, Jorja Leap, Kristine Chan, Molly Kraus, and Sean Malinowski. "Does violence interruption work?" 2018.

[CCC19] Achmad Choiruddin, Francisco Cuevas-Pacheco, Jean-François Coeurjolly, and Rasmus Waagepetersen. "Regularized estimation for highly multivariate log Gaussian Cox processes." *arXiv preprint arXiv:1905.01455*, 2019.

[CDF13] Michael D Conover, Clayton Davis, Emilio Ferrara, Karissa McKelvey, Filippo Menczer, and Alessandro Flammini. "The geospatial characteristics of a social movement communication network." *PloS ONE*, **8**(3):e55957, 2013.

[CGB14] Yoon-Sik Cho, Aram Galstyan, P Jeffrey Brantingham, and George Tita. "Latent self-exciting point process model for spatial-temporal networks." *Discrete & Continuous Dynamical Systems-B*, **19**(5):1335–1354, 2014.

[CML11] Eunjoon Cho, Seth A Myers, and Jure Leskovec. "Friendship and mobility: user movement in location-based social networks." In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1082–1090. ACM, 2011.

[CMR19] Jeremy G Carter, George O Mohler, and Bradley Ray. "Spatial concentration of opioid overdose deaths in Indianapolis: An application of the law of crime concentration at place to a public health epidemic." *Journal of contemporary criminal justice*, **35**(2):161–185, 2019.

[Cre15] Noel Cressie. *Statistics for spatial data.* John Wiley & Sons, New York, 2015.

[CSS17] Shizhe Chen, Ali Shojaie, Eric Shea-Brown, and Daniela Witten. "The multivariate Hawkes process in high dimensions: beyond mutual excitation." *arXiv preprint arXiv:1707.04928*, 2017.

[CYL19] Wen-Hao Chiang, Baichuan Yuan, Hao Li, Bao Wang, Andrea L Bertozzi, Jeremy Carter, Brad Ray, and George O Mohler. "SOS-EW: System for overdose spike early warning using drug mover's distance-based Hawkes processes." 2019.

[DCJ19] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. "Are we really making much progress? A worrying analysis of recent neural recommendation approaches." In *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 101–109. ACM, 2019.

[DFA15] Nan Du, Mehrdad Farajtabar, Amr Ahmed, Alexander J Smola, and Le Song. "Dirichlet-Hawkes processes with applications to clustering continuous-time document streams." In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 219–228. ACM, 2015.

[DGQ18] Yoni Dukler, Yurun Ge, Yizhou Qian, Shintaro Yamamoto, Baichuan Yuan, Long Zhao, Andrea L Bertozzi, Blake Hunter, Rafael Llerena, and Jesse T Yen. "Automatic valve segmentation in cardiac ultrasound time series data." In *Medical Imaging 2018: Image Processing*, volume 10574, p. 105741Y. International Society for Optics and Photonics, 2018.

[DHS11] John Duchi, Elad Hazan, and Yoram Singer. "Adaptive subgradient methods for online learning and stochastic optimization." *Journal of Machine Learning Research*, **12**(Jul):2121–2159, 2011.

[Dig83] Peter J Diggle. *Statistical analysis of spatial point patterns.* Academic press, Cambridge, 1983.

[Dig85] Peter Diggle. "A kernel method for smoothing point process data." *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **34**(2):138–147, 1985.

[DM15] Toby Davies and Elio Marchione. "Event networks and the identification of crime pattern motifs." *PloS ONE*, **10**(11):e0143638, 2015.

[DV07] Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: Volume II: General theory and structure.* Springer Science & Business Media, New York, 2007.

[EDD17] Michael Eichler, Rainer Dahlhaus, and Johannes Dueck. "Graphical modeling for multivariate Hawkes processes with nonparametric link functions." *Journal of Time Series Analysis*, **38**(2):225–242, 2017.

[FH16] Santo Fortunato and Darko Hric. "Community detection in networks: A user guide." *Physics Reports*, **659**:1–44, 2016.

[FSG16] Eric W Fox, Frederic P Schoenberg, and Joshua S Gordon. "Spatially inhomogeneous background rate estimators and uncertainty quantification for nonparametric Hawkes point process models of earthquake occurrences." *The Annals of Applied Statistics*, **10**(3):1725–1756, 2016.

[FSS16] Eric W Fox, Martin B Short, Frederic P Schoenberg, Kathryn D Coronges, and Andrea L Bertozzi. "Modeling e-mail networks and inferring leadership using self-exciting point processes." *Journal of the American Statistical Association*, **111**(514):564–584, 2016.

[FTS17] Seth Flaxman, Yee Whye Teh, and Dino Sejdinovic. "Poisson intensity estimation with reproducing kernels." *Electronic Journal of Statistics*, **11**(2):5081–5104, 2017.

[FWR15] Mehrdad Farajtabar, Yichen Wang, Manuel Gomez Rodriguez, Shuang Li, Hongyuan Zha, and Le Song. "Coevolve: A joint point process model for information diffusion and network co-evolution." In *Advances in Neural Information Processing Systems*, pp. 1954–1962, 2015.

[GG14] Samuel Gershman and Noah Goodman. "Amortized inference in probabilistic reasoning." In *Proceedings of the annual meeting of the cognitive science society*, volume 36, 2014.

[GHP17] Ben Green, Thibaut Horel, and Andrew V Papachristos. "Modeling contagion through social networks to explain and predict gunshot violence in Chicago, 2006 to 2014." *JAMA Internal Medicine*, **177**(3):326–333, 2017.

[Gra69] Clive W J Granger. "Investigating causal relations by econometric models and cross-spectral methods." *Econometrica: Journal of the Econometric Society*, pp. 424–438, 1969.

[Haw71] Alan G Hawkes. "Spectra of some self-exciting and mutually exciting point processes." *Biometrika*, **58**(1):83–90, 1971.

[HLZ17] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. "Neural collaborative filtering." In *Proceedings of the 26th International Conference on World Wide Web*, pp. 173–182. International World Wide Web Conferences Steering Committee, 2017.

[HO74] Alan G Hawkes and David Oakes. "A cluster process representation of a self-exciting process." *Journal of Applied Probability*, **11**(3):493–503, 1974.

[Hol15] Petter Holme. "Modern temporal network theory: a colloquium." *The European Physical Journal B*, **88**(9):234, 2015.

[HR12] Cory P. Haberman and Jerry H. Ratcliffe. "The predictive policing challenges of near repeat armed street robberies." *Policing: A Journal of Policy and Practice*, **6**(2):151–166, 2012.

[HW16] Eric C Hall and Rebecca M Willett. "Tracking dynamic point processes on networks." *IEEE Transactions on Information Theory*, **62**(7):4327–4346, 2016.

[HYL17] Will Hamilton, Zhitao Ying, and Jure Leskovec. "Inductive representation learning on large graphs." In *Advances in Neural Information Processing Systems*, pp. 1024–1034, 2017.

[JBJ19] Lucas GS Jeub, Marya Bazzi, Inderjit S. Jutla, and Peter J. Mucha. "A generalized Louvain method for community detection implemented in Matlab." 2011–2019. Version 2.1.

[JGM15] Abdollah Jalilian, Yongtao Guan, Jorge Mateu, and Rasmus Waagepetersen. "Multivariate product-shot-noise Cox point process models." *Biometrics*, **71**(4):1022–1033, 2015.

[JZS16] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. "Structural-RNN: Deep learning on spatio-temporal graphs." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5308–5317, 2016.

[KB15] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization." In *International Conference on Learning Representations*, 2015.

[KBB17] Da Kuang, P Jeffrey Brantingham, and Andrea L Bertozzi. "Crime topic modeling." *Crime Science*, **6**(1):12, 2017.

[KCP15] Da Kuang, Jaegul Choo, and Haesun Park. "Nonnegative matrix factorization for interactive topic modeling and document clustering." In *Partitional Clustering Algorithms*, pp. 215–243. Springer, 2015.

[KDP12] Da Kuang, Chris Ding, and Haesun Park. "Symmetric nonnegative matrix factorization for graph clustering." In *Proceedings of the 2012 SIAM International Conference on Data Mining*, pp. 106–117. SIAM, 2012.

[KJK18] Márton Karsai, Hang-Hyun Jo, and Kimmo Kaski. *Bursty human dynamics.* Springer International Publishing, Cham, 2018.

[KK17] Hyeon-Woo Kang and Hang-Bong Kang. "Prediction of crime occurrence from multi-modal data using deep learning." *PloS one*, **12**(4):e0176244, 2017.

[KP15] Mikko Kivelä and Mason A Porter. "Estimating interevent time distributions from finite observation periods in communication networks." *Physical Review E*, **92**(5):052813, 2015.

[KW14] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes." In *International Conference on Learning Representations*, 2014.

[LA14] Scott Linderman and Ryan Adams. "Discovering latent network structure in point process data." In *International Conference on Machine Learning*, pp. 1413–1421, 2014.

[Lau96] Steffen L Lauritzen. *Graphical models*, volume 17. Clarendon Press, Oxford, 1996.

[LGO15] Chris Lloyd, Tom Gunter, Michael Osborne, and Stephen Roberts. "Variational inference for Gaussian process modulated Poisson processes." In *International Conference on Machine Learning*, pp. 1814–1822, 2015.

[LHR15] Wenzhao Lian, Ricardo Henao, Vinayak Rao, Joseph Lucas, and Lawrence Carin. "A multitask point process predictive model." In *International Conference on Machine Learning*, pp. 2030–2038, 2015.

[LKH18] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. "Variational autoencoders for collaborative filtering." In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pp. 689–698. International World Wide Web Conferences Steering Committee, 2018.

[LM11] Erik Lewis and George O Mohler. "A nonparametric EM algorithm for multiscale Hawkes processes." *Journal of Nonparametric Statistics*, **1**(1):1–20, 2011.

[LMY16] Eric L Lai, Daniel Moyer, Baichuan Yuan, Eric Fox, Blake Hunter, Andrea L Bertozzi, and P Jeffrey Brantingham. "Topic time series analysis of microblogs." *IMA Journal of Applied Mathematics*, **81**(3):409–431, 2016.

[LPA09] David Lazer, Alex Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, et al. "Computational social science." *Science*, **323**(5915):721–723, 2009.

[LS99] Daniel D Lee and H Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization." *Nature*, **401**(6755):788, 1999.

[LZO10] Yifei Lou, Xiaoqun Zhang, Stanley Osher, and Andrea Bertozzi. "Image recovery via nonlocal operators." *Journal of Scientific Computing*, **42**(2):185–197, 2010.

[MA03] Shmoolik Mangan and Uri Alon. "Structure and function of the feed-forward loop network motif." *Proceedings of the National Academy of Sciences of the United States of America*, **100**(21):11980–11985, 2003.

[MAA18] Pablo Moreno-Muñoz, Antonio Artés, and Mauricio Álvarez. "Heterogeneous multi-output Gaussian process prediction." In *Advances in Neural Information Processing Systems*, pp. 6711–6720, 2018.

[ML08] David Marsan and Olivier Lengline. "Extending earthquakes' reach through cascading." *Science*, **319**(5866):1076–1079, 2008.

[Moh14] George O Mohler. "Marked point process hotspot maps for homicide and gun crime prediction in Chicago." *International Journal of Forecasting*, **30**(3):491–497, 2014.

[MRM10] Peter J Mucha, Thomas Richardson, Kevin Macon, Mason A Porter, and Jukka-Pekka Onnela. "Community structure in time-dependent, multiscale, and multiplex networks." *Science*, **328**(5980):876–878, 2010.

[MRW18] Benjamin Mark, Garvesh Raskutti, and Rebecca Willett. "Network Estimation from Point Process Data." *arXiv preprint arXiv:1802.04838*, 2018.

[MSB11] George O Mohler, Martin B Short, P Jeffrey Brantingham, Frederic P Schoenberg, and George E Tita. "Self-exciting point process modeling of crime." *Journal of the American Statistical Association*, **106**(493):100–108, 2011.

[MSI02] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. "Network motifs: simple building blocks of complex networks." *Science*, **298**(5594):824–827, 2002.

[MSM15] George O Mohler, Martin B Short, Sean Malinowski, Mark Johnson, George E Tita, Andrea L Bertozzi, and P Jeffrey Brantingham. "Randomized controlled field trials of predictive policing." *Journal of the American Statistical Association*, **110**(512):1399–1411, 2015.

[Neu98] Arnold Neumaier. "Solving ill-conditioned and singular linear systems: A tutorial on regularization." *SIAM Review*, **40**(3):636–666, 1998.

[New06] Mark E J Newman. "Finding community structure in networks using the eigenvectors of matrices." *Physical Review E*, **74**(3):036104, 2006.

[New18] Mark E J Newman. *Networks*. Oxford university press, Oxford, 2018.

[NG04] Mark E J Newman and Michelle Girvan. "Finding and evaluating community structure in networks." *Physical Review E*, **69**(2):026113, 2004.

[NSL12] Anastasios Noulas, Salvatore Scellato, Renaud Lambiotte, Massimiliano Pontil, and Cecilia Mascolo. "A tale of many cities: Universal patterns in human urban mobility." *PloS ONE*, **7**(5):e37027, 2012.

[NW72] John Ashworth Nelder and Robert WM Wedderburn. "Generalized linear models." *Journal of the Royal Statistical Society: Series A (General)*, **135**(3):370–384, 1972.

[Oga78] Yoshiko Ogata. "The asymptotic behaviour of maximum likelihood estimators for stationary point processes." *Annals of the Institute of Statistical Mathematics*, **30**(1):243–261, 1978.

[Oga88] Yosihiko Ogata. "Statistical models for earthquake occurrences and residual analysis for point processes." *Journal of the American Statistical association*, **83**(401):9–27, 1988.

[Oga98] Yosihiko Ogata. "Space-time point-process models for earthquake occurrences." *Annals of the Institute of Statistical Mathematics*, **50**(2):379–402, 1998.

[Pap09] Andrew V. Papachristos. "Murder by structure: Dominance relations and the social structure of gang homicide." *American Journal of Sociology*, **115**(1):74–128, 2009.

[Pei19] Tiago P Peixoto. "Bayesian stochastic blockmodeling." *Advances in network clustering and blockmodeling*, pp. 289–332, 2019.

[PG16] Mason A Porter and James Gleeson. *Dynamical systems on networks: A tutorial*, volume 4 of *Frontiers in Applied Dynamical Systems: Reviews and Tutorials*. Springer, Cham, 2016.

[PH17] Mason A Porter and Sam D Howison. "The Role of Network Analysis in Industrial and Applied Mathematics." *arXiv preprint arXiv:1703.06843*, 2017.

[POM09] Mason A Porter, Jukka-Pekka Onnela, and Peter J. Mucha. "Communities in networks." *Notices of the AMS*, **56**(9):1082–1097, 1164–1166, 2009.

[PW12] Michael D Porter, Gentry White, et al. "Self-exciting hurdle models for terrorist activity." *The Annals of Applied Statistics*, **6**(1):106–124, 2012.

[PW13] Patrick O Perry and Patrick J Wolfe. "Point process modelling for directed interaction networks." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **75**(5):821–849, 2013.

[Rei18a] Alex Reinhart. "Rejoinder: A review of self-exciting spatio-temporal point processes and their applications." *Statistical Science*, **33**(3):330–333, 2018.

[Rei18b] Alex Reinhart. "A review of self-exciting spatio-temporal point processes and their applications." *Statistical Science*, **33**(3):299–318, 2018.

[RG18] Alex Reinhart and Joel Greenhouse. "Self-exciting point processes with spatial covariates: modelling the dynamics of crime." *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **67**(5):1305–1329, 2018.

[SBB10] Martin B Short, P Jeffrey Brantingham, Andrea L Bertozzi, and George E Tita. "Dissipation and displacement of hotspots in reaction–diffusion models of crime." *Proceedings of the National Academy of Sciences of the United States of America*, **107**(9):3961–3965, 2010.

[SBG14] Frederic P Schoenberg, David R Brillinger, and Peter Guttorp. "Point processes, spatial-temporal." *Wiley StatsRef: Statistics Reference Online*, 2014.

[Sch13] Frederic P Schoenberg. "Facilitated estimation of ETAS." *Bulletin of the Seismological Society of America*, **103**(1):601–605, 2013.

[Sch18] Frederic P Schoenberg. "Comment on "A review of self-exciting spatio-temporal point processes and their applications" by Alex Reinhart." *Statistical Science*, **33**(3):325–326, 2018.

[SG02] Alexander Strehl and Joydeep Ghosh. "Cluster ensembles—a knowledge reuse framework for combining multiple partitions." *Journal of Machine Learning Research*, **3**(Dec):583–617, 2002.

[SGH18] Frederic P Schoenberg, Joshua Seth Gordon, and Ryan J Harrigan. "Analytic computation of nonparametric Marsan–Lengliné estimates for Hawkes point processes." *Journal of Nonparametric Statistics*, **30**(3):742–757, 2018.

[SJ10] Aleksandr Simma and Michael I Jordan. "Modeling events with cascades of Poisson processes." In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, pp. 546–555. AUAI Press, 2010.

[SNM11] Salvatore Scellato, Anastasios Noulas, and Cecilia Mascolo. "Exploiting place features in link prediction on location-based social networks." In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1046–1054. ACM, 2011.

[SPR13] Tiziano Squartini, Francesco Picciolo, Franco Ruzzenenti, and Diego Garlaschelli. "Reciprocity of weighted networks." *Scientific Reports*, **3**:2729, 2013.

[SSB11] Alexey Stomakhin, Martin B Short, and Andrea L Bertozzi. "Reconstruction of missing data in social networks based on temporal patterns of interactions." *Inverse Problems*, **27**(11):115013, 2011.

[SU09] Didier Sornette and S Utkin. "Limits of declustering methods for disentangling exogenous from endogenous events in time series with foreshocks, main shocks, and aftershocks." *Physical Review E*, **79**(6):061110, 2009.

[TDR15] Benjamin Taylor, Tilman Davies, Barry Rowlingson, and Peter Diggle. "Bayesian inference and data augmentation schemes for spatial, spatiotemporal and multivariate log-Gaussian Cox processes in R." *Journal of Statistical Software*, **63**:1–48, 2015.

[TKM11] Amanda L Traud, Eric D Kelsic, Peter J Mucha, and Mason A Porter. "Comparing community structure to characteristics in online collegiate social networks." *SIAM Review*, **53**(3):526–543, 2011.

[UKB11] Johan Ugander, Brian Karrer, Lars Backstrom, and Cameron Marlow. "The anatomy of the Facebook social graph." *arXiv preprint arXiv:1111.4503*, 2011.

[VS08] Alejandro Veen and Frederic P Schoenberg. "Estimation of space–time branching process models in seismology using an EM-type algorithm." *Journal of the American Statistical Association*, **103**(482):614–624, 2008.

[Wan18] Huanchen Wang. *Assessing the impact of interventions on retaliatory violent crimes using Hawkes models with covariates.* PhD thesis, UCLA, 2018.

[WB12] Xiaofeng Wang and Donald E. Brown. "The spatio-temporal modeling for criminal incidents." *Security Informatics*, **1**(1):2, 2012.

[WLZ18] Bao Wang, Xiyang Luo, Fangbo Zhang, Baichuan Yuan, Andrea L Bertozzi, and P Jeffrey Brantingham. "Graph-based deep modeling and real time forecasting of sparse spatio-temporal data." *arXiv preprint arXiv:1804.00684*, 2018.

[WR06] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT Press Cambridge, MA, 2006.

[WYB19] Bao Wang, Penghang Yin, Andrea Louise Bertozzi, P Jeffrey Brantingham, Stanley Joel Osher, and Jack Xin. "Deep learning for real-time crime forecasting and its ternarization." *Chinese Annals of Mathematics, Series B*, **40**(6):949–966, 2019.

[YCI17] Baichuan Yuan, Sathya R Chitturi, Geoffrey Iyer, Nuoyu Li, Xiaochuan Xu, Ruohan Zhan, Rafael Llerena, Jesse T Yen, and Andrea L Bertozzi. “Machine learning for cardiac ultrasound time series data.” In *Medical Imaging 2017: Biomedical Applications in Molecular, Structural, and Functional Imaging*, volume 10137, p. 101372D. International Society for Optics and Photonics, 2017.

[YLB19] Baichuan Yuan, Hao Li, Andrea L Bertozzi, P Jeffrey Brantingham, and Mason A Porter. “Multivariate Spatiotemporal Hawkes Processes and Network Reconstruction.” *SIAM Journal on Mathematics of Data Science*, **1**(2):356–382, 2019.

[YSB19] Baichuan Yuan, Frederic P Schoenberg, and Andrea L Bertozzi. “Fast estimation of multivariate spatiotemporal Hawkes processes and network reconstruction.” 2019. submitted.

[YTM19] Baichuan Yuan, Yen Joe Tan, Maruti K Mudunuru, Omar E Marcillo, Andrew A Delorey, Peter M Roberts, Jeremy D Webster, Christine NL Gammans, Satish Karra, George D Guthrie, et al. “Using machine learning to discern eruption in noisy environments: A case study using CO2-driven cold-water geyser in Chimayó, New Mexico.” *Seismological Research Letters*, **90**(2A):591–603, 2019.

[YWM20] Baichuan Yuan, Xiaowei Wang, Jianxin Ma, Chang Zhou, Andrea L. Bertozzi, and Hongxia Yang. “Variational Autoencoders for Highly Multivariate Spatial Point Processes Intensities.” In *International Conference on Learning Representations*, 2020.

[ZOV02] Jiancang Zhuang, Yosihiko Ogata, and David Vere-Jones. “Stochastic declustering of space-time earthquake occurrences.” *Journal of the American Statistical Association*, **97**(458):369–380, 2002.

[ZOV04] Jiancang Zhuang, Yosihiko Ogata, and David Vere-Jones. “Analyzing earthquake clustering features by using stochastic reconstruction.” *Journal of Geophysical Research: Solid Earth*, **109**(B5), 2004.