UNIVERSITY OF CALIFORNIA

Los Angeles

Causality in Point Processes

A dissertation submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Statistics

by

Ian Richard McGovern

2024

ABSTRACT OF THE DISSERTATION

Causality in Point Processes

by

Ian Richard McGovern
Doctor of Philosophy in Statistics
University of California, Los Angeles, 2024
Professor Frederic R. Paik Schoenberg, Chair

This dissertation discusses observing causality within point processes. Firstly, a novel hypothesis test for distinguishing between inhomogeneity and causal clustering is described, validated on simulations, and then used in shooting data. Next, the hypothesis test is also applied to temporal disease data to identify social contagion mechanisms. Lastly, a discussion of a novel potential outcomes framework as applied to point process data is defined and applied to simulated data.

The dissertation of Ian Richard McGovern is approved.

Paul Jeffrey Brattingham

Yingnian Wu

Chad J. Hazlett

Frederic R. Paik Schoenberg, Committee Chair

University of California, Los Angeles

2024

Dedicated to Mom, Dad, Craig, Kyle, and Derek(the McGovern Family)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Dr. Frederic Schoenberg-Thank you so much for guiding me through my academic journey at UCLA. I wouldn't have made it through without your knowledge, experience, and kindness.

Thank you to Project Tycho, CDC Wonder, and Paul Jeffrey Brattingham for the data used in this dissertation.

VITA

2015–2019	BA Pure Mathematics – University of California, Santa Cruz
2022–2022	Data Science Intern – Twitter
2023 – 2024	Statistical Intern – Lawrence Livermore National Laboratory.
2019–	Ph.D. Student, Statistics, University of California, Los Angeles. Advisor: Frederic Schoenberg.
2020–2024	Teaching Assistant University of California, Los Angeles.

CHAPTER 1

Introduction and Preliminary Discussion

1.1 Introduction

This dissertation is, broadly, a discussion on looking at causality in the process of the point process framework. Point processes, which will be defined more formally, are essentially a collection of points that are located on some mathematical space. Primarily, the focus of this work is on point processes that occur in either spatio-temporal or temporal spaces.The introduction explores the basic definitions and types of point processes that are relevant to this dissertation.

Chapter 2 and 3 primarily focus on a specific problem, which is distinguishing inhomogeneity from causal clustering. The basic issue of these chapters is looking at data in which the mechanics of the function are unknown and attempting to decide if points are truly "triggering" other points, meaning increasing the likelihood of points occurring in the future, or if the points are simply clustered together in some way due to inhomogeneity. The study of this arose from an observation that many researchers were fitting models that involved triggering to data, and when the data fit well enough to it, deciding that there was truly triggering occurring. This is a dangerous conclusion to make-it is akin to observing a correlation and then deciding that this is a causal relationship. For example, it makes sense to fit disease data to triggering functions-usually, it is known that infections truly trigger other infections if the disease is contagious. However, many of the triggering models typically used in point process literature are flexible enough that if one were to, say, fit a model involving triggering

to parking ticket data in Los Angeles, you would observe that there is in fact triggering occurring. In reality, it would be clear to any observer that parking tickets would just be clustered during certain times(Friday evenings for example) in certain places(Downtown, or places with limited available parking).

The issue comes when the amount of triggering, if there is any, is unknown to the researcher. An example would be in crime data-there has been a long history(which will be explored further in Chapter 2) of individuals fitting triggering models to crime data, and when they fit well, creating a theory of crime that crime can truly "trigger" other crime to a significant extent. Another example involves any sort of social contagion mechanism. Again, while social contagion is a possible explanation, it is also very possible that there has simply inhomogeneity in the data.

It is necessary, therefore, to develop a clear method for dealing with situations when the triggering method is unknown, and to provide some sort of guardrails or alternate explanations for any clustering behavior that does not lead immediately to assumptions of causal triggering within the data. Chapter 2 will introduce and explore a proposed hypothesis test to do so, and explore crime data with this hypothesis test. In Chapter 2, the hypothesis test introduced is found to be accurate at distinguishing between clustering and causal clustering models, with relatively high power when compared to previous methods that have been used for similar problems.

Chapter 3 explores using this method on temporal, instead of spatio-temporal, data, which is the type of data that was used for Chapter 2. Specifically, the type of data that is explored is temporal data that has been binned into weekly counts. The focus of this chapter is on disease data. In order to validate this method, both contagious and non-contagious infectious diseases are studied. Measles and Chlamydia, which are both contagious diseases, are studied to make sure that the hypothesis test is correctly able to identify contagious diseases. Lyme disease, which has clustering properties due to its transmission that occurs through ticks, is also studied to make sure that this method works correctly on diseases that

are clustered but not contagious. Finally, adolescent suicide data is studied to see if there is a measurable social contagion within the suicide data of adolescents. The results find that the test is able to correctly distinguish between contagious and non-contagious diseases, and that the suicide data shows no evidence of clustering.

Chapter 4 explores a more formal investigation in how causality can be described with point process data. Taking inspiration from the potential outcomes framework, a framework for describing the causal effect of a point processes is explored. Specifically, models in which there is triggering that can cause spill-over effects are explored, and a method is identified to allow for an accurate representation of the causal effect in this situation. These methods are explored through simulations, with both non-triggering and triggering models explored. In addition, a method to allow for covariates to be included in the analysis is also discussed, with simulations also performed. The simulations show that this method seems to have both unbiased as well as consistent results, with the estimated causal effects in simulations being centered around 0, as well as the amount of error decreasing as the amount of data increases.

1.2 Point Process Definition

A point process X on $\mathbb{R}^d$ is typically defined as a random set of points such that for any $B \in \mathbb{R}^d$ such that B is a bounded Borel set, the number of points that are in B, denoted typically as $N(B)$, is a finite number(van Lieshout, 2019). This dissertation will primarily focus on the case of a spatio-temporal Hawkes process, meaning that the point process is occurring on the product of some spatial space $\mathcal{S} \in \mathbb{R}^2$ and some temporal space $T \in \mathbb{R}^+$.

A more technical explanation can focus on a framework involving integer valued measures and σ-algebras. A σ-algebra of a set X is defined as a collection Σ of subsets of X such that:Σ contains X,Σ is closed under complement, Σ is closed under countable unions andΣ is closed under countable intersections. A Borel Set is a set that can be obtained by taking a countable number of operations from a collection of open sets, with the operations consisting

of unions, intersections, and complements(Srivastava, 1998). Let $\mathcal{X}$ be a complete separable measure space, and let $\mathcal{B}_{\mathcal{X}}$ be the σ-field of the Borel sets of $\mathcal{X}$. This means that $\mathcal{B}_x$ is a collection of subsets of X that contain Borel sets as defined previously.

It is necessary to ensure that we do not have an infinite accumulation of points in a certain area. Specifically, it is necessary for any bounded Borel set in the space to not contain an infinite amount of points. Let $\mathcal{N}_{\mathcal{X}}^{\#}$ represent the space of all boundedly-finite measures on the σ-algebra of Borel sets of $\mathcal{X}$, $\mathcal{B}_{\mathcal{X}}$, with integer values. These are also called "counting measures"(Daley and Vere-Jones, 2008).

$\mathcal{N}_{\mathcal{X}}^{\#}$, is now, in fact, a complete separable measure space on its own. Therefore, it is possible to define another σ-algebra over $\mathcal{N}_{\mathcal{X}}^{\#}$. Specifically, define $\mathcal{B}(\mathcal{N}_{\mathcal{X}}^{\#})$ as the Borel σ-algebra over $\mathcal{N}_{\mathcal{X}}^{\#}$. This will be the smallest σ-algebra over $\mathcal{N}_{\mathcal{X}}^{\#}$ such that the mappings of $N \to N(A)$ are measurable for all $A \in \mathcal{B}_{\mathcal{X}}$, the σ-algebra of Borel sets of $\mathcal{X}$Daley and Vere-Jones (2008).

This allows for the formal definition of a point process, which is a mapping from a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ to $(\mathcal{N}_{\mathcal{X}}, \mathcal{B}(\mathcal{N}_{\mathcal{X}}))$.

1.3 Conditional Intensity

Point processes are, in most cases, simply defined by the conditional intensity of the process. The notation in the following paragraphs will involve spatio-temporal processes, however these can be defined for any space that the process is occurring on. The conditional intensity is typically denoted as $\lambda(x, y, t|\mathcal{H}_t)$. The "conditional" portion of this intensity is represented by conditioning on $\mathcal{H}_t$, which is the history of the process up until the time t. This conditional intensity is defined as

$$\lambda(x, y, t|\mathcal{H}_t) = \lim_{\delta_x, \delta_y, \delta_t \to 0} \frac{\mathbb{E}[N((x, x+\delta_x) \times (y, y+\delta_y) \times (t, t+\delta_t))]}{\delta_x \delta_y \delta_t}. \tag{1.1}$$

This essentially refers to the expected number of points in a given ϵ-ball around the point (x, y, t) as the ϵ goes to 0-keeping in mind that $N()$ is the number of points in a given region of the point process.

Typically, a point process is solely defined by its conditional intensity and the form that it takes. Different types of point processes are noted when the forms of their conditional intensities follow different structures-for example, you can have a constant conditional intensity, one that is based only on the location and ignores the time, or a variety of types of conditional intensities that depend on the previous history of the process.

1.4 Likelihood of Point Process

Beyond the formal definition of a point process, it is also necessary to rigorously define the likelihood of a point process. This section will specifically cover the case of a finite point process, meaning that there is only a finite number of points. The likelihood of a finite point process can be defined under some conditions. The conditions are as follows(Daley and Vere-Jones, 2003):

1. As previously, let $\mathcal{X}$ represent a complete separable measure space. The point process much take place within this space.

2. The point process has a probability distribution that corresponds to the total number of points that occur within the point process. Let this distribution be denoted as $\{p_n\}$, which is a distribution over the non-negative integers $n = 0, 1, ...$ such that $\sum_{n=0}^{\infty} p_n = 1$. This allows for the number of points in the point process to be a random process, so the total number of points in the process is a random variable.

3. For each non-negative integer m that is in the outcome space of$\{p_n\}$, define a distribution $\Pi_m()$ over the Borel sets of $\mathcal{X}^{(m)} = \mathcal{X} \times ... \times \mathcal{X}$ that determines the joint distribution of the m points in that process, given that there are m points.

his defines a process in which firstly the number of points in the process is randomly chosen by $\{p_n\}$, and then the location of those points is then chosen by choosing a random vector from $\Pi_m()$.

For point processes, the sets of points are assumed to be unordered. This implies that specific points found in certain areas are not relevant(for example, if the points consisted of individuals, the only relevant information would be the location of the individuals, not necessarily which exact individual is in a location). Ordered sets of points would be necessary to display information about one specific individual. Therefore, it is necessary to further specify the distribution Π_m to properly further the goal of point processes dealing with unordered sets. Π_m should not give a different probability, therefore, to any permutation of a set of points occurring in certain locations(as in, the order of the points should not change the probability that the points occur in certain locations). Therefore, the symmetric form of this distribution is defined(Daley and Vere-Jones, 2003).

For any partition $A_1, ..., A_n$ of $\mathcal{X}$, the symmetric distribution, denoted as Π_m^{sym} can be defined asReinhert (2018)

$$\Pi_{sym}(A_1 \times ... \times A_n) = \frac{1}{n!} \sum_{perm} \Pi_n(A_{i_1} \times ... \times A_{i_n})$$

The notation $\sum_{perm}$ is the summation over all permutations of the integers $1, ..., n$, denoted as $i_1, ..., i_n$. Since there are $n!$ such permutations, this sum is divided by $n!$ to allow for this to still be a distribution.

The symmetric distribution can, in turn, be used to define the Janossy measure,J_n. The Janossy measure, J_n for a partition of a complete separable measure space $\mathcal{X}$, denoted as $A_1, ..., A_n$, is defined asReinhert (2018)

$$J_n(A_1 \times \ldots \times A_n) = n! p_n \Pi_n^{sym}(A_1 \times \ldots \times A_n).$$

Recall that p_n is the probability that there are n points in the process, and Π_n^{sym} is a probability measure, set up to be symmetric over the partitions of $\mathcal{X}$, for the joint distribution of the points in the point process.

The Janossy measure is not a probability measure. However, the derivative of this is easily interpretable, which is one reason why the Janossy measure is often used despite it not being a probability density. Suppose the space the point process is occurring in is $\mathbb{R}^d$. Let $j_n(x_1, ..., x_n)$ be the density of the Janossy measure, $J_n()$, with respect to the Lebesgue measure on $(\mathbb{R}^d)^n$. Then $j_n(x_1, ..., x_n)dx_1dx_2...dx_n$ can be interpreted as the probability that there are n points in the process, each occurring within the region of $x_i + dx_i$. Essentially, this is the probability that the point process occurs, with the first stage being the random variable corresponding to the total number of points, and then also specifying that each point occurs in a certain location.

This clearly leads to a way to define the likelihood of a point process under this framework. In order to do so, however, firstly the concept of a local Janossy measure must be defined. Given a bounded Borel set A, the Janossy measures localized to A, denoted as $J_n(|A)$, for $n = 1, 2, ..,$ with locations of $x_1, .., x_n \in A$, can be defined such that $J_n(dx_1, ..., dx_n|A)$ is the probability that there are n points in A, at locations dx_1 through dx_nDaley and Vere-Jones (2003).

Clearly, this simply specifies a Janossy measure over a certain bounded Borel set.This directly allowed for a construction for the likelihood of a point process. The likelihood of a realization of points $x_1, ..., x_n$ of a regular point process N on a bounded Borel set $A \subseteq \mathbb{R}^d$, with $n = N(A)$, is the local Janossy density(Reinhert, 2018)

$$L_A(x_1, ..., x_n) = j_n(x_1, ..., x_n|A)$$

The "regular" point process definition is simply stating that the local Janossy measures behave properly with regards to the Lebesque measure. This definition, while clearly more formal and accurate, is not commonly used in the point process literature. Instead, the likelihood is calculated from the conditional intensity directly, which will be shown later in this introduction.

1.5 Relevant Types of Point Processes

There are three primary types of point processes discussed in this dissertation: Poisson processes, Poisson cluster processes,Neyman-Scott processes, and Hawkes processes. A Poisson process is simply a process in which the conditional intensity is not dependent on $\mathcal{H}_t$, meaning that the intensity at a given spatio-temporal location does not depend on the history of the process. In cases where the intensity is constant throughout the spatial and temporal domain, this process is referred to as a "homogeneous Poisson process," and in cases where this process is not constant throughout the spatial and temporal domain, this is referred to as a "heterogeneous Poisson process." Therefore, the conditional intensity of a Poisson process(λ_P) is of the form

$$\lambda_P(x, y, t|\mathcal{H}_t) = \lambda_P(x, y, t). \tag{1.2}$$

This dissertation, as previously stated, focuses on cluster processes. Before examining specific cluster processes, the idea of a cluster process needs to be more rigidly defined. A cluster process can really be thought of as two separate processes occurring. Firstly, there is a process that determines the "centers" of the clusters, or where the clusters are originating from. This is often called a "center process" or "parent process." Within this paper, the parent process will be denoted as $N_c()$. Sometimes this parent process is included in the final point process, although sometimes this is considered an "invisible" or "latent" process, in which these points are not observed in the defined point process.

Next, once the cluster center points are determined, there needs needs to be a distribution or defined process for the points within a cluster. This process can be much more general than the analysis in this dissertation extends to-it is possible to have each clustering process be unique for each cluster center, or to have this clustering process defined by other covariates. Therefore, it is possible to define a cluster process on a complete separable measure space $\mathcal{X}$. It can be stated that N is a cluster process on $\mathcal{X}$ if, for the center process $N_c()$ occurring in the complete separable measure space $\mathcal{Y}$, with a clustering process of $\{N(|y) : y \in \mathcal{Y}\}$,

that for every bounded $A \in \mathcal{B}_x$,

$$N(A) = \sum_{y_i \in N_c()} N(A|y_i)$$

is finite(Daley and Vere-Jones, 2003). Observe that a clustering process can be thought of as an overlay of each individual clustering distribution, added up over all the points occurring in the parent process.

In general, a Poisson clustering process is any process in which the underlying parent process is a Poisson process, and each of these cluster centers creates some number of points within the cluster according to some distribution(Daley and Vere-Jones, 2008). The number of points triggered and the distribution can be as general as needed. These can be applied to both spatial and spatio-temporal data, or in fact any point process.

A Neyman-Scott process is a certain type of clustering process with a two-part conditional intensity(Neyman and Scott, 1958). This process consists of a hidden "parent" process and an observed "children" or offspring process. The process begins with the parent points being distributed through the domain space-this can be either homogeneous or heterogeneous. Each parent then triggers a random number of offspring points that are observed, with a mean number of points that are produced denoted as A. These offspring points are distributed through some triggering density that is typically done as isotropic for each parent point. These processes are used in a variety of spatial contexts-notably forestry(Penttinen et al., 1992), galaxies(Neyman and Scott, 1958), and rainfall(Guttorp, 1996). The intensity of a Neyman Scott process can be difficult to write out in a condensed form in most cases(Moller and Waagepetersen, 2004), however the parameters can be well estimated by a variety of methods such as maximum likelihood estimation(Baddeley et al., 2022). It is possible to write out an intensity given a σ-algebra $\mathcal{K}$(Zhuang, 2018); however this is not, strictly, a conditional intensity as it does not depends on the history of the process. This intensity

function can be written as

$$\lambda(x, y, t|\mathcal{K}) = A \sum_{(x_i,y_i,t_i)\in N^c} g(x - x_i, y - y_i, t - t_i). \tag{1.3}$$

Hawkes processes are an example of a "self-exciting" point process, and in this dissertation will frequently be referred to as "causal clustering" point processes(Hawkes, 1971b,a). They have been widely used in earthquake analysis and infectious disease analysis(Ogata, 1988; Reinhart, 2018). Both of these can be thought of as events in which there is some event that increases the likelihood of the event occurring in the future. In the case of earthquake analysis, a "main shock" will increase the likelihood of aftershocks of earthquakes occurring nearby in the future. In the infectious disease model, a person being infected with the disease will increase the likelihood of others acquiring the disease through contagion methods.

A Hawkes process, similar to a Neyman Scott process, involves both a parent process and an offspring process. However, unlike a Neyman Scott process, both the parent and offspring process are observed. The parent process consists of a (potentially) inhomogenous Poisson process that is denoted as $\mu(x, y, t)$ for spatiotemporal Hawkes processes. Each parent then triggers an average of κ points according to some triggering density $g()$. The κ value is often referred to as the "productivity" of the Hawkes process. This triggering density can only trigger points forward in time. The value of κ can change based on other variables or change over time-a common example is the ETAS model for earthquakes in which the productivity can change depending on the magnitude of the earthquake(Ogata and Zhuang, 2006).

The conditional intensity for a spatiotemporal Hawkes process is written as

$$\lambda(x, y, t|\mathcal{H}_t) = \mu(x, y, t) + \kappa \sum_{i:t_i<t} g(x - x_i, y - y_i, t - t_i). \tag{1.4}$$

The choice of $g()$ is arbitrary, although typically a parametric distribution is chosen and then the parameters are fit once that choice has been made. As explained later, typically the triggering density is chosen as a distribution for computational reasons.

The likelihood of a point process with conditional intensity $\lambda(x, y, t|\mathcal{H})$ and with observed points $p_i = (x_i, y_i, t_i)$ can be expressed as

$$\prod_{1 \leq i \leq n} \lambda(x_i, y_i, t_i|\mathcal{H}_t) \times \exp\left(- \int_{\mathcal{S}} \lambda(x, y, t) dx dy dt \right) \tag{1.5}$$

such that $\mathcal{S}$ is the observation window of the point process.

CHAPTER 2

Testing for Causal Clustering within Point Processes

2.1 Background and Framing Question

One of the key problems in point process research is distinguishing between inhomogeneity and causal clustering patterns. It is very common for points, in a certain spatio-temporal area, to occur closer together than expected for a random pattern. This results in a common question-is this spatial and/or temporal clustering due to inhomogeneity, in which certain spatiotemporal areas are simply more likely to have points occur, or is there some level of triggering(or causal clustering) occurring in the process. A simple motivating example would be to look at parking tickets in Los Angeles. Plotting these events in a spatio-temporal map would show clear clustering occurring. One might argue that this shows that there is some triggering within the clustering-as in, a parking ticket is "causing", or increasing the likelihood, of a parking ticket to occur near by. A reasonable person would then say that this is clearly the result of inhomogeneity-There is simply more parking tickets occurring in downtown Los Angeles on a Friday due to the increased number of people and the likely increased police presence.

This problem is of such importance that it is been called one of the most important issues in point process analysis(Diggle, 2014). In the case of the parking tickets, it is clear that the underlying mechanism is inhomogeneity, so fitting a causal point process model would not make any sense. However, in cases in which the underlying mechanisms of the process are not known *a priori*, it is possible to make this error without realizing an error is made. For

example, if the disease process of a new illness is not known, a researcher might make the mistake of identifying it as an illness that is contagious due to the clustering of the data in certain spatio-temporal locations. However, it could be possible that the illness only occurs in areas next to certain rivers and it is an illness transmitted through some water-based method.

This chapter will identify a method to try to distinguish between these two cases, where there is the possibility of the observed process resulting from inhomogeneity and the observed process resulting from causal clustering.

2.2 Previous and Related Methods

This section will go over solutions that have been proposed for this problem as well as a look into their limitations as well as their issues with certain applications.

Granger causality is a statistical method that is used for two sets of temporal data sets, originally proposed as part of econometrics research (Granger, 1969a). The method looks at whether it is possible to significantly predict a time series better when another, "causal" time series is also included in the modeling, as opposed to only using the time series itself. Since this is so widely used for "causal" analysis in time series and point process data, it is necessary to more accurately define this method. Let y_t and x_t be two time series. Let $\mathcal{H}_t$ be the history of both x_t and y_t up until time t. Let $\mathcal{H}_{t_x}$ be the history of x_t up until time t. Let $P(x_t)$ be the best prediction for the time series x_t at time t. Then you would state that y_t is "causal" to x_t if

$$var[x_t - P(x_t|\mathcal{H}_{t-1})] < var[x_t - P(x_t|\mathcal{H}_{t-1_{x_t}}].$$

This corresponds to saying that the variance of the error term for predicting x_t with the information included in y_t is less than the variance of the error term for the prediction of x_t only based on x_t(Shojaie and Fox, 2022).

While Granger chose to use the term "causality" for this method, it is not a truly causal method-it really studies a correlational relationship between two time series. This method has been widely applied to point process research(Kim et al., 2011a; Xu et al., 2016), although these studies do not help to answer the question that has been outlined in the introduction of this section. This method can provide valuable insight into which variables to include in prediction models, however it does not answer the question of a true causal relationship. For example, it is possible that looking at a time series of the inflation rate in the United States would "granger cause" a time series that looks at the unemployment rate. While this even could appear to be a reasonable assumption based on economic principles, the unemployment rate could be caused by a variety of other factors that happen to be correlated with the inflation rate.

Another method that was suggested by Diggle was to observe multiple realizations of the point process in order to distinguish between causal clustering and inhomogeneity (Diggle, 2014). If the clusters of points occur in different locations in each realization, then it can be concluded that the clustering is the result of a causal clustering mechanism. If the clustering occurs in the same locations in each realization, then it can be concluded that the clustering occurred due to inhomogeneity. This method is most likely among the best ways to distinguish these two situations when multiple realizations of the data can be observed. However, as in most areas of statistics, seeing multiple realizations of a process is nearly impossible to do-even if the process occurs multiple times, each of those times are by definition observed in a different time window, which makes them not the same process observed multiple times. Therefore, it is necessary to find a method that can be done on a single observed realization of a point process. However, this definition is still a good framing for what the other methods should attempt, in some way, to accomplish if possible.

Cordi proposed a solution that works on only an observed singular realization of the point process(Cordi et al., 2017). This method was primarily tested on temporal data, although a brief exploration into multiple dimensions was explored without the same amount of results

presented. This paper primarily looked at temporal data with an exponential triggering density. The basic outline of this method was to fit a Hawkes model to both the original data, as well as to the "reversed" data-meaning looking at the process that started at the end and went backwards. The reasoning for this was that for a true Hawkes model, having points trigger points backwards in time is not reasonable-for example, this could be thought of as an earthquake causing an aftershock at some point in the past.

To estimate how well the model fit a Hawkes distribution, the compensators, which for a Hawkes process with a triggering density of $g()$ and a background rate of μ is calculated as

$$\Lambda(t_{i-1}, t_i) = \int_{t_{i-1}}^{t} \mu + \sum_{t_k<s} g(s - t_k)ds$$

were calculated. For a Hawkes process, this distribution should follow an exponential distribution with a rate of 1(Papangelou, 1972). A Kolmogorov-Smirov test than can be used to establish how well a temporal dataset will fit to a Hawkes model.

This paper explored a variety of different values for the parameters of the exponential distribution, and it resulted in a low Type I and Type II error. This method is most similar to the method that will be developed and tested in this paper, however the method described in this paper is much more suitable for spatio-temporal data. This method was also not heavily tested in the case of spatially heterogeneous but non-causal data, which this paper also seeks to distinguish from casual clustering more rigorously.

Another method that was previously explored for this purpose was a paper looking at gang-related violence in the Los Angeles area(Park et al., 2021). This method focused on attempting to distinguish between the causal portion of the Hawkes process-meaning the points that are being triggered by other points- from the background rate $\mu(x, y)$, which was assumed to be spatially heterogeneous. This method used generative additive modeling to incorporate as much information as possible that could explain crime rates to isolate the causal clustering that cannot be explained by any background variables.

This method is helpful in that it does allow to incorporate spatial variables into the

analysis, however it has several shortcomings. Notably, this method can only incorporate observable variables, and in fact can only incorporate available and measured variables. This leaves out any variables that were not collected as well as any variables that are unobservable. This limits the applications of this method, as well as the accuracy in identifying true causal clustering. It also assumes that the background rate is being modeled correctly, which is difficult to do. Namely, it becomes an issue of determining if any extra points that are not represented by the background process are evidence of causal clustering or if they are evidence that the background rate does not fit properly.

2.3 Proposed Methods of Distinguishing Inhomogeneity and Causal Clustering

Another possible method to distinguish between causal clustering and heterogeneity is to look at a hypothesis test involving a Poisson cluster process. The basic idea of the method is similar to that of the method involving time reversal, notably that if the clustering of a process is truly causal and not simply a result of inhomogeneity, then a paradigm in which points can be "triggered" backwards should not fit as well as a model going forward. The primary difference between the reversal method and this method is that this method involves a hypothesis test that can be used for both spatial and spatio-temporal processes, which allows for a clear cut-off point for whether or not there is evidence of causal clustering.

2.3.1 Estimation of Parameters and Likelihoods

The first step of the test is to fit a Poisson cluster model and a Hawkes process to the observed data. This can be done by maximum likelihood. For the purpose of the simulations done to demonstrate this technique as well as the example data used for this method, it is assumed that the triggering densities for these distributions are both Gaussian distributions. Specifically, for the Poisson cluster process, it is assumed it is of the form of 1.3 with the

triggering density $g(x - x_i, y - y_i, t - t_i)$ given as

$$g(x-x_i, y-y_i, t-t_i) = (2\pi)^{\frac{3}{2}}\sigma^3 \exp\left(\frac{\frac{1}{2}(x - x_i, y - y_i, t - t_i)(x - x_i, y - y_i, t - t_i)^T}{\sigma^2}\right). \quad (2.1)$$

This is a multivariate normal distribution with a mean of the given parent point as well as a variance that is assumed to be σ^2 times the identity matrix. This assumes that the parent process is both isotropic as well as independent. To fit the values for the maximum likelihood, since this is equivalent to a Gaussian clustering process, a package called 'mclust'(Scrucca et al., 2023) is used, which identifies the best Gaussian clustering process. This involves selecting the number and location of parent points, what the value of σ^2 is, as well as which points are in each cluster. The average number of points per cluster is then used as the A value, and the calculated σ^2 and parent points are used for the estimation.

From the estimation of the A and σ^2 value, as well as the location of the parent points, it is then possible to calculate the likelihood of the Poisson cluster process for use in the hypothesis test. This is done by calculating the intensity at every given point, which is given by

$$\hat{\lambda}(x, y, t) = \hat{A} \sum_{(x_i,y_i,t_i)\in N^c} \hat{g}(x - x_i, y - y_i, t - t_i). \quad (2.2)$$

If the triggering density is chosen such that

$$\int_{\mathcal{S}} g(x - x_i, y - y_i, t - t_i)ds = 1$$

then we can write the second part of the integral as

$$\int_{\mathcal{X}}\int_{\mathcal{Y}}\int_{\mathcal{T}} \sum_{i:(x_i,y_i,t_i)\in N^c} g(x - x_i, y - y_i, t - t_i)dxdydt =$$

$$\sum_{i:(x_i,y_i,t_i)\in N^c} \int_{\mathcal{X}}\int_{\mathcal{Y}}\int_{\mathcal{T}} g(x - x_i, y - y_i, t - t_i)dxdydt = \sum_{i:(x_i,y_i,t_i)\in N^c} (1) = M$$

where M is the number of points in the parent process. Therefore, the integral can be simplified to

$$\int_{\mathcal{X}}\int_{\mathcal{Y}}\int_{\mathcal{T}} A \sum_{i:(x_i,y_i,t_i)\in N^c} g(x - x_i, y - y_i, t - t_i)dxdydt = AM.$$

Without this simplification of the integral, calculating the likelihood of a Poisson cluster point process would be much more computationally intensive. It would be necessary to do a Monte Carlo integration over the entire observation window, which in the case of spatio-temporal data is a three dimensional space with a highly irregular integral. However, this computation allows for much faster calculation of the integral which then allows for a much faster calculation for the hypothesis test.

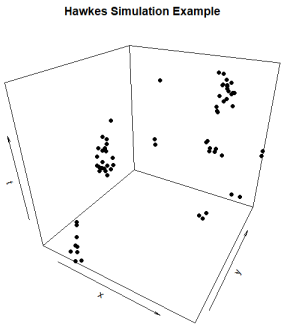
The next step in the process is to fit a Hawkes process to the model. This is done using maximum likelihood estimation. The triggering density in the Hawkes intensity, $h(x - x_i, y - y_i, t - t_i)$ can be any distribution in which the density over $(-\infty, 0]$ is 0, however, to match with the Poisson cluster process in the simulations in this section, the triggering density in the temporal dimension is assumed to be given by a truncated normal distribution with a lower bound of 0 and an upper bound of $+\infty$. The triggering density in the spatial dimensions is a bivariate normal distribution with independent dimensions for latitude and longitude. Therefore, the full triggering density is given as

$$\begin{aligned} h(x - x_i, y - y_i, t - t_i) &= \frac{\phi(\frac{t-t_i}{\sigma})}{-\Phi(0)} 2\pi\sigma_t^2 \exp\left(\frac{\frac{1}{2}(x - x_i, y - y_i)(x - x_i, y - y_i)^T}{\sigma_s^2}\right) \\ &\sim TrN(t, \sigma_t, 0, \infty)N((x, y), \sigma_s^2 I). \end{aligned}$$

For the integral over the observation window of the Hawkes process, it is also possible to simplify the integral for computational purposes. For the Hawkes process, observe that if $h(x - x_i, y - y_i, t - t_i)$ is a density, then

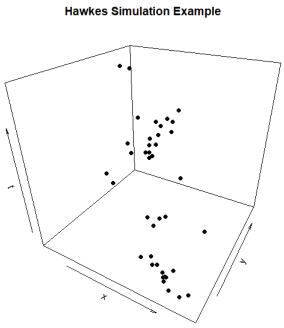
$$\begin{aligned} \int_X \int_Y \int_T \lambda(x, y, t|\mathcal{H}_t) &= \int_X \int_Y \int_T \mu(x, y) + \kappa \sum_{i:t_i<t} h(x - x_i, y - y_i, t - t_i) \\ &= \mu T + \kappa \sum_{i:t_i<t} \int_X \int_Y \int_T h(x - x_i, y - y_i, t - t_i) = \mu T + \kappa M \end{aligned}$$

where T is the size of the temporal domain, and M is the total number of points in the process. Similar to the Poisson cluster estimation, this allows for quick calculation of the integral instead of relying on Monte Carlo methods that, as the number of dimensions increase, get exponentially more computationally intensive.



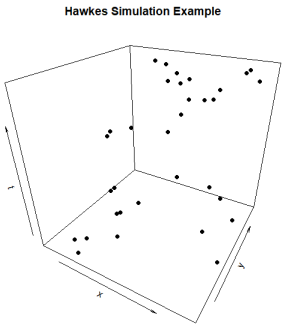
(a)

μ	κ	σ
10	*.95*	*.0005*



(b)

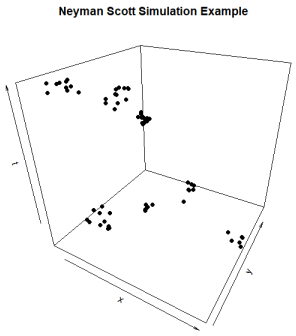
μ	κ	σ
10	*.9*	*.001*



(c)

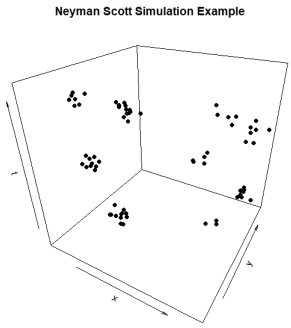
μ	κ	σ
10	*.8*	*.01*

Figure 2.1: *Hawkes Process Examples*



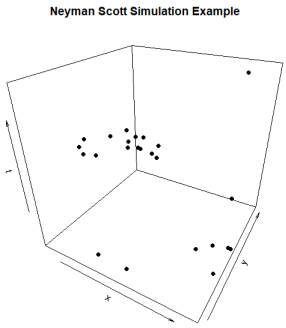
(a)

μ	A	σ
10	*10*	*.001*



(b)

μ	A	σ
10	*5*	*.0005*



(c)

μ	A	σ
10	*3*	*.01*

Figure 2.2: *Poisson Cluster(Neyman Scott) Process Examples*

2.3.2 BIC Method

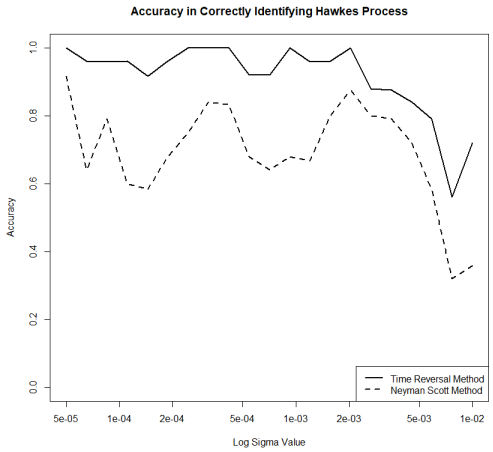
Originally, the Bayesian Information Criterion(BIC) was considered to make a decision of whether the model had evidence of causal clustering. The BIC is a model selection method that is helpful for distinguishing between models with different numbers of parameters(Schwarz, 1978). The BIC is defined as

$$BIC = k\ln(n) - 2l\ln(\hat{L}). \tag{2.3}$$

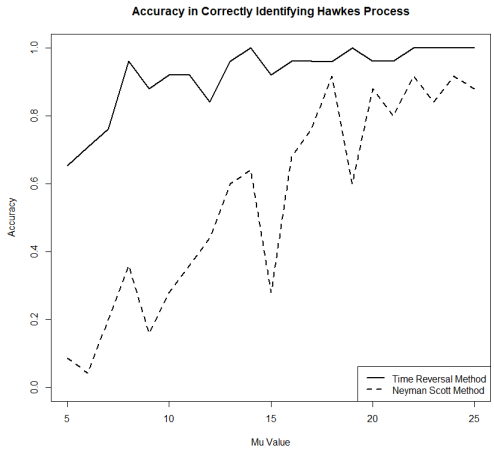
$\hat{L}$ is the estimated likelihood of the model, k is the number of parameters in the model, and n is the number of observed points. The Poisson cluster process, in the way that it was used in this research, has much more parameters than the Hawkes process. For Hawkes, the only parameters that can be changed are κ, μ, and, in the Gaussian triggering densities described previously, σ. The Poisson cluster process has the equivalent parameters of A and σ, however each hidden parent point is also estimated as well. This creates a situation in which the number of parameters for the Poisson cluster process is much higher than the parameters for the Hawkes process, so the Poisson cluster process will nearly always have a higher likelihood than the Hawkes process.

Hawkes processes were simulated to measure the power of this test for the BIC. These simulations occurred in a spatio-temporal domain of a unit cube, consisting of the spatial region with boundaries of $[0,1] \times [0,1]$ and the temporal region with boundaries $[0,1]$. The baseline values for the Hawkes parameters were $\kappa = 6/7$, $\mu = 17$, and $\sigma = .0005$. For each set of parameters where the power was calculated, 25 different Hawkes processes were simulated.

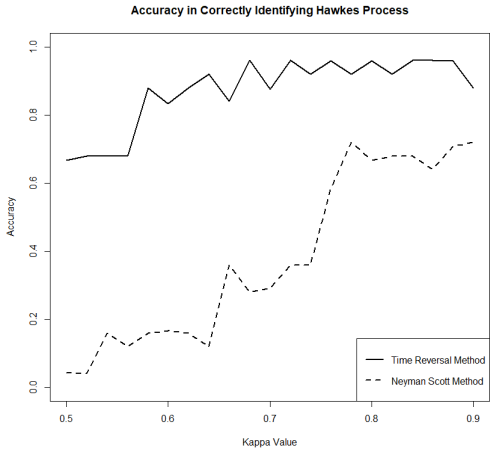
For each simulation in this analysis, three likelihood and BIC values were estimated. First, the likelihood and BIC for the standard Hawkes model was computed using maximum likelihood. In addition, the likelihood and BIC were calculated for the Poisson cluster model. Finally, the temporal dimension was reversed(in a similar process to the Cordi et. al. process). This made the last event the first event and vice versa, as well as swapping all the times so the whole process was backwards. The spatial coordinates, however, were kept



(a) Accuracy for σ Values



(b) Accuracy for μ Values



(c) Accuracy for κ Values

Figure 2.3: *Power of the BIC test over the inputted σ, μ and κ values for a simulated Hawkes process. The dashed lines represent the power for the Poisson Cluster(Neyman Scott) method and the solid lines represent the power for the Time Reversal method*

the same. The likelihood and BIC were then found for the reversed Hawkes process. For each simulation, the accuracy value that is shown refers to the percentage of simulations for which the BIC correctly picked the non-altered Hawkes model as the correct process. This is technically the "power" of this process.

For the first set of simulations, the κ value was fixed at 6/7 and μ was fixed at 17. The values of σ varied from .00005 to .01, with an even spread of 21 values in this range. The results of this accuracy, as shown in Figure 2.3a, shows that there is a clear trend to how accurate the BIC test is on different values of σ. The accuracy generally gets lower as the value of σ increases. This is potentially due to the detection of clustering and fitting the parameters when σ is larger. When σ is larger, the points appear to be much more random, so distinguishing between the causal clustering processes is more difficult. In addition, the time reversal method is universally more accurate than the Poisson cluster method, however the two methods follow a similar pattern in shape.

In addition the μ values were altered. The κ value was fixed at 6/7 and the σ value was fixed at .0005. The μ values were 21 values evenly spaced between 5 and 25. Figure 2.3b shows the results of the accuracy of the tests when using Hawkes models with these parameters. For both the time reversal and Poisson cluster comparisons, as the value of μ increases, the power of the test increases. Again, the time reversal method is more powerful through all chosen values of μ. The reason for the increase in power as the value of μ increases is most likely due to the fact that the power of tests is naturally raised when the number of points increases. Logically, distinguishing between models should be much easier when you have more data to make that decision as compared to less(in an extreme, only have one or two points would make distinguishing between models impossible). The higher the value of μ is the more points that will be observed on average.

Finally the κ values were altered. The value of μ and σ were kept at their default values, and the value of κ was between .5 to .9, with 21 values evenly spaced on that grid. Figure 2.3c shows the power of the BIC test on this range of κ values. Similar to the μ values,

the higher values of κ show a higher level of power for both the Poisson cluster and time reversal method of the BIC comparison. Again, the time reversal method is more accurate over all values of κ. The most likely reasoning for this increase in accuracy as κ increases is partly due to the increased amount of clustering and partly due to the increased amount of points as κ increases. Similarly to the analysis of the simulations in which μ was changed, the power of a test increases as the number of points observed increases. However, the value of κ also increases the number of points in each "cluster" of points, or each set of points that can be traced back to the same background point. This allows for more information about what the clustering process looks like, which is crucial when distinguishing between different types of spatio-temporal clustering.

2.3.3 Issues with the BIC Method

There are several issues involving the BIC method that made this method not be desirable for general use. Firstly, the largest issue was the relatively low power over the values that were examined, especially for the Poisson cluster method. The values of σ in fact were specifically chosen as rather small, as any σ values that were chosen outside this range had almost no power. In cases in which a Poisson cluster process or inhomogeneous Poisson process were simulated, the test nearly always correctly did not select the Hawkes process, however this is not actually helpful in distinguishing if the power of the test is so low.

The other primary issue with the Poisson cluster BIC method is that the Poisson cluster model, being a clustering model, has the ability to fit to very small clusters that can lead to over fitting. The clusters found with the Poisson cluster process often only have a small number of points, which means that likelihood for the Poisson cluster model is very high. This level of over-fitting is not cancelled out completely by the use of the penalty for the cluster centers.

In addition, this test does not really have a measure to see whether or not the evidence for causal clustering is higher or lower. The only value that would be returned, should one

do this test on a single simulation, would be the BIC for two different models, and this is used to select the model with the lower BIC. The scale of the difference in the BIC is not known to a user of this method, and such the user does not get a clear understanding of whether a Hawkes model fits exceptionally better or simply just slightly better than another model. This method also does not work very well when comparing a Poisson cluster to a Hawkes model, as the number of parameters are very different, and this could potentially also be the cause of the low power of this method.

2.4 Hypothesis Testing Method

To look for evidence of causal clustering, the expected information gain statistic is used. The expected information gain statistic measures the change in entropy between some null model and some alternate model(Daley and Vere-Jones, 2016). Information gain measures how well a model predicts the next occurring point in the point process. The calculation of this statistic is often difficult to compute directly, however there are other methods of estimation. The mean log-likelihood ratio is a good approximation for this statistic, and it can computed as

$$\hat{G}_n = \frac{1}{N} \log \left(\frac{L_1}{L_0} \right). \tag{2.4}$$

L_1 is the likelihood for the alternate model in the hypothesis test, L_0 is the likelihood for the null model in the hypothesis test, and N is the number of points observed in the point process(Harte and Vere-Jones, 2005).

The idea of this hypothesis test is to set up a test with the following null and alternative hypothesis:

H_0: The data are generated from a Poisson cluster process.

H_1: The data are generated from a Hawkes process.

These null and alternate hypothesis do seem to only distinguish between two separate pro-

cesses, but the idea is that this can be extended to a more general test for testing between inhomogeneity and causal clustering. A Poisson cluster process, by virtue of having parent points "triggering" points both forwards and backwards in time, should mimic inhomogeneity very well. Meanwhile, a Hawkes process often fits well to both inhomogeneity as well as truly causal clustering, which creates an issue in distinguishing the two cases.

The logic in having the Poisson cluster process be the null hypothesis that there must be evidence to reject is based primarily on the higher bar for causal associations. Just like how causality is a much stronger claim than a correlation response, the assertion that certain points are "triggering" points should be taken as a much stronger claim than simple inhomogeneity. It should be noted that this method does not truly involve using causal analysis, and the "causal clustering" is not representative of modern causal theory. In cases where the underlying physical process is unknown, it is generally more acceptable to assume that there is not any sort of contagion occurring, as the implication of a contagion model is often much more substantial than a inhomogeneity model.

2.4.1 Monte Carlo Method for Estimation of Log Likelihood Ratio

Calculating the true information gain statistic is difficult to do, so as mentioned previously the mean log-likelihood ratio is used instead. In addition, the sampling distribution for the information gain statistic is not very well defined by any sources, so instead a Monte-Carlo method is used in order to perform the hypothesis test.

Once the Poisson cluster and Hawkes models have been fit to the data using the methods mentioned previously, the next step is to create a sampling distribution for the information gain statistic between these two models. Since there is no known sampling distribution that can be created using, for example, the central limit theorem, it is necessary to instead simulate values under the null hypothesis and compare these values to the true estimated value. This is similar to a randomization test or other test when the sampling distribution is unknown.

The full process for this is as follows. First, a Poisson cluster process is fit to the data, and the values of A, the average number points triggered by each parent point, μ, the expected number of parent points, and σ, the input to the Gaussian triggering density, are calculated. From here, Poisson cluster processes are then simulated with the same values of A, μ, and σ. From this, both a Hawkes and Poisson cluster model are fit to each simulation, and the likelihood values for the Hawkes and Poisson cluster model are calculated. Finally, the information gain statistic is calculated for each simulation. Then, the information gain statistic from the true data can be compared to the simulated sampling distribution, and the null hypothesis can either be rejected or be failed to be rejected.

2.4.2 Simulations for Hypothesis Test Method

Hawkes process are simulated to determine the power of the test, or the faction of the time that the test correctly rejects the null hypothesis of a Poisson cluster process for a Hawkes model. As before, a two dimensional Gaussian distributions is used fro the spatial triggering density, and a truncated Gaussian distribution with a lower bound of 0 is used for the temporal triggering density. The background rate is done using a constant rate of μ and the spatio-temporal region is a unit cube($[0, 1] \times [0, 1] \times [0, 1]$).

In each simulation, the likelihood for the standard Hawkes process, a Poisson cluster model, and the "backwards" or "reversed" Hawkes model were calculated. By design, this model will fail to reject the null hypothesis with a probability of .95 given that the data is truly a Poisson cluster model. However, similar simulations showed that consistently this method also rejected the data when the data was a simulated inhomogeneous Poisson model with a probability of at least .95, an even lower in cases without any significant clustering such as with a homogeneous Poisson model.

In particular, the simulations were focused on looking at different values of μ, the background rate, κ, or the productivity, and σ_s and σ_t, the spatial and temporal triggering density variances. The spatial and temporal values were allowed to vary to allow models in which

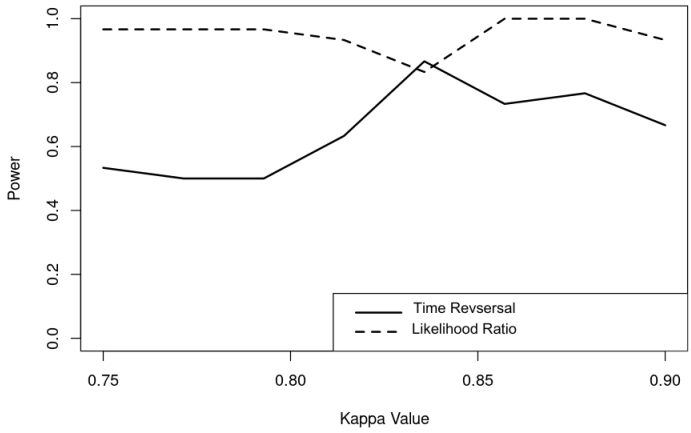
the observation window is not a unit cube.

The values of κ, σ_s, σ_t and μ were set to values similar to the values there were used in the applied problem of crime data in a later section. For κ, 100 Hawkes processes were simulated, with $\mu = 18$, $\sigma_t = .0002$ and $\sigma_s = .0002$. The power over values of κ from .75 to .9 can be found in Figure 2.4a. The Poisson cluster test has higher power over the time reversal test for nearly all values of κ. The power of the test also increases as the value of κ increases, with a peak at $\kappa = .834$ followed by a slight decline.

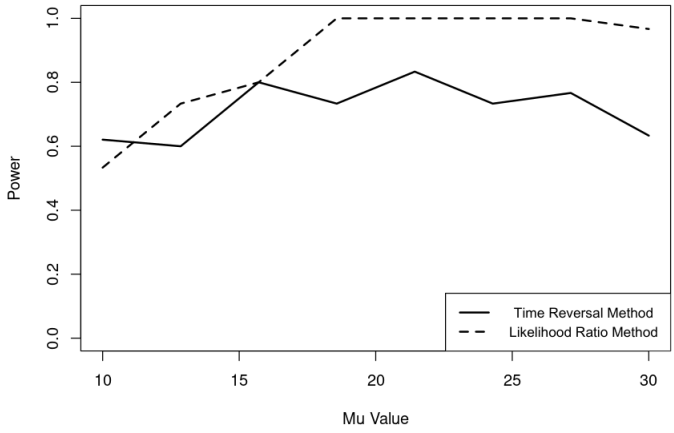
The power of Poisson cluster and Time-Reversal tests as a function of σ_t, for 100 simulated Hawkes processes with temporal standard deviation σ_t, each with $\mu = 18$, $\kappa = .81$, and $\sigma_{xy} = .0002$, where for each simulated process can be found in Figure 2.4c. The power of the Poisson cluster method is high over all values of σ_t, although the power of the time reversal test decreases as the value of σ_t increases.

Next the power of the Poisson cluster and time reversal tests were examined as a function of σ_s, the spatial clustering variable. This was done for 100 simulated Hawkes processes with spatial standard deviation σ_s, each with $\mu = 18$, $\kappa = .81$, and $\sigma_t = .0002$, where for each simulated process. 100 Poisson cluster processes were fit by MLE in order to obtain the sampling distribution. As seen in Figure 2.4d, the power decreases as the value of σ_s increases for both tests. The Poisson cluster method is more powerful than the time reversal method all values.

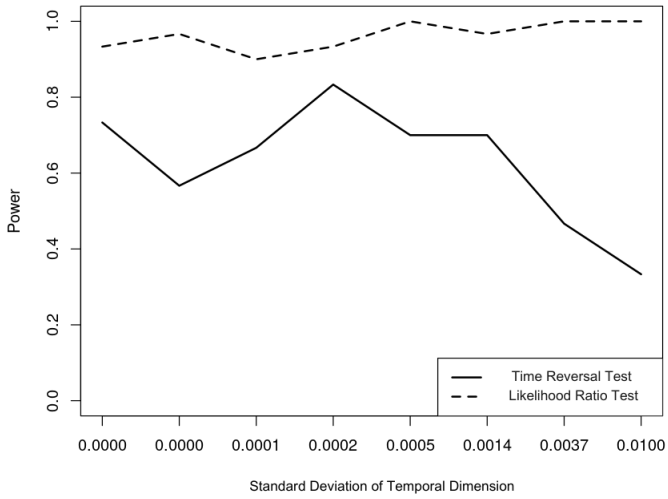
Finally, the power of the hypothesis test and reversal tests were examined as a function of μ. For 100 simulated Hawkes processes with background rate μ, each with $\kappa = .81$, $\sigma_t = .0002$ and $\sigma_{xy} = .0002$, where for each simulated process, 100 Poisson cluster processes were fit by MLE in order to obtain the sampling distribution. The results can be seen in Figure 2.4b. The power of the test generally increases for both methods as μ increases. This is true for both the time reversal and Poisson cluster method, and the Poisson cluster method shows a higher power for nearly all values of μ except for lower some lower values.



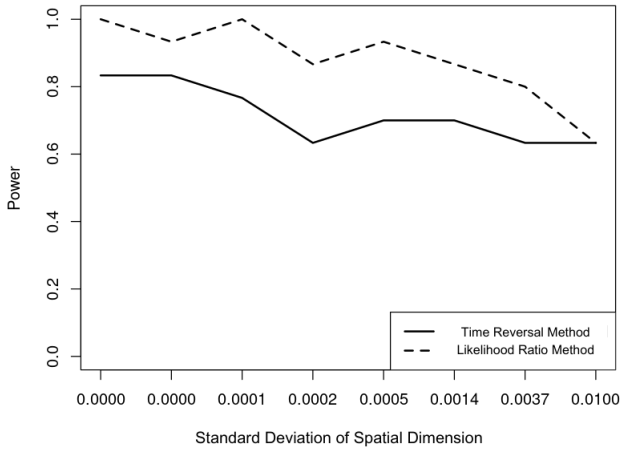
(a) κ



(b) μ



(c) σ_t



(d) σ_{xy}

Figure 2.4: *Power of the BIC test over the inputted* μ*,* κ*,* σ_t *and* σ_{xy} *values for a simulated Hawkes process. The dashed lines represent the power for the Poisson Cluster(Neyman Scott) method and the solid lines represent the power for the Time Reversal method*

The reasoning for the power fluctuating throughout these observed values are most likely due to the appearance of these different clustering processes. As shown in Figure 2.2, the appearance of the clusters within the spatio-temporal process can vary wildly depending on the inputted parameters. For κ, the higher values of κ have a larger amount of points in each cluster. The shape of the cluster is critically important in determining which model will fit better. For a Hawkes process, there is an individual starting point that sets off a chain of points. Each point then potentially triggers additional points, creating a series of chains coming out of the singular starting point. Meanwhile, for a clustering process, there will be less of that "chain-like" appearance to the data. When the value of κ is higher, the individual clusters are much larger, which most likely makes differentiating these two types of clustering easier.

The reasoning for the decrease in power as either σ parameter increases has a similar cause. As the value of σ increases, as shown in Figure 2.2, the appearance of the clusters are much harder to identify. In some cases, the process ends up looking similar to even a homogeneous Poisson process. The core difference in these two processes, which are going to be the structure of the clusters, are going to be difficult to determine if the relationship between the different points in the clusters are difficult to determine.

The power increasing as μ increasing most likely is due to the increased information with more points being observed. Any test will increase in power with more data and information, and the more points that are observed means the more power a test will have. This also allows for more clusters to examine and determine the causal structure of the data. Overall the function of power with μ is similar to that of the function of power with κ.

2.4.3 Application to Crime Data

Recorded data on 8,862 reported illegal shootings in Boston between 2015 and 2021 were collected from the public data source for the Boston government (https:// data.boston.gov /dataset/shootings). Figure 2.5 shows a kernel smoothing of the locations of these reported

crimes. All points with a time and location were used. The excluded data included data in which the latitude and longitude were entered as a default value of a location in Disney World in Florida, and it was assumed that this meant that the data did not have a location recorded.

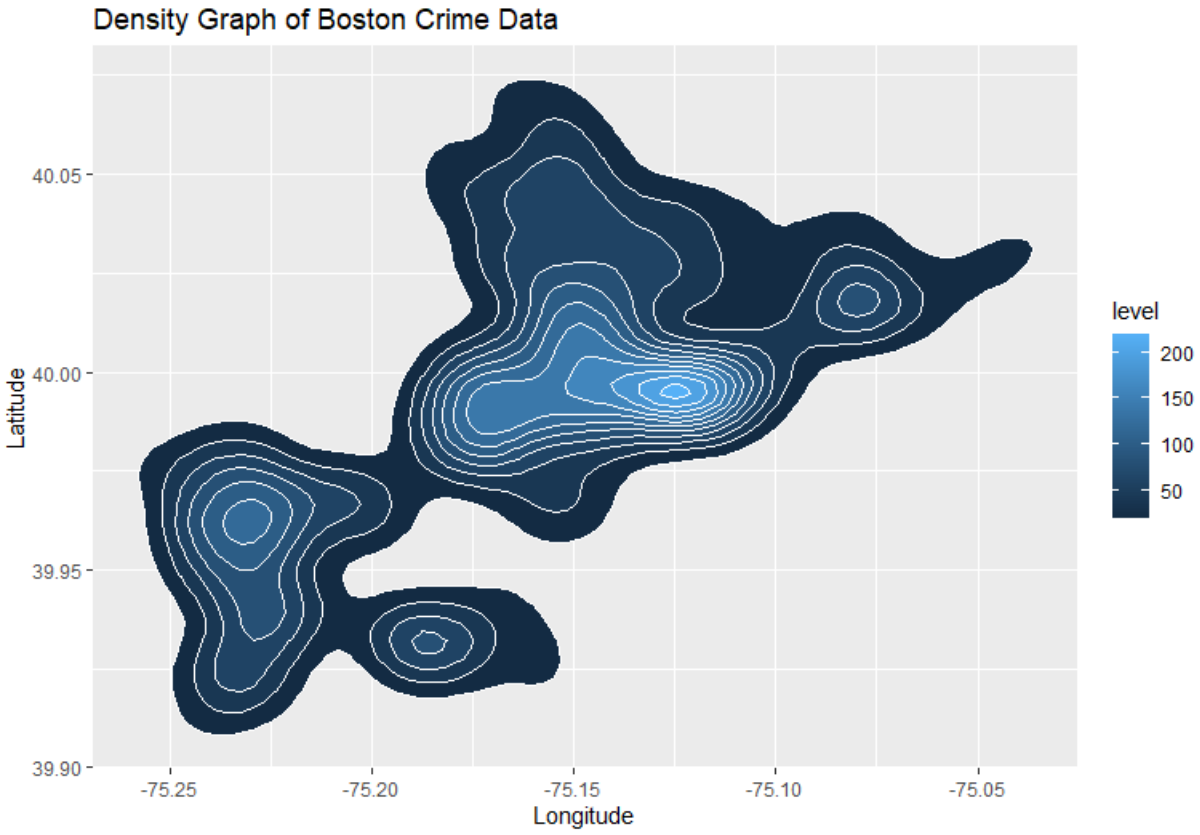


Figure 2.5: *Density Graph of Boston Crime Data*

The data were divided into uniform 10×10 grid cells, each analyzed individually using the tests described previously. Grid cells including less than 5 points were excluded from the analysis as the tests have insufficient power in such cases, and the amount of data is too low to make any real conclusions. The data was split into this grid for computational reasons-the fitting of the maximum likelihood for the Hawkes, especially, is incredibly computationally intensive, even without having to calculate the integral through Monte Carlo methods. This allowed for each grid to be analyzed individually. Any spillover effects were considered to be

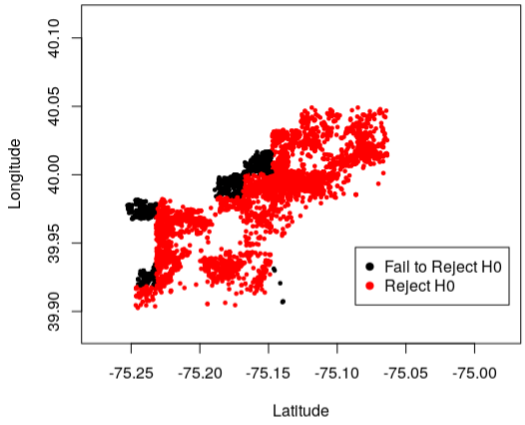
minimal and not considered, however a more in-depth analysis might be needed to confirm if that is true.

The data shows some evidence of spatial clustering. This is most likely due to the geography of Boston, with the highest level of crimes occurring in areas that are the most populated. Crime is typically clustered in certain areas of a city which is also likely the cause of some of the spatial clustering.

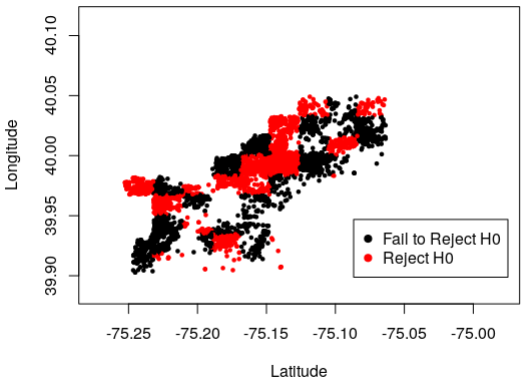
For each remaining grid cell, a Poisson cluster model was fit by maximum likelihood estimation to the data within the cell, and then realizations of Poisson cluster models were simulated repeatedly with parameters equal to these maximum likelihood estimates, to create a sampling distribution for the information gain statistic.

For each simulation, the likelihood, L_1, for a Hawkes model, and the likelihood L_0, for a Poisson cluster model, were calculated, and used to calculate $\hat{G}_n$. This creates a sampling distribution for the value of the information gain statistic, and the value of $\hat{G}_n$ for the actual data is then compared to this sampling distribution. If the value of $\hat{G}_n$ is above the 95% percentile for the simulated sampling distribution, then we say the test rejected the null hypothesis. For the time-reversal test this procedure was repeated, but with L_1 as the likelihood of the Hawkes model given the data and L_0 as the likelihood of the Hawkes model with the times reversed.

Figure 2.6a shows the results of the Poisson cluster hypothesis test. Each dot represents a single shooting within a grid, and the data points within each grid section have been colored based on the result of the hypothesis test within that grid section. The results suggest that for the majority of locations in Boston, there is significant causal clustering present in the data on recorded shootings. Of the grid sections that were included within the analysis, 84.6% resulted in the test rejecting the null hypothesis, and these sections contained 83.7% of the total reported shootings. At the same time, there are several locations, especially on the Northwest borders of the data set, where the test fails to reject the null hypothesis and suggests that the local aggregation of points in these locations may be entirely due to



(a) *Boston Clustering Poisson cluster Hypothesis Test Results*



(b) *Boston Time Reversal Test results*

Figure 2.6: *Boston*

inhomogeneity.

The time reversal test results, shown in Figure 2.6b, have far more grid cells where the test fails to reject the null hypothesis. The time reversal test only rejected the null hypothesis in 44.2% of the grid cells, corresponding to a total of 46.8% of the reported shootings. The majority of grid cells where the Poisson cluster test failed to reject the null hypothesis also had the time reversal test fail to reject the null hypothesis, again suggesting inhomogeneity as the dominant cause of aggregation of points in these areas.

The results of the Poisson cluster test indicate that, in the vast majority of locations within Boston, the reported shooting data from 2015-2021 are significantly better fit by a Hawkes model with causal clustering than by a Poisson cluster model. Since the test fails to reject about 95% of the time when either a Poisson cluster model or inhomogeneous Poisson model is the actual data generating mechanism, the results suggest that the clustering these points is truly causal.

The results provide evidence that a Hawkes model with causal clustering may be ap-

propriate for certain crime data. However, there are still some areas, especially near the Northwestern borders of the observation region, where causal clustering is not indicated. This could possibly be due to spatially varying covariates, differences in gang territory, or other factors resulting in more causal clustering in certain locations rather than others.

The Poisson cluster and time reversal tests resulted in substantially different classifications. One possible explanation for this can be seen in the power analysis indicated by the simulations, since the Poisson cluster test had higher power than the time reversal test in most cases. Therefore, the reason that so many more sections failed to reject the null hypothesis using the time reversal test could be because the power of this test was too low.

Distinguishing between causal clustering and inhomogeneity in point processes is still a problem requiring much further study. Simulations show that under certain conditions, a simulated Hawkes model can be correctly distinguished from a Poisson cluster model using the information gain statistic, and furthermore, the test appears to have high power in distinguishing a Hawkes model from an inhomogeneous Poisson model as well. The time reversal test, by contrast, has somewhat lower power. This power is affected by the parameters of the simulation, with larger data sets and more intense clustering resulting in higher power for both tests.

Hawkes models have been used extensively in crime data analysis(Sha et al., 2020; Olinde and Short, 2020), typically without much investigation into whether or not the assumption of causal clustering is indicated. Models without causal clustering, such as inhomogeneous Poisson models or Poisson cluster models, may fit just as well to the data in some situations. However, with regard to the application to the recorded shooting data in Boston, our results do suggest strong evidence of causal clustering in most areas of the city.

However, while the level of clustering was significant(meaning that the value of κ was statistically significant from 0), there is still some analysis of κ that should be investigated when looking at application of Hawkes processes. The value of κ can be transformed to calculated the expected number of points that are triggered by each starting point($\mathbb{E}[g]$)

from the mathematical expression

$$\mathbb{E}[g] = \frac{1}{1-\kappa}. \tag{2.5}$$

This equation should be used in determining whether or not that Hawkes process makes sense in a certain case, and whether or not it is reasonable to have that level of triggering within the data. For this data, the value of κ was typically between .3 and .5, indicating that each point triggered between 1 and 2 other points. This could be considered reasonable, however it is likely this is an overestimation. Fitting a stronger background rate would most likely allow for a more reasonable estimation of the level of true causal triggering within this model.

Future research should investigate this evidence of causal clustering further. Here, we considered Gaussian triggering functions for both the Poisson cluster and Hawkes model, but alternative triggering functions could be considered. In addition, we allowed each spatial grid cell to have its own background rate, to account for spatially varying covariates such as poverty levels or education levels. Future work could alternatively model the background crime rate more more explicitly as a function of such socio-economic covariates(Park et al., 2021). In addition, other types of reported crime data should be analyzed and the relationship between different types of crimes and the strength of evidence of causal clustering should be studied.

2.5 Discussion of "Neyman Scott Process" language

The likelihood of a Neyman Scott process, as mentioned previously, is difficult to calculate exactly. There are methods for estimating the cluster centers and parameters(Tanaka et al., 2008; Zhuang, 2018), however these methods rely on using methods such as second-order intensities or palm likelihoods. For this reason, the process being estimated, while similar to the form of a Neyman Scott process and using the same ideas, might be more carefully described as a Poisson cluster process. There are times when these two processes are used

interchangeably in this dissertation, however this is not an issue as while the process can be generated as a Neyman Scott process, in reality the likelihood of the process is calculated as if it is a Poisson cluster process. This section will overview the specifics of this and also describe why this is not an issue for the hypothesis test described.

The clustering model is fit using a Poisson cluster algorithm with maximum likelihood estimation. The number of cluster centers is calculated using BIC, with a penalty for each addition cluster to balance fitting well to the data and preventing over-fitting. This process also calculates a value for σ from the Gaussian clustering algorithm, which is assumed to be a certain value spatially as well as a certain value temporally, resulting in a value of σ_t and σ_s.

Once that model is fit, it is assumed that the cluster centers, or parent points using a Neyman Scott paradigm, are known. This process can be labeled as N^c. The average number of points over all clusters is referred to as A. The conditional intensity for each point, given these cluster centers, can be calculated simply as

$$\lambda(x, y, t) = A \sum_{(x_i, y_i, t_i) \in N^c} g(x - x_i, y - y_i, t - t_i).$$

This provides an accurate measure of how well the data is fit to the data, meaning that data that fits very well to this clustering model will have a higher combined product for an intensity than data that has lower values overall for the intensity. The integral over the whole area is then subtracted, which, as explained previously, can be just seen as AM with M referring to the number of cluster centers. This therefore also allows for a penalty for the number of cluster centers that is then scaled by the amount of data.

it is important to note that the goal of this hypothesis test is not to compare two different models and determine which one fits better. The goal, rather, is to look at a model that simply shows some level of inhomogeneous clustering and compare that to a Hawkes model to see if the Hawkes model fits significantly better. Any model with clustering will fit well to the clustering model that was defined in this section, depending on what triggering function

is used. The models here have similar triggering functions in order to narrow the model space as well as to provide a fair comparison between these two types of models. The decision of which triggering function to use is largely an intractable problem-there are an infinite amount of triggering functions that could be examined and an infinite amount of parameters that could be considered.

The rejection of the null hypothesis is still consistent in this case-it should be assumed that if you reject the null hypothesis, the Hawkes process fits significantly better than a clustering model and therefore there is evidence that the clustering within the data is truly causal and not the result of inhomgeneity. However, failing to reject the null hypothesis does not mean that the data is actually generated by a Neyman Scott process. In fact, from the standard definition of a Neyman Scott process, this would not be reasonable in the majority of applications. With regards to crime, that would mean shootings are truly caused by some central parent process that is unobserved and triggers points both forwards and backwards. For diseases, it would again mean that some sort of unseen parent process is causing infections both forwards and backwards in time.

For that reason, the null and alternate hypothesis can also be written instead as

$$H_0\text{: The data are generated from a non-causal Poisson cluster model}$$
$$H_1\text{: The data are generated from a Hawkes process.}$$

Which more accurately describes the hypothesis being tested and provides a more clear explanation of what the failure to reject the null hypothesis should mean to an individual performing this test. Another proposed method of hypothesis tests would instead of focusing on the exact model that is involved, instead to specify some value β, which could be described as the level of contagious or causal clustering within the dataset. The null hypothesis would be that β is 0, or that all the clustering can be explained by inhomogeneity. Any other value of β greater than 0 would mean that some of the points can be explained by inhomogeniety(the points that are from the background process in a Hawkes model) while

some could be explained by causal triggering.

Therefore, a third way of phrasing the hypotheses, without changing the underlying method of rejection of the null hypothesis, would be using this value of β to define the null and alternate hypothesis, or

$$H_0\colon\ \beta = 0$$
$$H_1\colon\ \beta > 0.$$

. Again, the acceptance or rejection of these hypotheses remains consistent from the previously described method. The underlying idea is that in order to claim that there is evidence of causal clustering instead of inhomogeneity, it is necessary to prove that a Hawkes model with causal clustering fits significantly better than a Poisson cluster model, or a model with non-causal clustering.

The question of whether or not this is an unreasonable model to fit at all to the data, and that the hypothesis test is simply always going to show that a Hawkes model is better because of the underlying structure of the null model has also been argued as a downside to this method. This concern is not entirely unfounded, however from simulation and analysis it appears to not be as problematic as expected. Since the null model that is fit is essentially a generic clustering model, this model will fit reasonably well to both Hawkes models and inhomogeneous data. While Hawkes models have a different conditional intensity than clustering models, in practice the data is often fairly similar to a Poisson cluster process in appearance. Each "cluster" is started by a point that is triggered by the background process, and then the cluster consists of that original point and all of the points that it triggered. As seen in the simulated hypothesis tests, the power can often be very low if the level of triggering, or the κ value, is too low. This means that the worry about the amount of false positives is most likely unfounded, and that Poisson cluster models and Hawkes processes do look relatively similar.

This method has been demonstrated to work on spatio-temporal cases with a Gaussian

triggering function. There is significantly more options to explore for this method-including using different triggering functions, as well as applying this to datasets that have known triggering models and testing whether this process correctly identifies the type of triggering. The next chapter attempts to do that with disease data, where the underlying process of inhomogeneity or contagion is known due to the selection of diseases with known infection methods.

2.6 Further Research

There is much further research that can be done on this method. This analysis, notably, focused on only a Gaussian triggering density. This was done because of the clear analogue for a density that is contained on the real line and a density that is contained on the positive real line, in order to have a similar density for both the Poisson cluster as well as the Hawkes model. Truncated Gaussian distributions are, however, not frequently used for Hawkes models, and an exploration of other types of models such as exponential or power distributions should also be explored. There are also other types of spatio-temporal data in which the exploration using this method would be interesting-both data that has no causal clustering as well as data with known causal clustering. Due to the relative scarcity of data with unknown triggering methods that is spatio-temporal, the crime data used here was highly unusual. However, there are several other types of spatio-temporal data that Hawkes processes are used on that could potentially be explored.

Overall, this analysis shows that it is not entirely reasonable to simply fit a Hawkes model to data and assume that if the fit is good, then the process is truly causal. This is a method that allows for essentially no distinguishing between "correlation," or points simply being clustered in groups, and "causation," or points directly triggering other points. While this hypothesis test might not provide all the answers, it is also another way to check the reasonableness of claiming that a point process has causal clustering within the data. This

does not need to be the only check that one might use when determining if a Hawkes process is reasonable to use for the data. Another method is to look at the κ value and determine the reasonableness of this value in the context of the data. As explained before, you would expect each parent point to trigger $\frac{1}{1-\kappa}$ points total. Therefore, if the number of points triggered by a parent point is unreasonable from the data, it is another check to see if a clustering model is more appropriate to use.

Another area of exploration is involving looking at models in which clustering between distinct areas is explored. For exploration of disease data, which is usually observed on a regional level, it is common to model the spread between communities based on mobility data or other information(Chiang et al., 2022). There is a possible extension of this method that could be explored with this idea. While it is reasonable to assume some level of spread between communities, it is common when fitting Hawkes models to disease data to show a large amount of spread between communities that are far away.

As an example, when attempting to model the spread of Covid-19, it was found that there is a large amount of spread between New York and Los Angeles. This is, obviously, not reasonable. It is likely that this is similar to the core problem explored within this analysis-attempting to discern between inhomogeneity and causal triggering. In this case, the inhomogeneity is that there are certain times when Covid-19 was higher everywhere, and these periods were often slightly apart in similarly large regions. So, while New York City might have an increase in infections a week or two before Los Angeles does, it is not necessarily because there is a large amount of infections transferring from New York to Los Angeles specifically.

An extension of this method could therefore be done in which it could be determined if there is in fact significant triggering between different regions. For each set of regions to compare, both a model with triggering between the regions as well as a model where the triggering between the regions is non-causal could be done. This would provide a test to see if there should even be triggering between regions considered. Another method could be

fitting a Hawkes model to each region specifically, and do a hypothesis test on the points that are remaining. It might be assumed these points are the result of triggering between regions, but if that data could be explained with a non-causal clustering model then it might be decided that adding in triggering between regions is not needed.

Overall, this method should be considered whenever fitting Hawkes models to data in which the triggering mechanism is unknown. While it is not a perfect solution, just like with causal inference, the best solution would be observing the data over multiple realizations, which is not possible in the real world. For that reason, any guardrails when it comes to stating that a method involves true causality should be heavily questioned and looked at critically to prevent associating what is essentially correlation with causation.

Future work in this area of study will hopefully provide more research into this hypothesis test method, as well as potentially hypothesis tests with different a different set up for the null hypothesis or a different test statistic. Any of these changes might result in a more accurate hypothesis test to deal with this problem. Overall, the problem of distinguishing between causal clustering and inhomogeneity will most likely never be solved fully, as the theoretical solution is impractical, but hopefully further research will expand on the ideas and methods expressed in this chapter.

CHAPTER 3

Social Contagion Model Analysis with Causal Clustering

3.1 Introduction and Motivation

As in the previous chapter, this chapter looks at the case where there is debate as to whether clustering is simply the result of inhomogeneity or the result of causal clustering. In the previous chapter, simulations of true spatio-temporal Hawkes processes was followed by an analysis of spatio-temporal crime data. This method is slightly adjusted as well as expanded to different types of data in this section.

Hawkes processes are also commonly used for modeling infectious diseases(Sun et al., 2021; Rizoiu et al., 2018; Choi et al., 2015; Junhyung Park and Schoenberg, 2022), as the contagion model of an infectious disease intuitively makes sense for a Hawkes process. A point, in this case, refers to an individual that has contracted the infectious disease. The background points, in this case, would refer to either people introduced into the study area that are infected due to travel or people that potentially acquire the disease in a way that is not the result of a person-to-person transfer. The "triggering" within the Hawkes process is the spread of the disease, and the κ value would be the average number of people that each person infects. A simplistic Hawkes model without a changing κ value or other modifications would slightly struggle with the case of a very small population, however, in larger populations these models are still helpful to use.

There are cases, however, where there is doubt about the exact mechanism of a disease's

transmission, or if there is in fact any triggering involved at all. An example would be the concept of social contagion-a broad theory that involves looking at the effects of social interactions on certain behaviors or diseases(Christakis and Fowler, 2013). Social contagion models has been modeled previously using Hawkes processes, based on the idea that social contagion is essentially an infectious or causal clustering process(Palmowski and Puchalska, 2020) Specifically, there is research that suggests that suicide(Bearman and Moody, 2004; Cheng et al., 2014) or non-suicidal self-injury(Jarvi et al., 2013), especially in adolescents, can be affected by social contagion. Studies that advocate for this theory do typically find some level of spatio-temporal clustering within suicide data. However, it is possible that this spatial-temporal clustering might be explained as resulting from the large variation in certain covariates such as poverty, mental health issues, and gun ownership, all of which are correlated with suicide rates.

In addition, many studies simply use survey data that examines whether or not individuals who know suicide victims are more likely to proceed to suicide, which is technically a correlational approach, even when controlled for any observed covariates. Studies that have challenged this narrative typically rely upon attempting to isolate a "social contagion" factor amongst other variables that could lead to suicide(Ali et al., 2011), but such a method relies upon the assumption that no significant unobservable variables are confounding the results, which is not necessarily a reasonable assumption.

There are several additional points that also will be covered in this chapter to investigate the usefulness of the hypothesis test previously developed. The first major change is that this will look at temporal data that has been binned into weeks, instead of spatio-temporal data. In addition, this chapter will determine how well the test works on actual infectious diseases with known disease processes. The level of triggering will also be considered by investigating the κ value that is estimated to each value to determine if the levels of triggering correspond to what is expected for diseases with known disease processes and infection levels. Finally, this chapter will look at whether or not the result of this hypothesis test suggests that suicide

is indeed an epidemic that has noticeable and significant contagion.

3.2 Hawkes Models with Binned Temporal Disease Data

The collected data for diseases and deaths are often subject to privacy concerns, so the most granular data available for study is temporal data that has been binned, often into weeks or days. This presents a slight challenge as well as a potential modification of the power of the hypothesis test that was defined in the previous chapter. The previous chapter focused on a spatio-temporal Hawkes model, and a purely temporal Hawkes model has a conditional intensity of

$$\lambda(t|\mathcal{H}_t) = \mu + \kappa \sum_{i:t_i<t} g(t - t_i). \tag{3.1}$$

The μ and κ parameters function similarly to the spatio-temporal case. However,for the temporal case, the triggering density, represented as $g(\delta_t)$, is different in that it only expresses temporal triggering.

Within a contagious disease model, μ would typically represent the background rate of immigration of the disease into the population of interest, and the triggering element would represent spread between individuals. The value of κ would represent the speed at which the disease is spreading on average-a higher κ value corresponds to a more highly contagious disease.

It is possible and reasonable to fit a Hawkes model to data that has been binned into discrete values(Browning et al., 2021), as in this case some discrete density would be chosen for the triggering density. For instance, in the case that the data are weekly aggregates and the geometric distribution is used for g, then $g(k) = p(1-p)^k$, for $k = 0, 1, 2, ...$ weeks. This provides a model that follows a reasonable triggering density in which the highest values are closest to 0 and then decreases as the number of weeks increases. This replaces the previous parameter for the Gaussian spatio-temporal density of σ for the measure of spread with the parameter p, which also can be seen as a measure of the spread of the triggering temporally.

Simply fitting a Hawkes model to adolescent suicide data would not necessarily definitively determine whether or not there is evidence of contagion within the data. Hawkes models can fit well to clustered data, regardless of the true clustering mechanism. That is, even if the aggregation of points in the point process is purely the result of inhomogeneity in explanatory variables, a Hawkes model representing triggering of points might nevertheless offer satisfactory fit. For this reason, more specialized methods are needed to distinguish between contagious clustering and clustering that is the result of inhomogeneity or non-causal clustering.

Diseases with known contagion methods can be compared to suicide data in order to evaluate and compare the fit of different clustering models. Three diseases that can be compared to suicide data in adolescents in order to study the social contagion theory are measles, Chlamydia, and Lyme disease.

Measles is a highly contagious disease that spreads rapidly through populations, so it can potentially be modeled accurately using a Hawkes model(Laksono et al., 2016). Since the level of contagion in measles is very high, the value of κ should be very high and it should be very easy to distinguish between whether or not there is epidemic spread of the disease using a hypothesis test method. Chlamydia is a sexually transmitted disease that is not as highly contagious as measles since the level of contact needed for exposure is much higher than for measles(Althaus Christian L., 2012). Therefore, it is expected that the level of contagion will be lower than for measles, however it should still show evidence of contagion. While there are outbreaks of Chlamydia within populations, typically the amount of people that an individual can infect is much smaller than with measles. Lyme disease is non-contagious from human to human but cases tend to be highly clustered, as the disease is primarily spread through ticks, and this exposure is much more likely to happen during warmer weather(Roome et al., 2018). Therefore, it is expected that the hypothesis test should show that while there is clustering present, there is not contagion happening with the data.

3.3 Hypothesis Testing Method

A temporal Poisson cluster process N can be viewed as an example of a cluster process, where the intensity is random, as it depends on the random collection M of hidden parent points, but given M, the process N is a Poisson process with intensity

$$\lambda_{NS}(t|M) = A \sum_{i:t_i \in M} h(t - t_i).$$

The parameter A represents the average number of points that each parent triggers, and $h(\delta_t)$ represents the clustering density. Under this parameterization, this conditional intensity is not based on the history of the process N but instead is based on knowing the hidden parent process M. In addition, the clustering function $h(\delta_t)$ is not limited to positive values unlike the triggering density in a Hawkes model.

In order to estimate the sampling distribution of the information gain statistic under the null, the previous chapter proposed a Monte Carlo method. Specifically, in the context of the datasets analyzed in the present analysis, a Poisson cluster process is first fit to the data using a Gaussian triggering density. A Gaussian clustering algorithm is then performed using maximum likelihood estimation, and the parameters for the Poisson cluster process are estimated using the number of clusters (μ), the mean number of points per cluster (A), and the standard deviation of the clusters (σ). Poisson cluster processes with these same parameters are then simulated, and a Hawkes model is fit to each Poisson cluster simulation by maximum likelihood. The information gain statistic is then calculated, and the sampling distribution used is the collection of information gains between the Hawkes and Poisson cluster model log-likelihoods for all of the simulations. Here, 500 simulations were used for each dataset. The information gain statistic is then calculated for the original data in the same manner and compared to the simulated sampling distribution to determine if the null hypothesis can be rejected.

Even though the Poisson cluster model that is fit here is not discrete, there is intuition behind why this is not problematic under this hypothesis testing paradigm. Firstly, this

hypothesis test is not attempting to distinguish, truly, between a Poisson cluster and Hawkes model. Instead, this process is attempting to distinguish between inhomogeneity, which can be modeled well with a Poisson cluster model, from causal clustering. Therefore, the theoretical properties of this Poisson cluster model, or even whether or not this model is a "true" Poisson cluster model, is not relevant to the analysis. In addition, since this involves a simulated sample distribution, the only relevant information is how extreme the general fit of the Poisson cluster model is as compared to a Hawkes model for the simulations as compared to the true data. Therefore, while the scale of the differences are affected, the relative values of the fits between simulated and true data is still comparable for this method.

3.4 Disease Data and Descriptive Analysis

Adolescent suicide statistics were collected from the CDC Wonder Provisional Mortality Statistics database which begins at 2018 and collects data through the most recent week. The provisional mortality statistics are based upon death certificates for United States residents, and any category with less than nine deaths is suppressed, as the CDC cannot guarantee the accuracy of low numbers of deaths. No suppressed results were included in this analysis. While it would be preferable to have data that was more centralized to a certain location, any level of

The age range selected was from the five year age groups of 10-14 and 15-19. The method of death was limited to UCD- ICD-10 Codes of X60-X84, which are all intentional self-harm deaths. Specific method of self-harm death was not considered. The data selected was based upon the years that the CDC did not label as "provisional" in their reporting, which are the dates 2018-01-01 to 2021-31-12. The data was collected weekly as this was the smallest time interval is released, and there were no suppressed values in the output. Measles data(Van Panhuis et al., 2018b), Chlamydia data(W. et al., 2018), and Lyme data(Van Panhuis et al., 2018a) for the United States were collected from Project Tycho, which compiles

weekly case counts of various diseases(van Panhuis, 2013), with some geographical information included.

Measles data consisted of weekly measles cases in Los Angeles County from 1928-01-01 to 1931-31-12 collected from Project Tycho. The length of the data collected was chosen to match the number of weeks available for the adolescent suicide data, to make sure the power of the tests were similar. The data was limited to Los Angeles as different counties in California reported at different frequencies, and Los Angeles county reported data consistently. The time period was selected as the first 4-year time period in which data was consistently reported weekly.

Lyme data compiled in Project Tycho consisted of weekly case counts of Lyme disease in California from 2008-01-01 to 2011-12-31.The four year period matches the length of the observed data for adolescent suicide deaths. The years 2008 to 2011 were selected for this analysis as this was the first four year time period to have consistent data reported weekly. The data were limited to the state of California, which had the most consistent reporting of weekly case counts, with no weeks missing during 2008-2011.

Weekly case counts of Chlamydia in California from 2008-01-01 to 2011-12-31 were compiled in Project Tycho. The time period and state was chosen to match the Lyme disease data, and the Chlamydia data also had the same consistent weekly case reporting in California during that time period.

First pass observation of the data shows some key differences between the different temporal data sets. Firstly, the measles data set has the clearest evidence of clustering, with some obvious outbreaks starting at around 103 weeks and 155 weeks. These are major outbreaks that far exceed the average number of measles cases outside of those outbreaks. The Chlamydia and Lyme data do seem to show some outbreaks within the data. There seem sot be a trend of higher cases of Lyme, especially early within the time observation period. The Chlamydia data has an outbreak around the 250 week mark with some level of increases cases in that period.

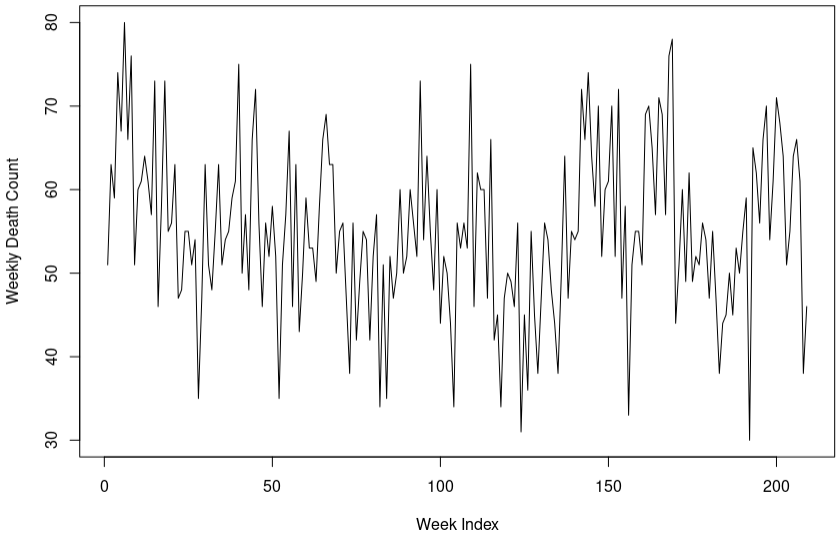


Figure 3.1: *Weekly reported case counts for adolescent suicide data in the United States from 2018-01-01 to 2021-31-12*

The adolescent suicide data it slightly harder to determine without a formal test, which will be done later in this chapter. There does seem to be some level of increased and decreased periods within the temporal observation region. However, it is also possible that there is simply randomness within the data that is suggesting a level of clustering. Regardless, it is possible to check for both clustering as well as if there is significant temporal clustering looking at the significance of the value of κ within the Hawkes model. There are other measures of significance in clustering that could also be done, however since this is concerned primarily with a point process framework that is the metric that will be examined.

3.5 Results of Hypothesis Testing

For measles, the estimated value of the information gain statistic is well above any of the values in the sampling distribution, as seen in Figure 2a. The p-value is essentially 0, so the null hypothesis is therefore rejected. This corresponds to the expected result, since measles

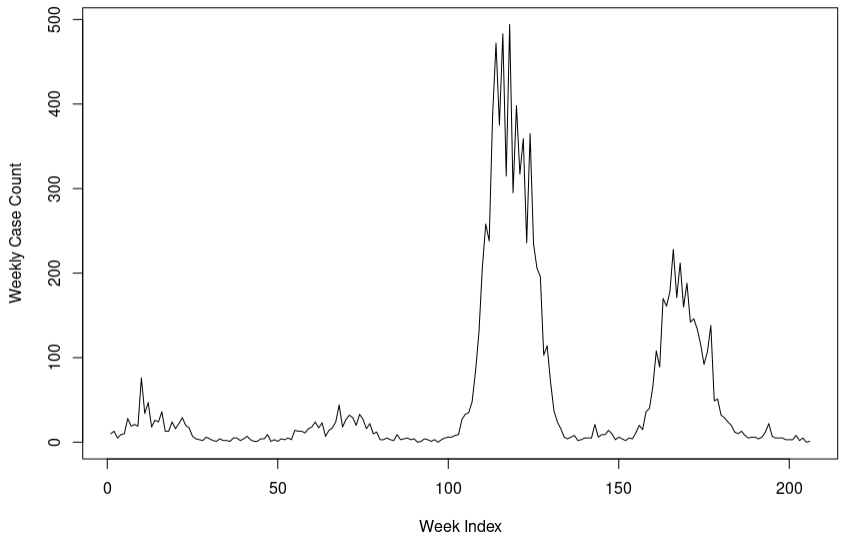


Figure 3.2: *Weekly reported case counts for measles cases in Los Angeles County, from 1928-01-01 to 1931-31-12*

is highly contagious. The estimated value of the κ parameter is 0.973 with a confidence interval of (0.954, 0.993), indicating significant clustering in the Hawkes process.

For the Chlamydia data, the p-value of the estimated information gain statistic is 0.006, so the null hypothesis is again rejected. The estimated κ parameter is 0.580 with a 99% confidence interval of $(0.569, 0.591)$.

For Lyme disease, the estimated information gain statistic has a p-value of 0.061, indicating that the null hypothesis is not rejected. The maximum likelihood estimate of κ is 0.575 with a 99% confidence interval of (0.429, 0.720).

The estimated information gain statistic applied to the youth suicide data results in a p-value of 0.004, indicating that the null hypothesis is rejected, though the corresponding κ estimate is just 0.128 with a 99% confidence interval of $(-.012, .269)$.

Of the four conditions considered here, measles, Chlamydia, Lyme disease, and suicide, the likelihood ratio tests for measles, Chlamydia, and suicide suggest that the Hawkes model

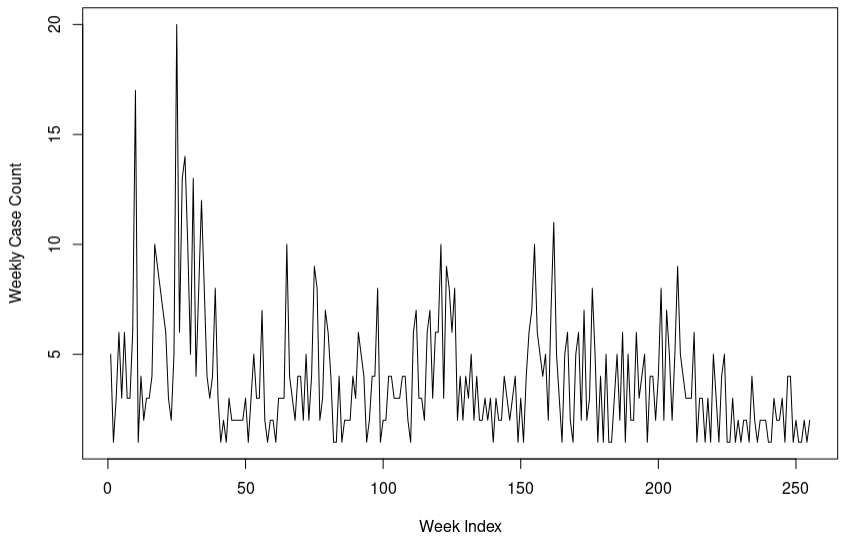


Figure 3.3: *Weekly reported case counts for Lyme cases in California, from 2008-01-01 to 2011-31-12*

fits significantly better than a Poisson cluster model. The only condition for which the likelihood ratio test did not reject the null hypothesis was Lyme disease.

This information is highly encouraging with regards to the accuracy of the test in investigating epidemic diseases. Lyme disease is not epidemic, however, it does have a level of clustering due to the nature of its infection. Lyme disease is much more common in the summer months(Department of Health, 2019) due to when tick populations are at their most active and there are increased outside summer activities.

The measles and Chlamydia data having a significant level of causal or epidemic clustering is also in line with the expected results. In fact, the measles data had by far the lowest p-value, indicating that a true Hawkes model is a much better fit to the data than a simple clustering model. The Chlamydia also fit to the data despite the clearly less obvious amount of contagion and clear outbreaks in the data. Finally, the hypothesis test did suggest that there is causal clustering within the adolescent suicide data, as the hypothesis test was

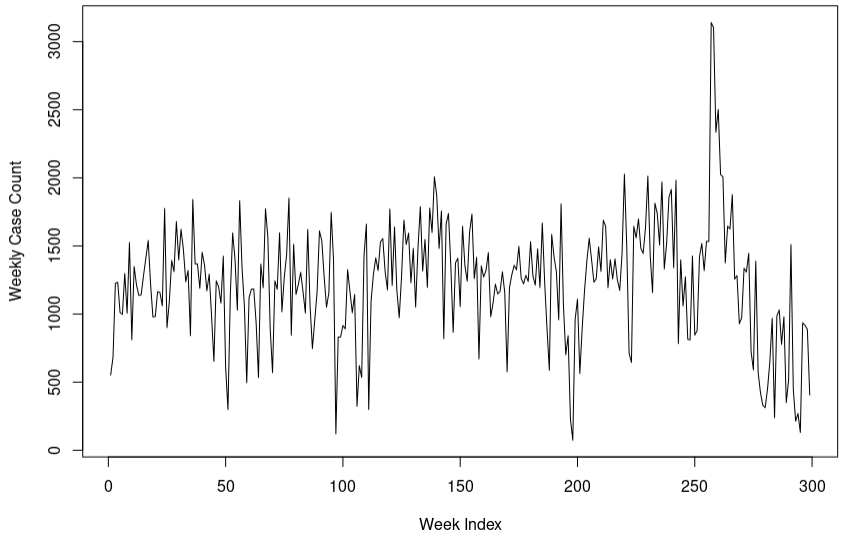


Figure 3.4: *Weekly reported case counts for Chlamydia Cases in California, from 2008-01-01 to 2011-31-12*

rejected. However, this simply means that the Hawkes model fit better than a clustering model to the data set, and not necessarily that there is epidemic contagion, as will be explored in the analysis of the residuals as well as the analysis of the estimated κ parameter.

The estimated κ values are also as expected for the epidemic diseases with a known method of contagion. Firstly, the value of the κ in measles is the highest, with an estimated value of .973. This would indicate that each person, on average, infects .973 other people. Hawkes models usually specify that the value of κ cannot be over 1, as this would mean that the number of points that each initial infection would trigger would be infinite. This corresponds to the infection level of measles, which has a history of intense epidemic outbreaks and can spread via aerosol methods.

The value of κ for chlamydia was estimated as .58. This is a reasonable value for chlamydia, especially in comparison to the measles value. It is unreasonable to expect a high number of infections from each individual for several reasons. Firstly, there will be some

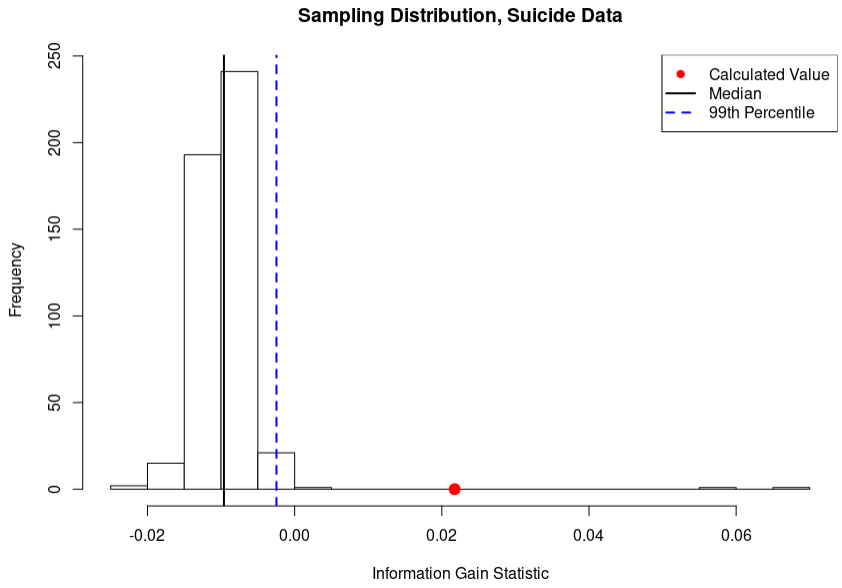


Figure 3.5: *Log-likelihood test statistic and simulated null distribution for adolescent suicide data in the United States from 2018-01-01 to 2021-31-12*

level of data that is not collected from people who are unaware of their infection. In addition, the method of contagion, sexual contact, results in each person infecting fewer number of people, on average.

The value of κ for Lyme, while still estimated in this case, is technically irrelevant as the hypothesis test does not confirm that there is any epidemic spread of the disease. Therefore, the value of κ should not be considered interpretable, although it is still notable that level of spread is still less than measles and chlamydia.

Finally, for the adolescent suicide data, the value of κ is actually not significantly different than 0 according to the 99% confidence interval. This is notable, as this would indicate that the level of clustering, whether it is causal or not, is not significant enough to warrant calling this data a Hawkes model. The reasoning for this is explored further in the analysis of the residuals, and steps for a two-step process of checking for any significant level of clustering will also be discussed.

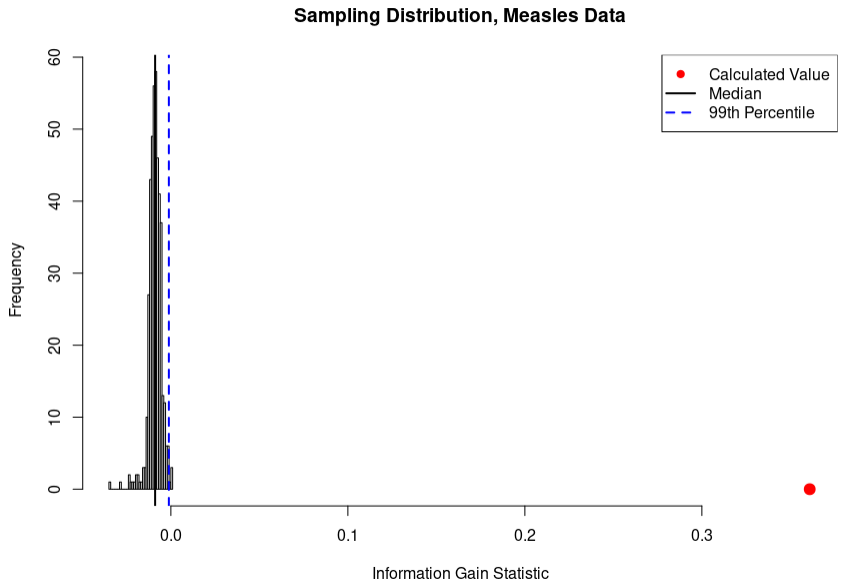


Figure 3.6: *Log-likelihood test statistic and simulated null distribution for measles cases in Los Angeles County, from 1928-01-01 to 1931-31-12*

3.6 Fit and Residual Analysis

The fit and residuals for each of these datasets show many insights into how the two different models fit to the dataset as well as potential explanations for the reasoning behind the κ values and the hypothesis test results. Starting with measles, as seen in Figure 3.10, the clustering model does not do well with estimating the conditional intensity for the several large spikes of diseases, especially between the 100 and 150 weeks of data. Meanwhile, the Hawkes model is much better suited at fitting to the data-while there is a small amount of lag within the estimation that is common in Hawkes models, the large spikes of infection are well modeled. The difference between the Poisson clustering process and the Hawkes process is the largest of any of the data analyzed, which can also be seen in the residuals in Figure 3.14.

For the chlamydia data, as seen in Figure 3.12, the Hawkes model again is better able to handle the larger outbreaks that occur, especially in the ending part of the dataset. The

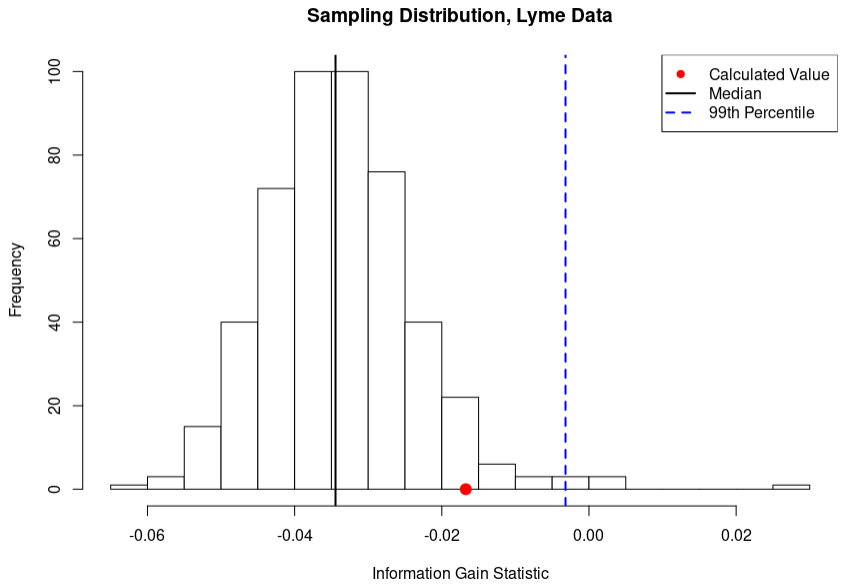


Figure 3.7: *Log-likelihood test statistic and simulated null distribution for Lyme cases in California, from 2008-01-01 to 2011-31-12*

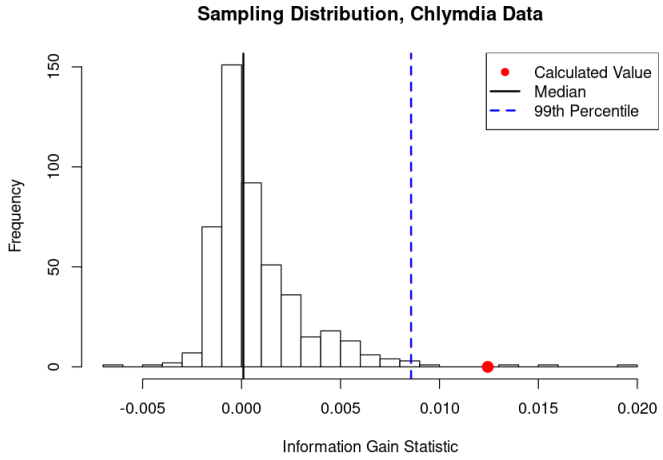


Figure 3.8: *Log-likelihood test statistic and simulated null distribution for Chlamydia Cases in California, from 2008-01-01 to 2011-31-12*

difference is not as pronounced as in the measles data, which is reasonable as there are not as large of outbreaks in the chlamydia data-the period of larger outbreaks do not explode

	κ(99% CI)	p-value of Hypothesis Test
Measles	(0.954 0.993)	.000
Chlamydia	(.569, .591)	.006
Lyme Disease	(.429,.720)	.061
Adolescent Suicide	(-.012,.269)	.004

Table 3.1: *Results of Tests for Clustering and Causal Clustering*

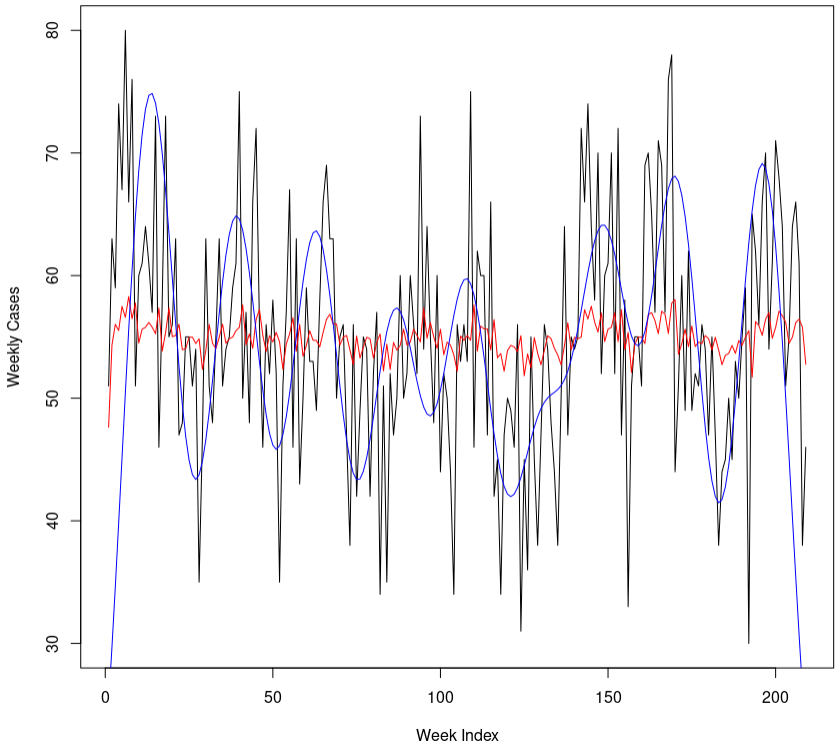


Figure 3.9: *Projected case counts in fitted Hawkes (red) and Poisson clustering (blue) models for adolescent suicide data in the United States from 2018-01-01 to 2021-31-12*

nearly as high as the measles data. In Figure 3.16, the difference in residuals between the two models also is not as pronounced as compared to measles. This explains the finding that the level of contagion in the chlamydia data was lower as well as the test statistic having a

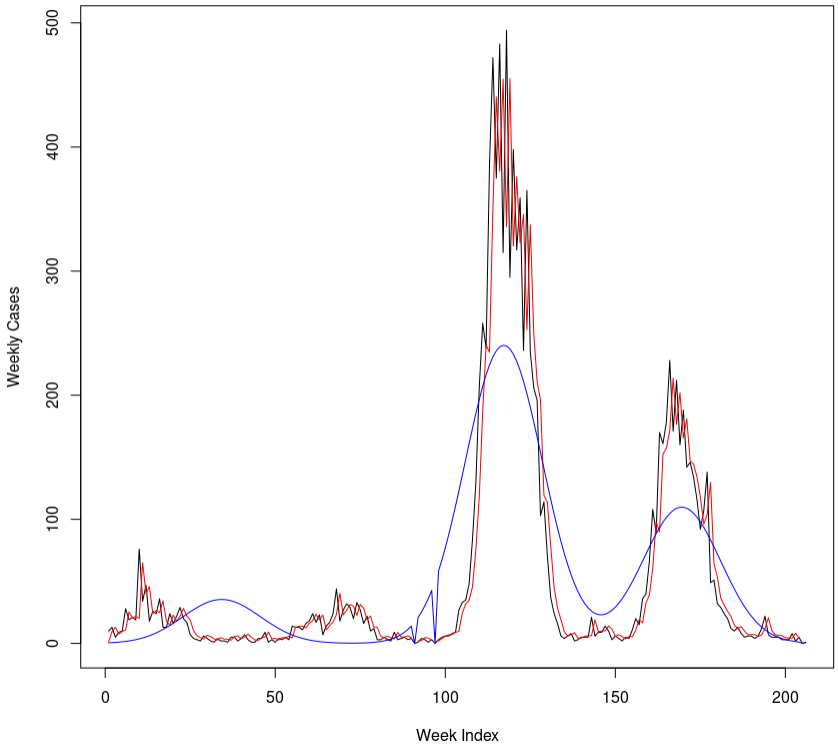


Figure 3.10: *Projected case counts in fitted Hawkes (red) and Poisson clustering (blue) models for measles cases in Los Angeles County, from 1928-01-01 to 1931-31-12*

lower value than the measles test statistic.

For the Lyme data, the last dataset where the contagion method is known, the hypothesis test was not able to reject the null. Looking at the fit of the two models in Figure 3.11, there is little difference between the two models. Both the Hawkes and the Poisson clustering model follow similar patterns throughout the data, and in cases where there is a larger spike in cases, those are typically one-off values that neither the Hawkes or the Poisson clustering model can handle well. In addition, as seen in Figure 3.15, both models have very similar patterns in the residuals, which makes sense as to why the hypothesis test failed to reject the null.

Finally, the adolescent suicide data analysis of fit reveals a potential reason as to the

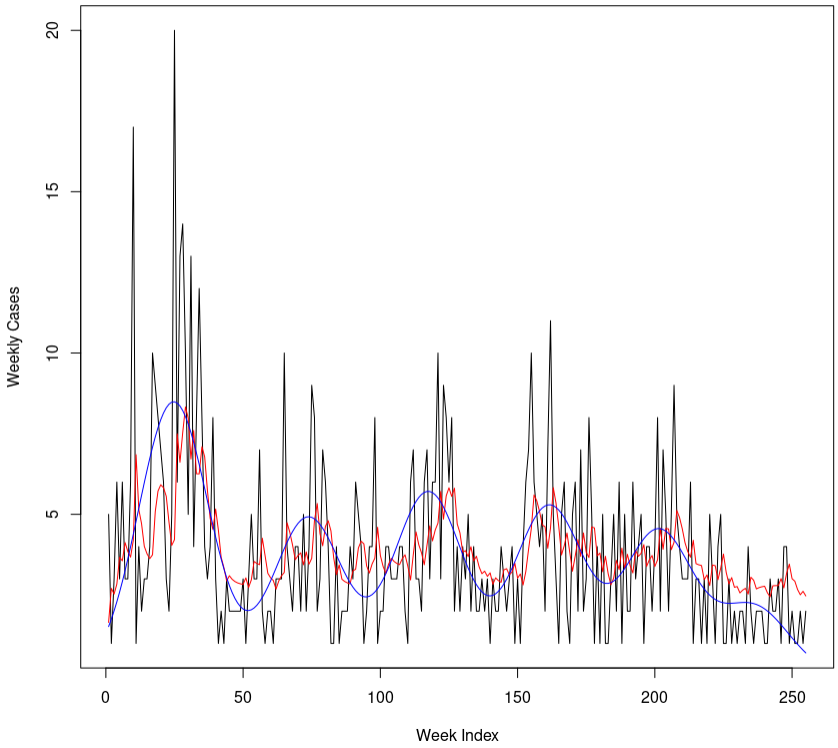


Figure 3.11: *Projected case counts in fitted Hawkes (red) and Poisson clustering (blue) models for Lyme cases in California, from 2008-01-01 to 2011-31-12*

rejection of the null hypothesis. As seen in Figure 3.9, the Hawkes and Poisson clustering models have very different fits. The Poisson clustering model seems to almost over fit to the rise and fall in cases that happen throughout the time period. However, the Hawkes model stays nearly constant throughout the entire time period. This is also reflected in the value of κ, which is incredibly low, and has a confidence interval that contains a value of 0. The residual analysis, seen in Figure 3.13 is less clear, with both models not fitting incredibly well to the data and having residuals that seem to correspond to the data.

The fact that the Hawkes model fits better for the adolescent suicide data therefore does not necessarily mean that there is causal clustering present, especially under the hypotheses that were eventually adapted for this model. In fact, the level of causal clustering still seems

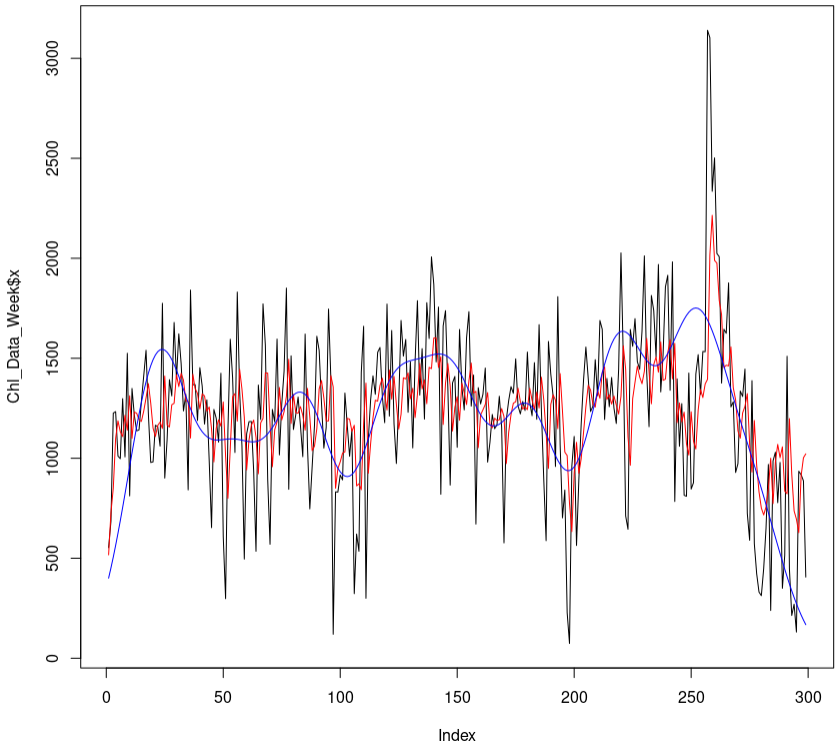


Figure 3.12: *Projected case counts in fitted Hawkes (red) and Poisson clustering (blue) models for Chlamydia Cases in California, from 2008-01-01 to 2011-31-12*

to be 0, since the level of significant clustering seems to be equal to 0. The Hawkes model fits better as essentially a Poisson model with a constant rate, which is just better estimated by a Hawkes model with a low κ value that a Poisson clustering model where there must be clustering present to fit.

Therefore, this analysis seems to indicate that it is necessary to add an additional level of analysis for claims of causal clustering in that the value of κ must also be significant in the Hawkes model to indicate there is causal clustering within the data. Otherwise, the null hypothesis, which is that any clustering present is caused my inhomogeneity, still holds. In this case, the level of clustering is just very small, meaning that the level of causal clustering is still 0.

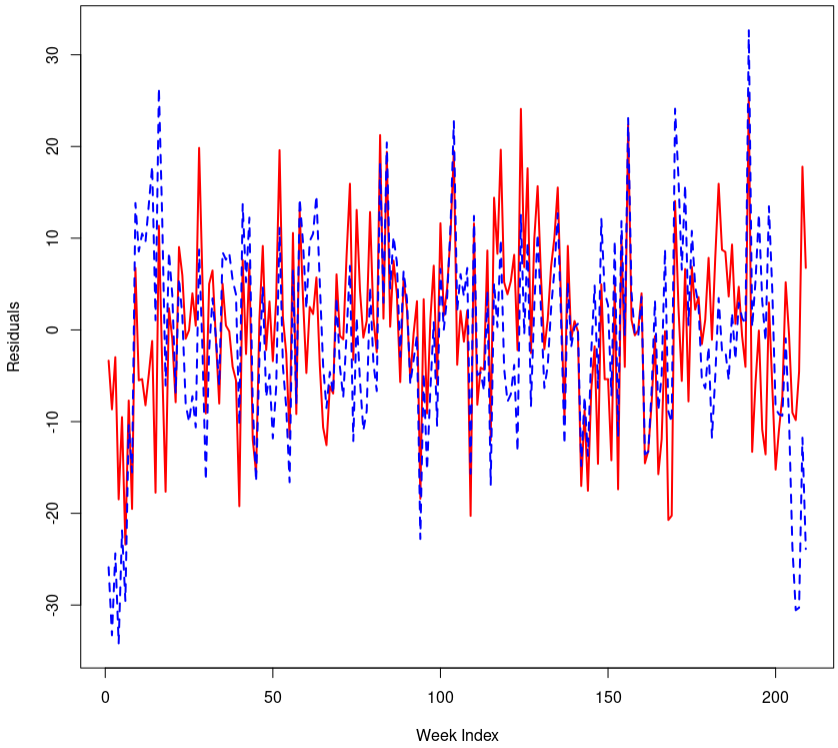


Figure 3.13: *Comparison of Residuals of projected case counts in fitted Hawkes (red) and Poisson cluster (blue) models for adolescent suicide data in the United States from 2018-01-01 to 2021-31-12*

This analysis does not, by any means, rule out the possibility that there is social contagion of suicide among adolescents. There are many possibilities that involve social contagion of suicide of adolescents that would still result in not observing significant causal clustering within the data. There could simply be a small enough amount of contagion that the data collected is not enough to have a value of κ that is statistically significant. In addition, this data was done at the national level and binned by weeks. Contagion in adolescent suicide could be more present on a local level and therefore more observable with that level of granular data. What this analysis does attempt to convey is that adolescent suicide data does not behave similarly on a national scale to diseases with known contagion methods.

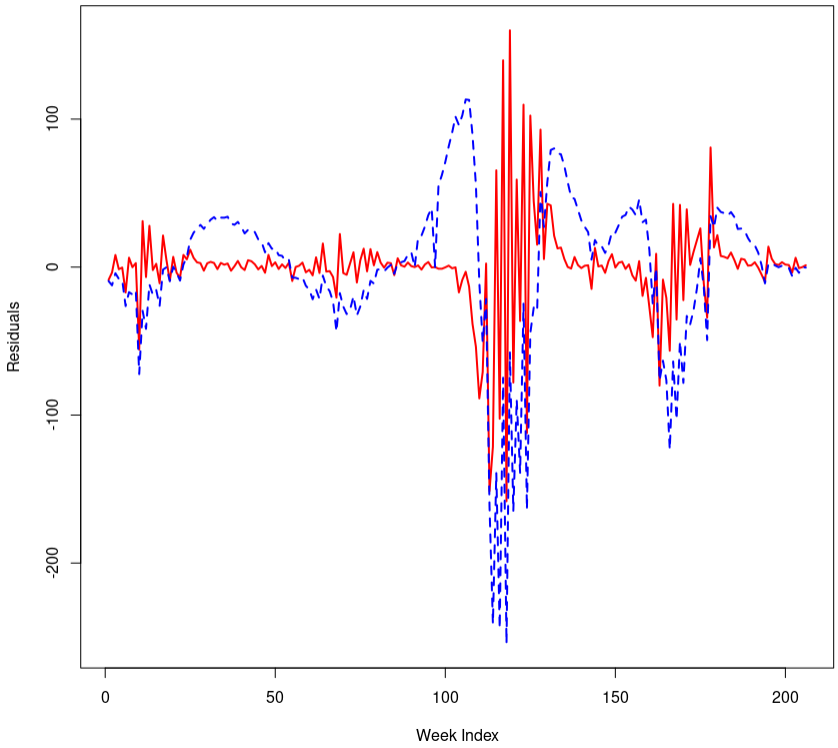


Figure 3.14: *Comparison of Residuals of projected case counts in fitted Hawkes (red) and Poisson cluster (blue) models for measles cases in Los Angeles County, from 1928-01-01 to 1931-31-12*

This does not necessarily contradict the analysis on suicide contagion, which typically looks at areas as small as towns or high schools to indicate when some level of social contagion has occurred.

This analysis attempted to answer three primary questions. Firstly, with regards to whether this proposed test can accurately distinguish an infectious disease from a non-infectious disease, this test was able to correctly distinguish between the infectious nature of measles and Chlamydia and the non-infectious but clustered behavior of Lyme. This indicates that the method is not fooled by diseases that are clustered but not contagious. In addition, for the question regarding the degree of contagion, the measles data had both the

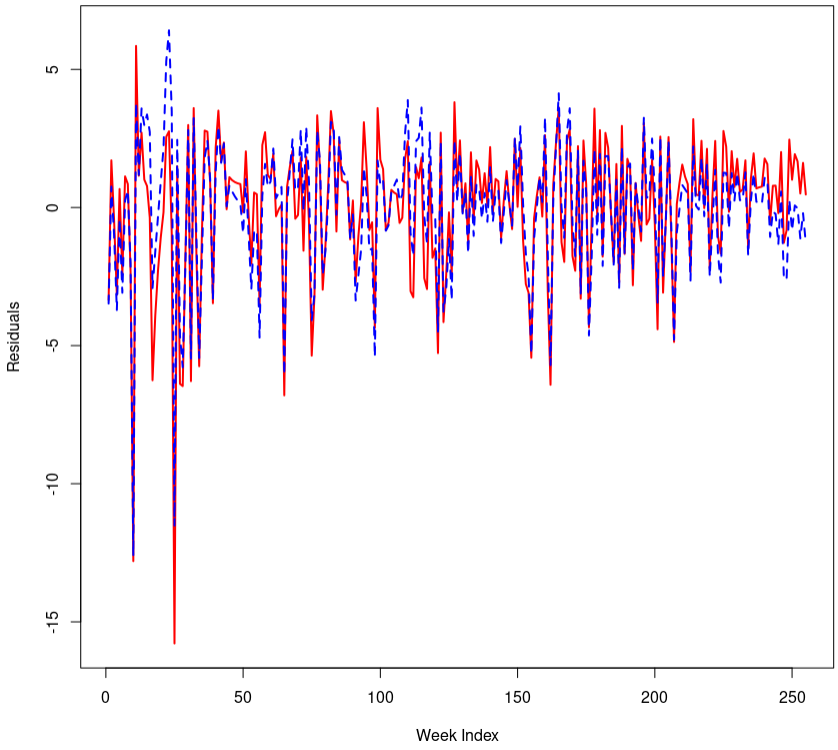


Figure 3.15: *Comparison of Residuals of projected case counts in fitted Hawkes (red) and Poisson cluster (blue) models for Lyme cases in California, from 2008-01-01 to 2011-31-12*

highest κ value as well as the lower p-value when compared to the Chlamydia data, which is consistent with the level of contagion between the two diseases. The test did not suggest that suicide is indeed an epidemic with significant contagion, however it does not necessarily reject the claim that youth suicide could have some level of social contagion that is too small to pick up on aggregate weekly data over the whole United States.

Further analysis for this would involve looking at different diseases to see how this hypothesis test performs, as well as looking at disease data that is spatio-temporal or has a more granular temporal range in order to increase the power of the tests and provide more accurate results.

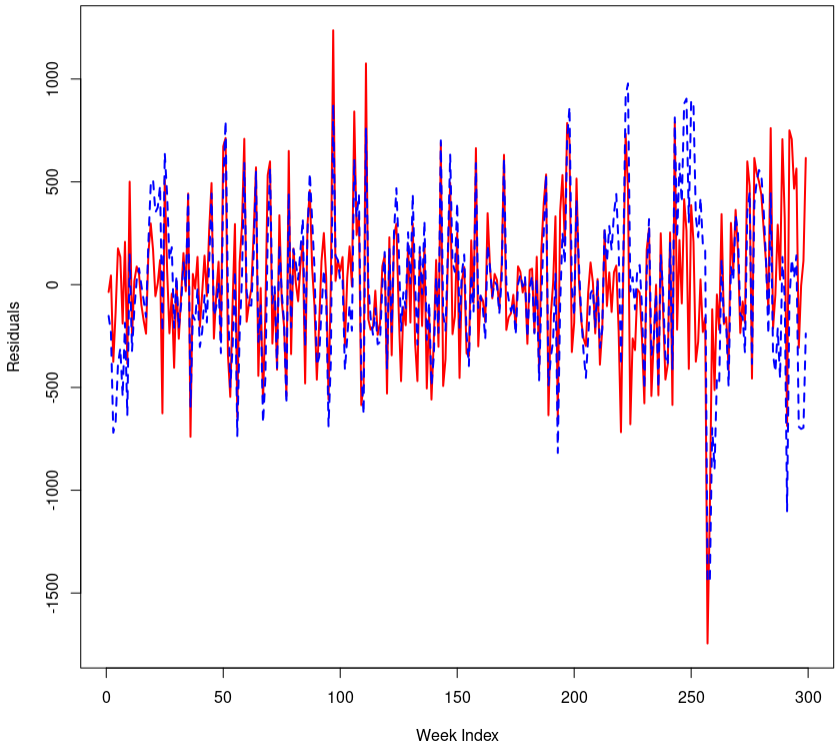


Figure 3.16: *Comparison of Residuals of projected case counts in fitted Hawkes (red) and Poisson cluster (blue) models for Chlamydia Cases in California, from 2008-01-01 to 2011-31-12*

3.7 Further Research and Literature Review: Neural Network Based Analysis

As with most areas of statistics, there has also been a recent analysis in using neural networks in Hawkes process literature. There is a potential to use these methods for an extension of this analysis. Specifically, the extension would be to allow for more complicated triggering functions with the data that are based on a neural network architecture, and to compare that to a neural network architecture that is not based on a Hawkes process to compare the fit.

As discussed previously, a limitation of this analysis is that the initial step of choosing a form for the triggering function can potentially be highly influential to the outcome of the hypothesis test. Since looking over every type of triggering function and optimizing it is not feasible for this test, a potential solution is to use neural network architecture. As most of the Hawkes process neural network architecture has been focused on temporal data, this topic will be explored in this chapter in the context of temporal data.

One way to do this is to use a recurrent neural network to model the intensity based off of a non-linear function of history(Du et al., 2016). A recurrent neural network is a type of neural network that is frequently used for temporal data, as it has a built in structure that allows for some understanding of "time." The recurrent neural network structure involves a

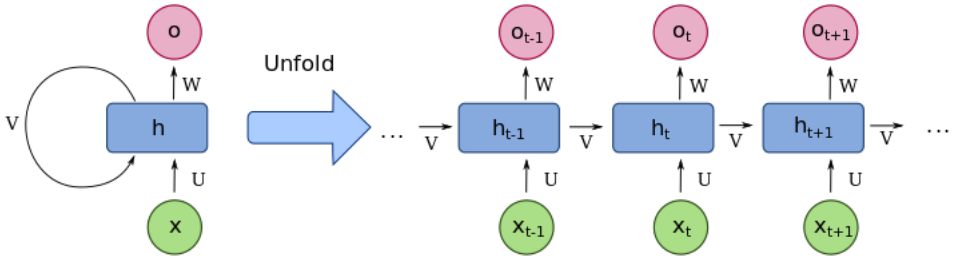


Figure 3.17: *Recurrent Neural Network: Source(fdeloche, 2013)*

series of inputs, $x_1, \ldots, x_p$ that are fed into a series of nodes, $h_1, \ldots, h_p$. At each step there is also the option of an output, denoted as $o_1, \ldots, o_p$. Another way they can be thought of is as marks in a point processes. Marked point processes are used for a variety of purposes, in which each point has not only a "location" within the observation window but a mark attached. An example would be using a marked point processes to model earthquake data, where the marks would refer to the magnitude of the earthquake.

In the case of a marked point process, the inputs x_t can be considered to be both the time y_t as well as the marker m_t for each point in the point processes. The hidden process h_j would represent the learned history for predicting the marker of the event from a non-linear

history function.

there is been several forms of recurrent neural networks that have also been popularized, including the LSTM model, which is essentially the same form of a recurrent neural network but has a more complicated function surrounding the hidden state cells. For a recurrent neural network, the hidden state at time t, h_t, is given as a nonlinear function of the inputs and the previous hidden state, represented as

$$h_t = n(W_{t1}x_t + W_{t2}h_{t-1} + b_t)$$

where n is some non-linear function, W_{t1} and W_{t2} are weights associated with the inputs of x_t and h_{t-1}, respectively, and b_t is some additional baseline constant. Meanwhile, LSTM, standing for Long Short-Term Memory, allows for a more complicated saving of information from the previous history. As the name implies, it is supposed to represent both a more "short term" memory of the history as well as a "long term" memory of the history to create better predictions. Figure 3.18 shows what a hidden cell looks like for the LSTM

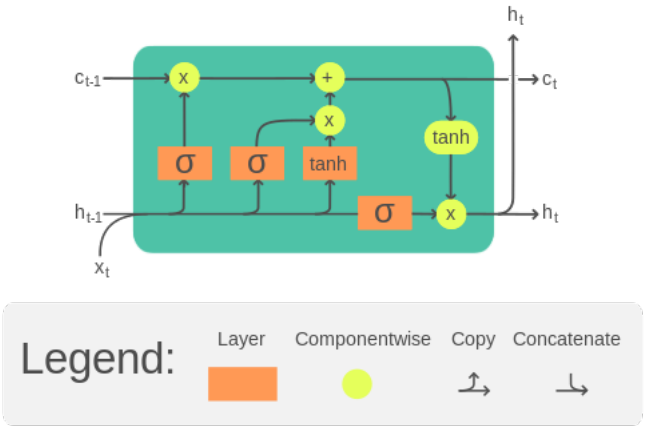


Figure 3.18: *LSTM Neural Network: Source(Chevalier, 2018)*

model. The LSTM model has also been used for modeling the intensity of point process models(Xiao et al., 2017). This allows for a better way to store the longer term effects of points in triggering additional points, which is something that is harder to train in a

more standard Recurrent Neural Network but is not unreasonable in the idea of a Hawkes process(namely, it is possible for the triggering function to allow for triggering significantly in the future).

There is been additional work in the application of Hawkes processes using machine learning techniques. There is also been work on a multivariate Hawkes process using neural networks(Mei and Eisner, 2017). This was slightly different than the previous methods in that the state is updated discontinuously, and the event occurrence also evolves continuously. This method allows for the future events to either be inhibited, or decreased in probability, or excited, or increased in probability, from previous events. This is a feature of a neural network that is slightly harder to do in Hawkes models. It is possible to have a triggering function theoretically that involves inhibition of points, although these models are not very frequently used.

In fact, the generality of these methods has only increased from there. Even more general methods of looking at point processes through a machine learning framework have been explored recently as well. Researchers have explored using even more general methods of calculating the intensity function as the derivative of a generic Neural Network(Omi et al., 2020) as well as using Reinforcement Learning to model the intensity function(Li et al., 2020). These methods are very helpful in estimation and prediction of future points, since the neural network architecture allows for highly complex and non-linear relationships between different points in how they trigger or inhibit future points.

However, as has been explored in this dissertation, it is necessary to point out that these methods should be cautioned against in the case where the triggering method is unknown to the researcher. Since these methods often allow for even more complex triggering functions than most standard Hawkes models, it is reasonable to assume that any data that is clustered would fit well to these types of Neural Hawkes models. This provides even more of an issue for distinguishing whether the clustering is truly causal or not. It is possible that a similar hypothesis test could be performed using a type of Neural Hawkes model and a Neural

Network based clustering algorithm that would allow for similarly non-linear and complex triggering functions both forwards and backwards in time.

3.8 Conclusion

Social contagion is a phenomenon that, as previously discussed, is not typically looked at through a statistical lens. The arguments for social contagion really come from the field of psychology, in which the reasonableness come from more of an analysis of how humans interact with each other.

Similar to the analysis of using Hawkes models on spatio-temporal data and then using that as evidence for some level of causal clustering, there should also be caution in assuming social contagion based off of clustering of behaviors or what appear to be increases of behaviors in certain groups, and assume this is due to social contagion. Again, this comes down to the assumption of causality that is built into a contagion model. As shown in this analysis, there can be clustering that is non-causal that is possible to distinguish between causal clustering. A social contagion model, as this is attempting to show a causal link between points, should not be simply assumed if the model fits well to Hawkes data, and in addition this type of model should be looked at as having the burden of proof of other causal methods.

There is a lot of future research that can be explored in this realm. In general, it would be helpful to get more granular data on certain data in which social contagion is often assumed to be occurring. The granularity can be either in having more specific times for the data, having the data over a certain smaller region, or having spatio-temporal information to the data. Since for many instances, data on psychological issues, especially for those of minors, is relatively confidential, this was not explored in this analysis. However, the field of social contagion is rather broad, so hopefully there is some social contagion data that has more granular data that can be studied in this way.

Other diseases could also be explored with this method, as the method of transmission in infectious diseases is often known to us directly through physical understanding of the disease process. Therefore, with additional information this method could be further tested on other diseases as well as compared t

CHAPTER 4

Formal Causal Framework for Point Process Data

4.1 Background on Causality Research

The Rubin Causal Model, named for Donald Rubin(Holland, 1986), is a causality model that is also sometimes referred to as the "Potential Outcomes" framework. The potential outcomes framework begins with a population of units to which a treatment may be applied. Each unit, Y_i, can either be exposed to the treatment or control. The assignment to treatment is coded as a variable τ that can take a value of 0 or 1(Neal, 2020). The value $Y_i(1)$ represents the value that the unit i would take if exposed to treatment $\tau = 1$, and the value $Y_i(0)$ represents the value that the unit i would take if exposed to treatment $\tau = 0$.

Ideally, from these values the individual treatment effect(ITE), denoted as δ_i can be calculated as

$$\delta_i = Y_i(1) - Y_i(0)$$

Therefore, if the ITE can be calculated for every unit in a population, a perfect estimate for the causal effect of a treatment can be calculated. However, the issue in that is the "Fundamental Problem of Causal Inference"(Holland, 1986), which is that it is impossible to observe both $Y_i(1)$ and $Y_i(0)$ for any unit. For example,in a drug trial, calculating the ITE would require collecting data from the same unit that both did and did not take the medicine at the same time.

Therefore, it is necessary to add additional assumptions in order to perform causal inference on a population using the potential outcomes framework. While the ITE is not

necessarily possible to calculate, the average treatment effect, or ATE, can be under certain assumptions. Some of the assumptions that will be discussed(Neal, 2020), are as follows:

1. Ignorability: $(Y(1), Y(0)) \perp\!\!\!\perp \tau$, or the potential outcome values are independent of the treatment variable, τ.

2. Exchangeability: $\mathbb{E}[Y(1)|\tau = 0] = \mathbb{E}[Y(1)|\tau = 1]$, or that the group assigned to $\tau = 1$ would have the same potential outcome values if that group had been assigned to $\tau = 0$.

3. No Interference: Any unit's treatment is unaffected by any other unit's treatment, or $Y_i(\tau_1, ..., \tau_{i-1}, \tau_i, ...\tau_n) = Y_i(\tau_i)$.

4. Consistency: If a unit is assigned to treatment $\tau = t$, then the observed outcome Y is necessarily the potential outcome under the treatment t, or $\tau = t \rightarrow Y = Y(t)$.

Under these assumptions, the ATE can be calculated as

$$ATE = \mathbb{E}[Y(1) - Y(0)] = \mathbb{E}[Y(1)] - \mathbb{E}[Y(0)] = \mathbb{E}[Y|\tau = 1] - \mathbb{E}[Y|\tau = 0]$$

Since $\mathbb{E}[Y|\tau = 1]$, which is the observed average outcome for units that received treatment $\tau = 1$ and $\mathbb{E}[Y|\tau = 0]$, the observed average outcome for units that received treatment $\tau = 0$ are both values that are observed, the ATE is therefore possible to compute with data as long as the proper assumptions have been met.

The Structural Causal Model(SCM) framework is a framework for causality first described by Judea Pearl(Pearl, 2009). This framework is intended to be a more complete framework of causality that contains elements of the potential outcomes framework, do-calculus, and causality diagrams. The ideas in structural causal models(also called structural equation models) are typically expressed through both causality diagrams and an equations framework.

The diagram in Figure 1 may also be represented as a series of structural equations that take into account what variables each variable relies upon based upon the diagram(Pearl,

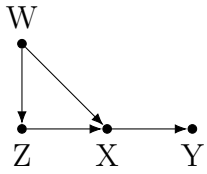


Figure 4.1: *Example of a Causality Diagram(also referred to as DAG under certain conditions)*

2010). For Figure 1, the corresponding structural equations could be represented as follows:

$$w = f_W(u_w)$$
$$z = f_Z(u_z, w)$$
$$x = f_X(u_x, w, z)$$
$$y = f_Y(u_y, x)$$

In each of these equations, the u_i variable present is the (typically undrawn) unobserved variables that also affect the I variable.

While there is extensive theory that focuses on this method of causality, for the purposes of this paper, the potential outcomes model will be focused on much more than this model. However, it should be noted that the Rubin Potential Outcomes framework is simply a special case of Pearl's model(Pearl, 2010), so this could be an area in which causality in point processes is explored much more in the future.

Most preceding research in causality for point process data has been on studying Granger causality, for example looking at Granger causality in Hawkes models(Xu et al., 2016), Neural Spiking Data(Kim et al., 2011b), and image data(Prabhakar et al., 2010). Granger causality,first introduced in 1969(Granger, 1969b), is a method that involves two times series, X_t and Y_t. It is said that X_t "Granger causes" Y_t if Y_t can be predicted significantly better with the information in X_t as opposed to only including information within Y_t.

It should be noted that while this seems to consist of a lot of the causality research within

point process and temporal data, Granger causality is not strictly a causal relationship as it is typically described. There is not evidence that X_t actually affected Y_t in any way, simply that the information contained within X_t is helpful to determine the future of Y_t. An example of this that would potentially show Granger causality but not a more traditional causality definition would be looking at temperature and crime rates-while higher temperatures typically occurs along with higher crime rates, the typical interpretation of this is not that the temperature is causing crime rates.

The idea of Granger causality is helpful in that temporal associations are considered, and if it is possible to control one of the variables, using this as a true causality measure would be better suited. However, overall despite the depth of research in this topic, Granger causality is not considered causality for the point of this paper and will not be discussed.

4.2 Causality as Applied to Spatial and Spatio-temporal Data

Using spatial data for causality research has some unique complications compared to traditional data. Spatial data differs from much other data that is used in causality in that the "units" in spatial data will typically be areas that are connected, while "units" in other forms of data are typically individuals that receive individual treatments. In addition, spatial data can have treatments, covariates, errors, or outcomes that are spatially correlated(Akbari et al., 2023)(Gao et al., 2022)(Reich et al., 2021). Each of these possible spatial correlation situations(or combinations of situations) can lead to unique challenges in the methods typically used for causal research in non-spatial data.

An issue when the treatment variable is spatially distributed is the "spill-over" effect-meaning that the treatment assignment in one location often affects the outcome variable in another location(Akbari et al., 2023). For example, if one was attempting to figure out the causal effect of a minimum wage increase in a certain town on rent prices, you would need to assume that the minimum wage increase only affects rental prices within the town, even

though it could affect rent prices outside the town. This clearly violates the "No Interference" assumption that is necessary for the potential outcome framework.

Another issue for traditional causality methods in spatial data occurs when the causal effect of a treatment varies spatially(Akbari et al., 2023). For example, one could attempt to do a study where the outcome variable is the amount of a specific pollution particle within the air, and the treatment variable is a policy that limits the use of a certain chemical fertilizer in the area. If one treatment region is an agricultural area and one treatment region is an urban area, then the causal effect of this policy could be different in the two areas. This violates SUTVA, since there is multiple versions of treatment.

Another issue in causality in purely spatial data is the lack of a time variable, which greatly complicates distinguishing the direction of cause and effects(Gao et al., 2022). This issue is prevented when we have temporal or spatio-temporal data, but with purely spatial data, the direction of causal structures are very difficult to determine. In addition, spatial data is typically going to consist of observational data, since in most scenarios assigning treatment to random spatial regions is not feasible. Because of this, common issues with causality in observational such as selection bias, confounding, and omitted variables are frequently present in spatial data.

To summarize, a causal framework involving potential outcomes for point process models needs to both consider several key factors. Firstly, what a unit is under a point process paradigm must be established. Following that, the formal definition of a potential outcome for that unit and how that will be measured also needs to be established. Next, there needs to be discussion on what assumptions are necessary for the treatment process in order to get accurate results for the data, as well as what assumptions need to be made for how the units interact with each other in a spatial framework. This chapter lays out a foundation for these questions as well as provides simulations to show how this process works.

4.3 Framework for Causality in Point Processes

The framework that is developed here provides key theoretical properties and allows for a discussion of the potential outcomes framework for point process data. For a point process that is observed on a window $\mathcal{X}$ consisting of a spatial window of $\mathcal{S}$, a subset of $\mathbb{R}^2$ and temporal window of $[0, T]$ such that T is a positive real number. The treatment, τ, is applied a time $t^* \in [0, T]$ in some regions of $\mathcal{S}$, while other regions are untreated. $\mathcal{X}$ can be partitioned into p cells $\mathcal{I}_1, \ldots, \mathcal{I}_p$ that from a partition of the spatial domain $\mathcal{S}$ such that $\bigcup_i^p \mathcal{I}_i = \mathcal{S}$. Each cell is either assigned to treatment or control at time t^*, creating an index of cells that have been assigned to treatment as some subset of $1, \ldots, p$. This provides a clear answer for what the "units" are when referencing point process data, in that each cell is an example of a unit that receives either the treatment or control assignment. In addition, this specifies what "treatment" means in the application to a point process.

The point process observed in this window can be thought of as two separate point processes that have been combined in a particular way. The point process Φ consists of a control process, Φ_c and a treatment point process, Φ_t with separate observation windows such that the union of their observation windows is $\mathcal{X}$. Each cell $\mathcal{I}_j$ is assigned a treatment condition, denoted by the index variable z_j. The value of z_j determines whether the cell has been assigned to the treatment condition, which would mean that $z_j = 1$, or the control condition, which would mean that $z_j = 0$.

The control process Φ_c is assumed to be observed on the full spatial domain until the time of the application of the treatment condition t^*, as well as after t^* in the cells that have been assigned to the control condition. The treatment process Φ_t is observed in the cells that have been assigned to the treatment condition after time t^*. In mathematical terms, Φ_c is observed on

$$\mathcal{S} \times [0, t^*] \cup \left(\bigcup_{j: z_j = 0} \mathcal{I}_j \times [t^*, T] \right).$$

Meanwhile, the treatment process is observed on

$$\bigcup_{j:z_j=1} \mathcal{I}_j \times [t^*, T].$$

This framework is based on the potential outcome framework, so it is necessary to define what a potential outcome means in a point process setting. The "units" in this analysis refer to a cell within the partition of the point process, represented by $\mathcal{I}_j$. The potential outcome of $\mathcal{I}_j$, $Y_{\mathcal{I}_j}(z_j)$ with assigned treatment of z_j would be given as

$$Y_{\mathcal{I}_j}(z_j) = N(\mathcal{I}_j \times (t^*, T]|z_j).$$

As defined previously, $N()$ refers to the counting measure of the point process of the given observation window. This is approximately equal to the integral over this observation window, given by the integral

$$\int_{\mathcal{I}_j} \int_{t^*}^{T} dN.$$

The individual treatment effect in this case(ITE) for cell $\mathcal{I}_j$ would be calculated as

$$Y_{\mathcal{I}_j}(z_j = 1) - Y_{\mathcal{I}_j}(z_j = 0)$$

which is the number of points that would be observed in the cell $\mathcal{I}_j$ after the treatment time t^* under the treatment condition subtracted by the number of points that would be observed in the cell $\mathcal{I}_j$ after the treatment time t^* under the control condition.

For the average treatment affect, since there is some level of discrepancy that can arise due t the size of the individual cells $\mathcal{I}_j$ of the created partition of $\mathcal{X}$, the average needs to be weighted over the size of the space. Therefore, the average treatment affect for simulations will focus on cases in which the control and treatment areas are equal in size, which in that case would mean that the average treatment effect would be given as

$$\tau = \sum_{j=1}^{p} \frac{Y_{\mathcal{I}_j}(1) - Y_{\mathcal{I}_j}(0)}{p}.$$

Note that just like in the case of standard causal analysis, the τ is not observable under this paradigm, as this would require observing both potential outcome values for a given cell. Only one of $Y_{\mathcal{I}_j}(1)$ or $Y_{\mathcal{I}_j}(0)$ is observed for each cell.

4.4 Synthetic Values for Estimation of the ITE/ATE

The way that the fundamental problem of causal inference is dealt with in these paradigm is by creating synthetic values for the unobserved potential outcome values. Without loss of generality, assume that the cell $\mathcal{I}_j$ has a treatment assignment vector value of $z_j = 1$. The potential outcome $Y_{\mathcal{I}_j}(1)$ can be calculated as the number of points within the cell $\mathcal{I}_j$ after time t^*. The synthetic or estimated value for $Y_{\mathcal{I}_j}(0)$ is denoted as $\hat{Y}_{\mathcal{I}_j}(0)$. Let $\hat{\lambda}_c$ be an estimated conditional intensity of the control process ϕ_c with conditional intensity of λ_c. $\hat{\lambda}_c$ is assumed to be calculated with a maximum likelihood estimation method. Then the synthetic value is given as

$$\hat{Y}_{\mathcal{I}_j}(0) = \mathbb{E}[\hat{Y}_{\mathcal{I}_j}(0)] = \int_{\mathcal{I}_j} \int_{t^*}^{T} \hat{\lambda}_c d\mu.$$

Given that the value of λ_c is calculated correctly in the region $\mathcal{I}_j$ and the time $(t^*, T]$, then this would provide an estimate of the counterfactual for the number of points that would have been observed had cell $\mathcal{I}_j$ been assigned to control. The next section will cover the necessary assumptions about the treatment and control processes that would allow for this counterfactual to be a reasonable estimation.

Assuming that these have been estimated correctly, then noting the tower property and linearity of the expectation operator,

$$\begin{aligned}
\mathbb{E}[\hat{\tau}] =& \mathbb{E}\left[\sum_{j=1}^{p} \frac{\left(Y_{\mathcal{I}_j}(1)\mathbb{I}_{z_j} + \mathbb{E}[Y_{\mathcal{I}_j}(1)](1-\mathbb{I}_{z_j})\right) - \left(Y_{\mathcal{I}_j}(0)(1-\mathbb{I}_{z_j}) + \mathbb{E}[Y_{\mathcal{I}_j}(0)]\mathbb{I}_{z_j}\right)}{p}\right] \\
=& \frac{1}{p}\sum_{j=1}^{p} \mathbb{E}\left[\left(Y_{\mathcal{I}_j}(1)\mathbb{I}_{z_j} + \mathbb{E}[Y_{\mathcal{I}_j}(1)](1-\mathbb{I}_{z_j})\right) - \left(Y_{\mathcal{I}_j}(0)(1-\mathbb{I}_{z_j}) + \mathbb{E}[Y_{\mathcal{I}_j}(0)]\mathbb{I}_{z_j}\right)\right] \\
=& \frac{1}{p}\sum_{j=1}^{p} \mathbb{E}\left[Y_{\mathcal{I}_j}(1) - Y_{\mathcal{I}_j}(0)\right] = \tau
\end{aligned}$$

where the last equality is given by the accuracy of the synthetic data assumption. This means that the estimator in this case is an unbiased estimator for τ. For Hawkes processes,

again assuming that the intensities are accurately calculated, this means that

$$\begin{aligned}
\mathbb{E}[\hat{\tau}] =& \frac{1}{p}\sum_{j=1}^{p}\int_{\mathcal{I}_j}\left(\hat{\lambda}_{\mathcal{T}}-\hat{\lambda}_{\mathcal{C}}\right)d\mu \\
=& \frac{1}{p}\sum_{j=1}^{p}\int_{\mathcal{I}_j}\left(\hat{\kappa}_{\mathcal{T}}\sum_{t<t'}\hat{f}-\hat{\kappa}_{\mathcal{C}}\sum_{t<t'}\hat{g}\right)d\mu \\
=& \frac{1}{p}\sum_{j=1}^{p}\int_{\mathcal{I}_j}\sum_{t<t'}\left(\hat{\kappa}_{\mathcal{T}}\hat{f}-\hat{\kappa}_{\mathcal{C}}\hat{g}\right)d\mu \\
=& \frac{1}{p}\sum_{j=1}^{p}|\mathcal{I}_j|\sum_{t<t'}\left(\hat{\kappa}_{\mathcal{T}}\hat{f}-\hat{\kappa}_{\mathcal{C}}\hat{g}\right).
\end{aligned}$$

4.5 Necessary Assumptions as Applied To Point Processes

The no interference assumption means that the potential outcome of any unit is unaffected by any other unit's treatment assignment. In terms of this framework, it would mean that $Y_{\mathcal{I}_j}(z_1,\dots z_p) = Y_{\mathcal{I}_j}(z_j)$. The consistency assumption requires that the observed counting measure in a cell is the same as the potential outcome of that cell for the treatment that was assigned. In other words, $Y_{\mathcal{I}_j}(z_j) = Y_{\mathcal{I}_j}$ if $\mathcal{I}_j$ received treatment z_j.

Suppose that the control process ϕ_c and the treatment process ϕ_t are both Poisson processes such that their conditional intensities are functions of the spatial dimensions. Let $\lambda_c = f(x,y)$ and $\lambda_t = g(x,y)$. Let $\mathcal{I}_j$ be a cell in the partition of $\mathcal{X}$. If $\mathcal{I}_j$ has a treatment assignment value of z_j, then due to the nature of a Poisson process, the potential outcome of $\mathcal{I}_j$ under treatment assignment value z_j will not depend on any z_i such that $i \neq j$. Therefore, under this paradigm, the no interference and consistency assumptions will be met.

These assumptions can be potentially broken without special considerations, given the type of point process. For example, in a Hawkes process, it is possible for points in one cell to trigger points occurring in another cell. This phenomenon is often called "spill-over" within the point process paradigm. If there is any level of spill-over, then the potential outcome of cell $\mathcal{I}_j$ which is given by the integral of the counting measure over $\mathcal{I}_J$ and the time $(t^*, T]$,

can be affected by the treatment assignment of another cell. For a simple example, suppose that ϕ_c is a process with a conditional intensity of $\lambda_c = 0$. Suppose that ϕ_t is a Hawkes process with an intensity

$$\lambda_t(x, y, t) = \mu(x, y) + \kappa \sum_{i:t_i<t} g(x - x_i, y - y_i, t - t_i)$$

such that μ and κ are non-zero. If all cells are assigned to the control condition, then for a specific cell $\mathcal{I}_j$, the value of $Y_{\mathcal{I}_j}(0)$ would be equal to 0, since the conditional intensity over the whole observation window $\mathcal{X}$ is 0. Now, assume that all cells of index w such that $w \neq j$ are assigned to the treatment condition. Then if a point within one of the cells $\mathcal{I}_w$ triggers a point to occur within $\mathcal{I}_j$, the potential outcome of $\mathcal{I}_j$ under the control condition is now equal to 1. Therefore, we have that

$$Y_{\mathcal{I}_j}(0) \neq Y_{\mathcal{I}_j}(1, 1, \ldots, 0, \ldots, 1)$$

which breaks the assumption of no interference. The exchangeability assumption, meaning that the expected values for the potential outcomes for cells assigned to treatment would not change had the cells been assigned to control(and vice versa), is not necessary for this assumption, as the potential outcome values are estimated directly for each cell. This assumption is only necessary when there is not an individual potential outcome calculated. However, this assumption would also be violated given this paradigm under certain conditions, specifically in the case of Hawkes processes.

The ignorability assumption means that the potential outcome values are independent of the treatment assignment. We assume that cells are randomly assigned treatment, and that the treatment assignment vector $\mathbf{z} \perp \mathbf{Q}$ for any spatial covariates Q if those are considered.

We need to assume that the covariate-dependent portion of the specified (control/treatment) intensities changes negligibly within the time window $(t^*, T]$, which can be referred to as locality. The final assumption that is necessary to make is that estimated conditional intensity of each process is calculated properly. Using MLE for calculating the parameters

as well as a thinning method described later, this assumption can be reasonably assumed given enough data under certain paradigms.

4.6 Homogeneous Poisson Process Application

Given a partition of the observation window $\mathcal{X}$, it is possible to calculate the average treatment effect τ as the difference between the observed potential outcome value and the synthetic estimated potential outcome value in each cell. Let ϕ_c and ϕ_t both be Poisson processes such that $\lambda_c = g(x, y)$ and $\lambda_t = f(x, y)$. Assuming that each cell is of equal area and the treatment and control sections have the same area, we purpose an estimator of

$$\hat{\tau} = \sum_{j=1}^{p} \frac{\left(Y_{\mathcal{I}_j}(1)\mathbb{I}_{z_j} + \hat{Y}_{\mathcal{I}_j}(1)(1 - \mathbb{I}_{z_j})\right) - \left(Y_{\mathcal{I}_j}(0)(1 - \mathbb{I}_{z_j}) + \hat{Y}_{\mathcal{I}_j}(0)\mathbb{I}_{z_j}\right)}{p}. \tag{4.1}$$

This estimator will be unbiased if an accurate estimation of λ_c and λ_t can be found. As a simple example, the case in which $\lambda_c = \mu_c$ and $\lambda_t = \mu_t$ will be explored.

For the case of λ_c, since the control process is observed on all of the spatial domain up until time t^*, using MLE estimation on λ_c will provide an unbiased estimate of μ_c, which will simply be the number of points observed divided by the length of the temporal domain. The case where the spatial domain is $[5 \times 5]$ and the temporal domain is $[0, 50]$ with $t^* = 50$ and $\lambda_c = 2$, the control process observed on $[0, 25)$ is observed in Figure 4.2 In this case, the MLE estimate of μ_c would simply be the number of points observed divided by 25, so in this case the estimate would be 2.04.

Next, the treatment and control process are observed on $t = [25, 50]$, however since this is after t^*, the treatment process is visible on cells with $z_j = 1$ and the control process is observed on cells with $z_j = 0$. In this case the value of $\mu_t = 20$. As seen in Figure 4.3, the observed process is now split into two separate treatment and control processes. From this, it is possible to estimate μ_t just from the cells that were assigned to the treatment value. Again, it is simply the number of points in those cells divided by 25 and multiplied by 2,to

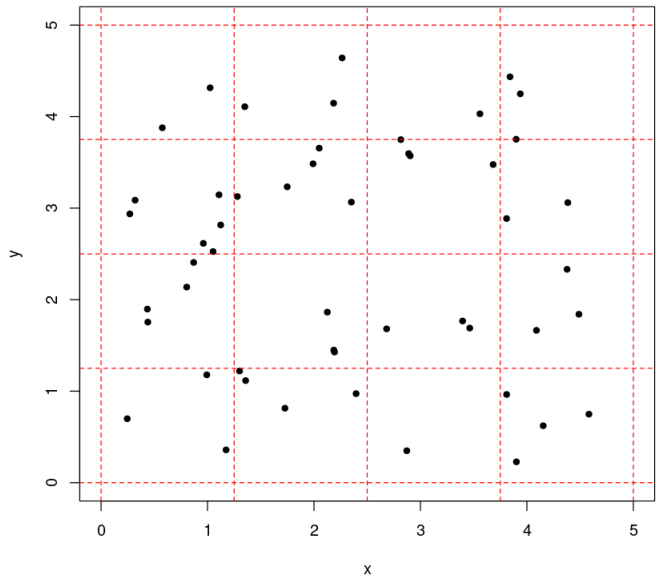


Figure 4.2: *Poisson process observed on* $[5 \times 5] \times [0, 25)$ *with* $\lambda_c = 2$*. The dashed red lines indicate the divisions of cells, with the spatial domain divided into 16 equal sized cells*

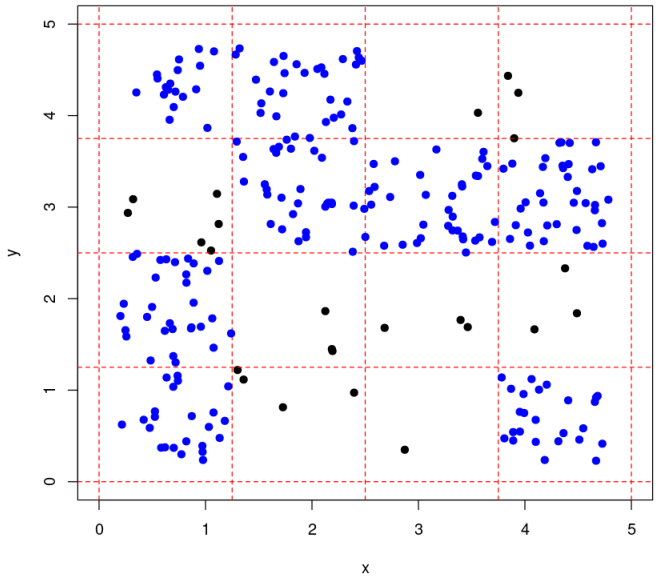


Figure 4.3: *Poisson process observed on* $[5 \times 5] \times [25 \times 50]$ *with* $\lambda_c = 2$ *and* $\lambda_t = 20$. *The dashed red lines indicate the divisions of cells, with the spatial domain divided into 16 equal sized cells. The blue process is the treatment process and the black process is the control process*

account for the fact this was only observed on half of the domain, giving an estimate of $\hat{\mu}_t = 18.24$.

From here, its trivial to calculate the an estimate for the average treatment affect τ. Using the formula previously defined, this would be given by

$$\hat{\tau} = \sum_{j=1}^{16} \frac{(228 + 228) - (24 + 24)}{16} = 25.5$$

In this case, the actual value of τ can be calculated directly as

$$\tau = \sum_{j=1}^{16} \frac{(250 + 250) - (20 + 20)}{16} = 28.75.$$

Obviously this is just one example, however it is clear that this method is essentially just following nearly the same method as looking at a difference in means in an experiment where equal people are assigned to control and treatment and observing the average treatment affect. This problem is quickly just reduced to finding mean values for a counting measure with certain means, which is trivial.

If it can be assumed that λ_c and λ_t are both estimated in an unbiased manner, it is possible to show that our proposed estimator is in fact unbiased. The estimation of λ_t and λ_c gets considerably more complicated when the background rate has some level of spatial variance. So far, there is not a reasonable way to deal with a Poisson process in which the background rate varies spatially, specifically between the treatment and control process. The reason for this is that since the treatment process is only observed on half the spatial domain, finding a reasonable counterfactual for the treatment process on the cells assigned to the control process is nearly impossible without many assumptions.

However, this paper will explore the case where the background rate depends on covariates in a specific manner. This will be explored specifically in a Hawkes model case, however the extension to Poisson processes is trivial, since a Poisson process can be thought of as a Hawkes process with a triggering intensity κ of 0.

4.7 Hawkes Process Application

In order to deal with spill-over effects in a Hawkes process model, an algorithm to accurately estimate both the MLE of λ_c and λ_t values was developed. From here, the estimation of τ can be completed as described previously. This section will also provide examples of this method with simulated data. Assume that the conditional intensity is

$$\lambda_c(x, y, t) = M_c + \kappa_c \sum_{t_i : i < t} g_c(x - x_i, y - y_i, t - t_i | \theta_c)$$

such that θ_c refers to any parameters that the triggering density g_c relies on and the conditional intensity of the treatment process is given as

$$\lambda_t(x, y, t) = M_t + \kappa_t \sum_{t_i : i < t} g_t(x - x_i, y - y_i, t - t_i | \theta_t)$$

such that θ_t are any parameters that the triggering density g_t relies on. For the sake of computation, the following triggering densities are also defined as

$$\alpha_c = \kappa_c \sum_{t_i : i < t} g_c(x - x_i, y - y_i, t - t_i | \theta_c)$$

and

$$\alpha_t = \kappa_t \sum_{t_i : i < t} g_t(x - x_i, y - y_i, t - t_i | \theta_t)$$

Step 1: Calculate the MLE estimates of $\hat{M}_c, \hat{\kappa}_c$ and $\hat{\theta}_c$ using the pre-treatment data. Specifically, in the region of $\mathcal{S} \times [0, t^*)$. Since this spatial-temporal region only contains the control intensity, there is no need to control for spill-over effects. This allows for an accurate and unbiased estimate of the control intensity, $\hat{\lambda}_c$.

Step 2: Thin the post-treatment cells assigned to treatment using $\hat{\lambda}_c$. Specifically in the region of $\bigcup_{z_j=1} \mathcal{I}_j \times (t^*, T]$, keep points with a probability of $\frac{1}{\hat{\alpha}_c}$, which is referred to as thinning. That means that, for the region of $\bigcup_{z_j=1} \mathcal{I}_j \times (t^*, T]$, the conditional intensity of those points will be

$$\lambda = \lambda_t \times \frac{\alpha_c}{\hat{\alpha}_c}$$

which means that if the estimate of $\hat{\lambda}_c$ is unbiased, then

$$\mathbb{E}[\lambda] = \mathbb{E}\left[\lambda_t \times \frac{\alpha_c}{\hat{\alpha}_c}\right] = \mathbb{E}[\lambda_t] = \lambda_t.$$

Step 3: Calculate the MLE estimates of $\hat{M}_t$, $\hat{\kappa}_t$ and $\hat{\theta}_t$ using the thinned points in the post-treatment region assigned to treatment. Since now these thinned points have been thinned of any spill-over effects from the control cells, it is now possible to get an accurate estimate of the parameters for the treatment conditional intensity. Specifically, this will be calculated on the post-thinned points in $\bigcup_{z_j=1} \mathcal{I}_j \times (t^*, T]$.

Step 4: Thin the post-treatment time control cells using the estimated value of $\hat{\lambda}_t$. Specifically, in the region of $\bigcup_{z_j=0} \mathcal{I}_j \times (t^*, T]$ keep points with a probability of $\frac{1}{\hat{\alpha}_t}$. This will deal with any potential spill-over effects from the treatment cells to the control cells. That means that now the value of the conditional intensity in the control region after time t^* is

$$\lambda = \lambda_c \times \frac{\alpha_t}{\hat{\alpha}_t}.$$

After thinning, this now means that

$$\mathbb{E}[\lambda] = \mathbb{E}\left[\lambda_c \times \frac{\alpha_t}{\hat{\alpha}_t}\right] = \mathbb{E}[\lambda_c] = \lambda_c.$$

With the expected value of the conditional intensity for both the treatment and control cells being unbiased estimators for the conditional intensity of the treatment and control processes respectively, the counting measures for both the treatment and control cells are now unbiased. As this is the basis of the potential outcome values, it is now possible to find an accurate estimate of τ.

A simulation is done in order to demonstrate the process of estimating the treatment effect using the thinning process as previously described. This simulation is done on a spatio-temporal region, with the spatial domain on $[0, 5] \times [0, 5]$ and the temporal domain on $[0, 50]$. As described in the assumptions of the process, only the control process, $\lambda_{\mathcal{C}}$ is observed up until the time of treatment $t^* = 25$.

In this case we assume that $\lambda_{\mathcal{C}}$ and $\lambda_{\mathcal{T}}$ are dependent only on $\mathcal{H}_t$ and not on any spatial covariates for the time being. Both the control and treatment triggering densities have exponential densities for the distance(s) and the time(t) dimensions. The triggering densities for the control and treatment process are both of the form

$$g(s-s',t-t') = f(s-s',t-t') = \alpha e^{-\alpha(s-s')} \times \beta e^{-\beta(t-t')}.$$

. In addition the background rate is of the form $M(s|\mathcal{Q}) = \mu$. The parameters for these distributions are as follows:

Parameter	Control	Treatment
κ	.4	.7
α	5	13
β	10	10
μ	10	10

The control process is then fit on the data up until t^*, and then the treatment assignment is applied to cells and the process after t^* is only observed on the cells that have been assigned to control.

Once the process is fit on the control process up until $t = 25$ a shown in Figure 4.4 using a standard maximum likelihood estimation, the control process after $t = 25$ is subset to only the cells that have been randomly assigned to control, as seen in Figure 4.5. The fitted parameters from the control process are then used to thin the treatment process, in order to deal with any spill over effect between two sections of data.

As seen in Figure 4.6, there are some points that have a likelihood of being thinned, as the ratio of the conditional intensity of that point between the control and treatment process is high. This typically occurs on the values that are spatially at a border between a treatment and control cell within the data. However since this is a spatio-temporal process the points must also be temporally distant from other points within the treatment process,

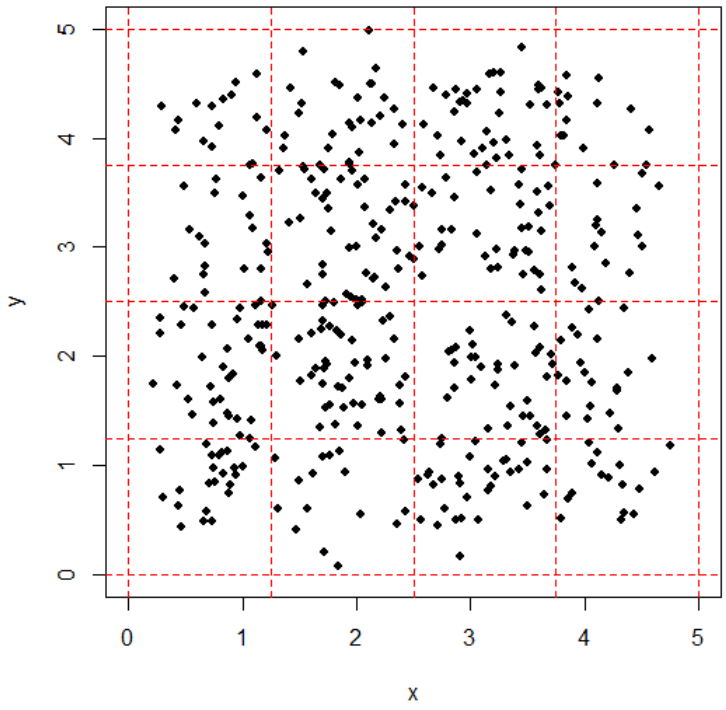


Figure 4.4: *Control Process until* $t = 25$*, with the data shown on all cells, regardless of treatment assignment. This is used to estimate the parameters of the control process.*

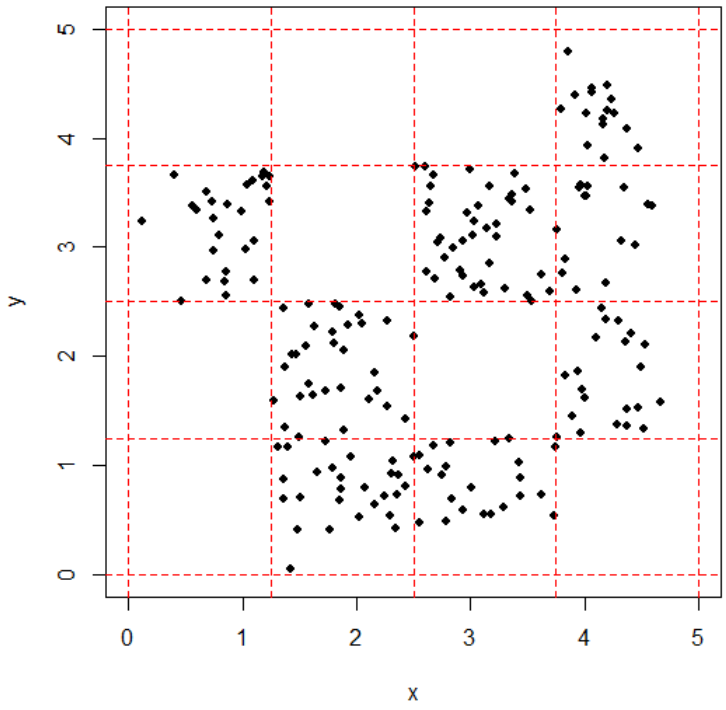


Figure 4.5: *Control process after* $t = 25$*, with only the cells that have been assigned to control observed. This is used to thin the treatment process*

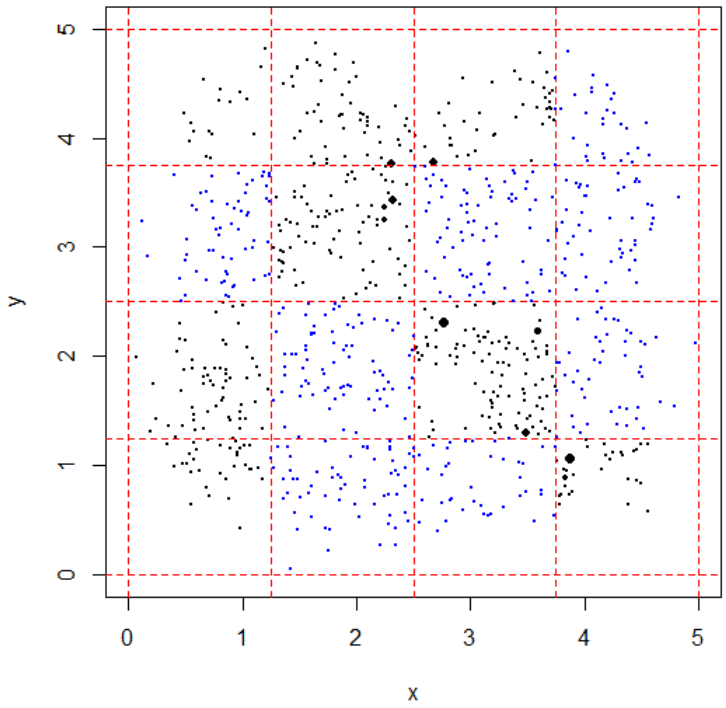


Figure 4.6: *The treatment process and control process after time* $t = 25$ *displayed on the same graph. The blue points are the control process and the black points are the treatment process. The likelihood of thinning is represented by the size of the point*

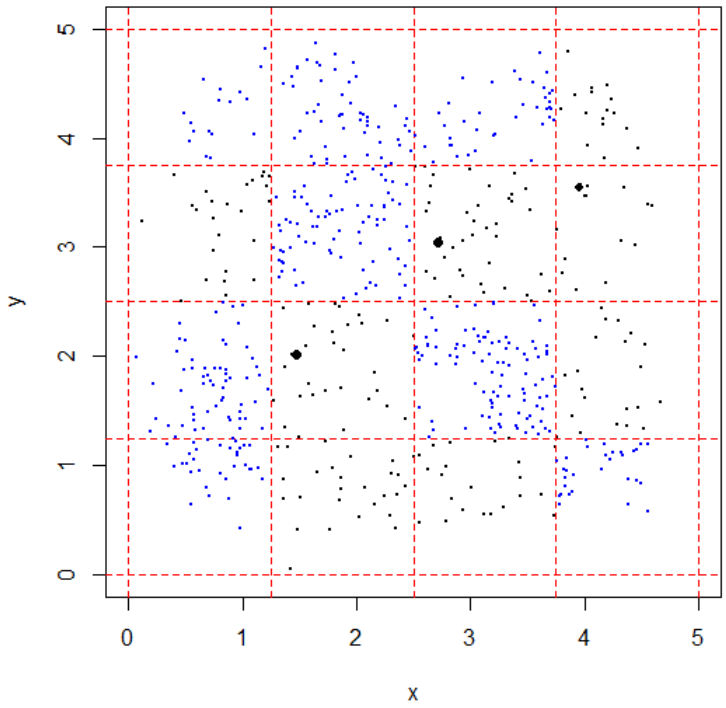


Figure 4.7: *The treatment process and control process after time* $t = 25$ *displayed on the same graph. The blue points are the treatment process and the black points are the control process. The likelihood of thinning is represented by the size of the point*

which explains why some points close to a border between treatment and control are not likely to be thinned.

Once this is done, the parameters for the treatment process are estimated using maximum likelihood estimation. Finally, the control process after time $t = 25$ is thinned using the parameters found for the treatment process, which is visualized in Figure 4.7. From here an estimate for τ can be made by comparing the number of points in the control and treatment process after time $t = 25$, and compared to the expected value of τ. 1000 simulations were done for this method and the resulting histogram can be seen in Figure 4.8. In addition, the same parameters were used with increasing values of T to see if the estimated values converged to the true value as the time increased, and this was found to be accurate as seen in Figure 4.9.

Both of these graphs indicate that this process provides an unbiased and consistent estimate of τ. Figure 4.8 shows that the estimates of τ follows an approximately normal distribution that is centered at the true value of τ for these simulations. In addition, with more data the level of variance becomes smaller, resulting in lower variance in the estimates as seen in Figure 4.9.

4.7.1 Hawkes process with Covariates

In spatio-temporal data, as well as the causality framework, there are very typically variables that must be controlled for in order to achieve accurate estimation. This method can also work when there are covariates that need to be controlled for in the background rate. In this case, the Hawkes process being fitted is of the form

$$\lambda(x, y, t|\mathcal{H}_t) = \mu(x, y) + \kappa \sum_{(x',y',t'):t'<t} g(x - x', y - y', t - t')$$

where the triggering density is exponential in both distance and time as before, and the background rate density is given by

$$\mu(x, y) = e^{-a|W(x,y)|}$$

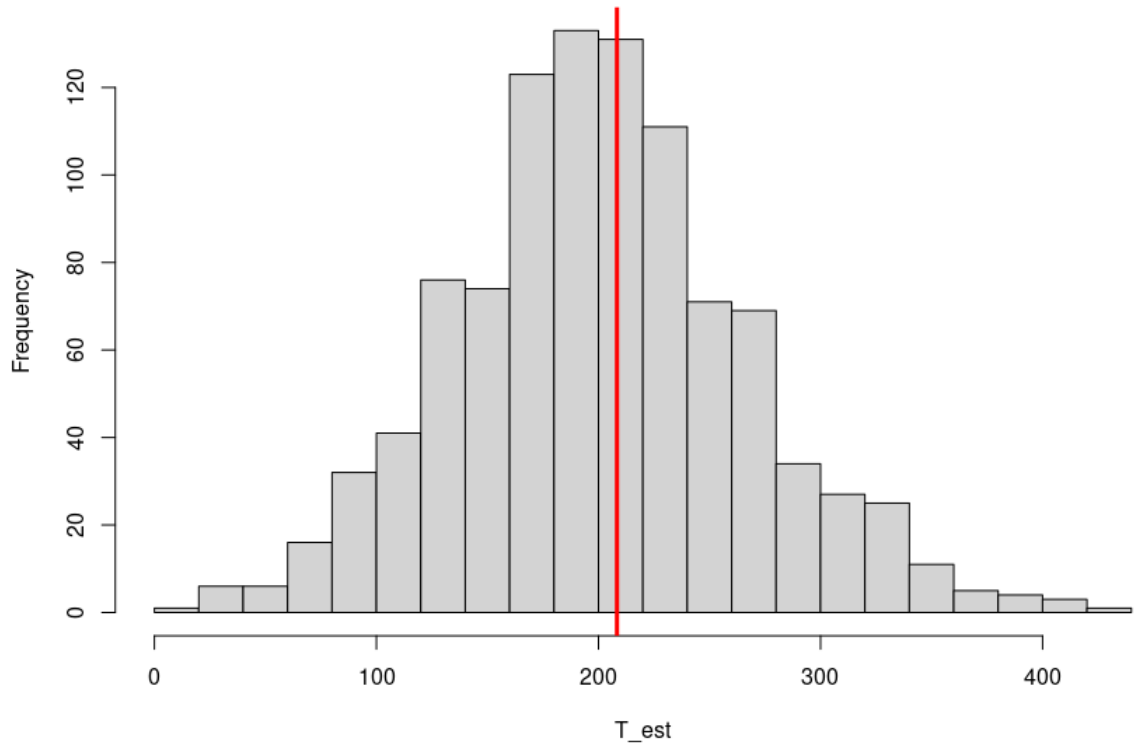


Figure 4.8: *Histogram of estimated τ values, with the true value of τ represented by the vertical line*

where a is a constant value and $W(x, y)$ is the value of the covariate located at the point (x, y). This can be expanded to multiple covariates by adding a linear combination of covariates to the power of the exponential in this background rate function. For the case of this simulation, only one covariate is considered. The value of $W(x, y)$ is simply values proportional to a bivariate normal distribution, on a 4x4 grid, as shown in Figure 4.10. The calculation and simulation process is similar, with the only difference being a varied background rate. The resulting estimate does need to consider the integral of the background rate over the treatment and control areas, which is why there is not a fixed value for the value of τ in every run. The value of τ will change depending on which regions are assigned to treatment and

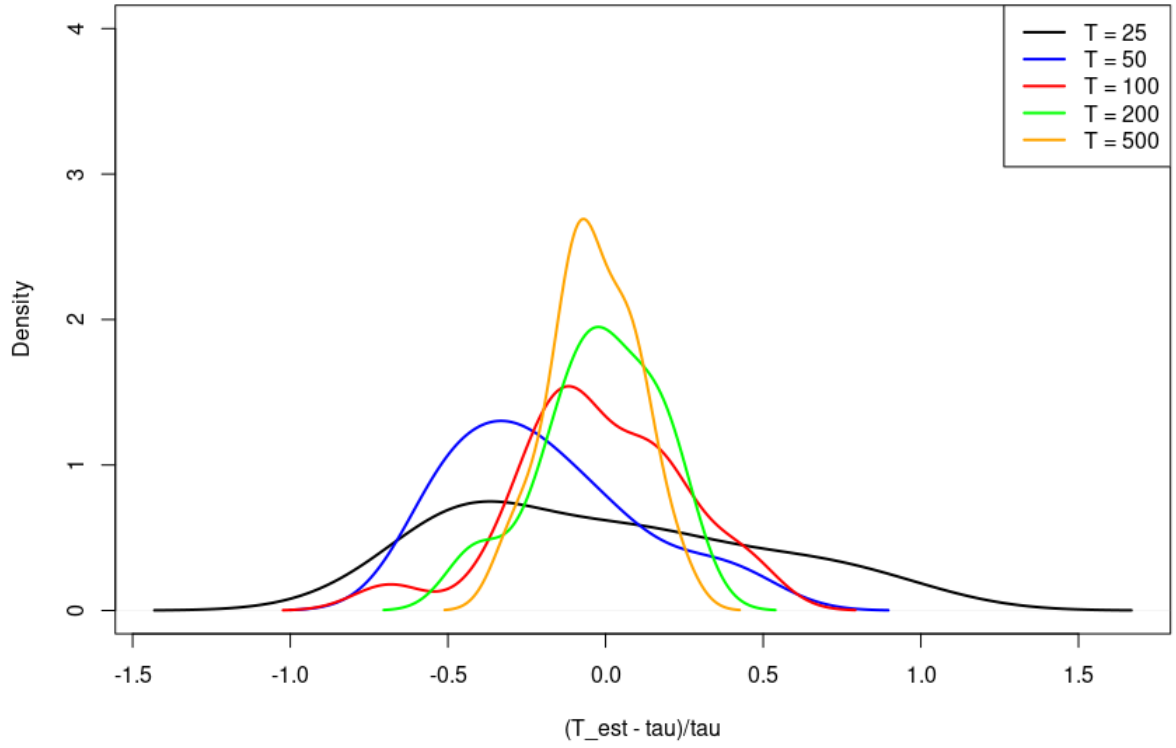


Figure 4.9: *The (scaled) estimates for tau converging to the true value of τ as the time increases-implying that as more data is collected, the estimates become more accurate.*

control, as this will affect the background rate and therefore the expected number of points in that region. In addition this will affect how thinning is done, since only the conditional intensity that is coming from the triggering density should be considered.

The parameters used for this study are found below.

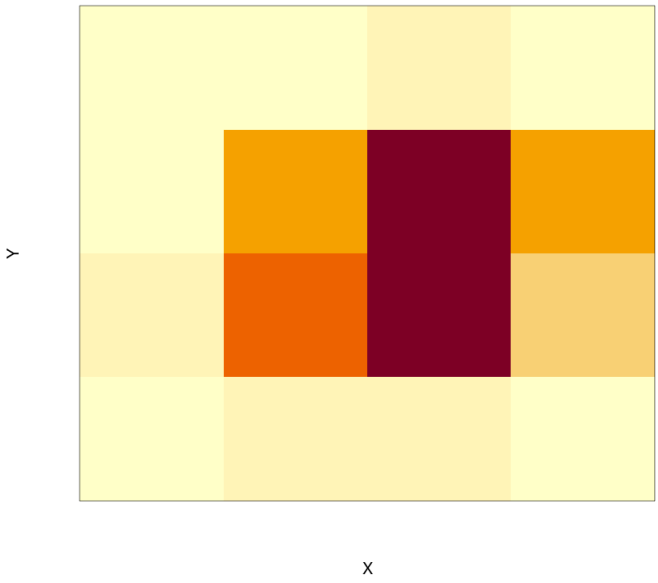


Figure 4.10: *Heat Map of Background rate variable*

Parameter	Control	Treatment
κ	.4	.7
α	5	13
β	10	10
μ	10	10
a	.07	.07

First, just as before, the background process is simulated at the time $t = t^*$, which in this case is 25 following the previous study. The control process until time $t = 25$ is fully observed, although the background μ is now not constant. The control process is shown in Figure 4.11. This allows for an accurate estimate for λ_c, specifically the parameters κ_c, α_c, β_c,μ_c and a_c. Next, the control process is only observed after $t = 25$ on the cells that have been assigned to treatment, as shown in Figure 4.12. The thinning process is shown in Figure 4.14 and Figure 4.15. The captions on the diagrams explain in more detail the steps. For

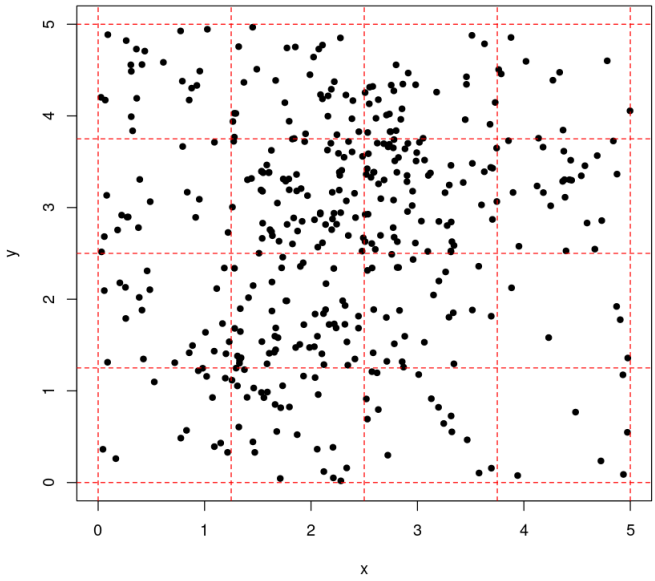


Figure 4.11: *The control process, observed over the entire observation spatial window, and up until time $t = t^*$. The background rate is non-constant, so there is obvious clustering in the middle sections of the grid, while the outer sections have less. This provides the estimate for λ_c.*

this simulation study, 200 simulations were performed for this method, and the results are found in Figure 4.16. Figure 4.16 shows the difference between each τ and $\hat{\tau}$ value that was calculated over the 200 simulations. Overall, it seems to suggest this is getting an unbiased or close to unbiased result, with a distribution that is approximately normal and centered at around 0. There is quite a bit of variation in this data, however this is most likely due to the increased variation that comes from having a non-constant background rate in how many points are present, along with which of the sections are placed into the control or treatment group.

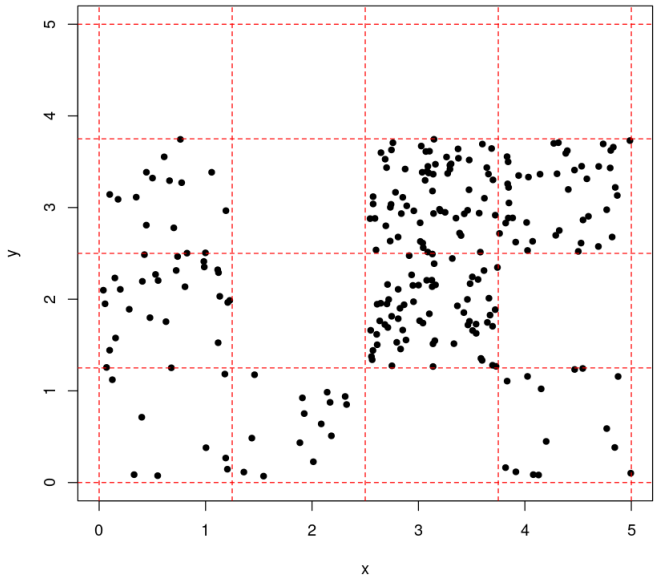


Figure 4.12: *The control process after time $t = t^*$, which is only observed on the sections that have been randomly assigned to control. Since there is already an estimate for λ_c, this data can be used to thin the treatment process in order to get an accurate estimate.*

4.8 Further Study and Applications

While this is limited to only simulation studies, this framework provides a method to deal with causality in a spatio-temporal sense under conditions that are potentially reasonable to observe in the real world. An overview of these conditions will help future work for this method as a way to verify if this is a reasonable method to apply to data. An example based on using this method for a disease outbreak will be given.

Firstly, the process must be observed over the entire observation window before any treatment is applied. This is necessary to get a reasonable estimate for the conditional intensity of the control process. In the context of a disease process, this would mean observing the spatio-temporal locations of infections before a treatment is applied to any group within the region.

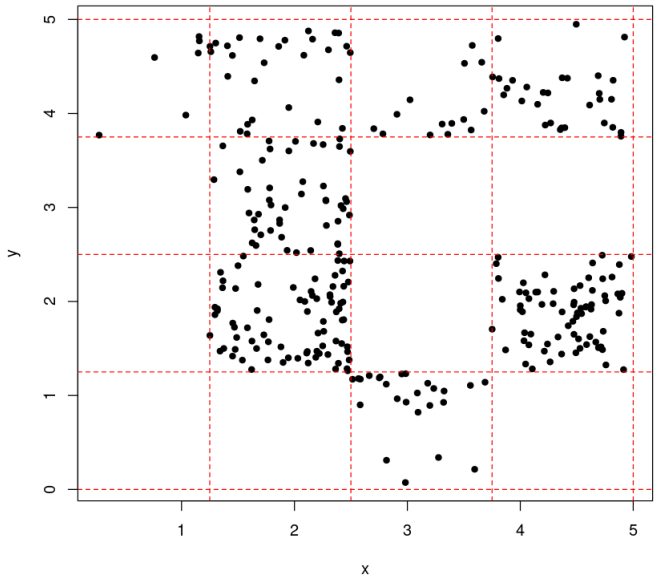


Figure 4.13: *The treatment process after time $t = t^*$, which is only observed on the cells that have been randomly assigned to treatment. Observe that since there is the possibility of spill-over, estimating λ_t from this data could potentially lead to a biased estimate as there could be points that were triggered from the control process in the treatment cells.*

Another condition is that the parameters θ_c and θ_t of the treatment and control process do not vary over time. This would correspond to a disease not changing its κ value over time significantly. In the context of a disease, the treatment would most likely be an intervention that would change the speed of the spread of the disease, κ. This would mean that the intervention should change κ at the time of the intervention, and this should not vary greatly throughout the observation window. For the control process, it is assumed that the value of κ or the amount of spread is relatively consistent throughout the whole observation window.

It is also assumed in this that the treatment and control sections are randomly chosen. This is an assumption that is very difficult to find in actual data. This is often a problem that is found in causal analysis literature. The hope of the example involving covariates is to allow certain values it to allow for the same structure of conditioning on variables

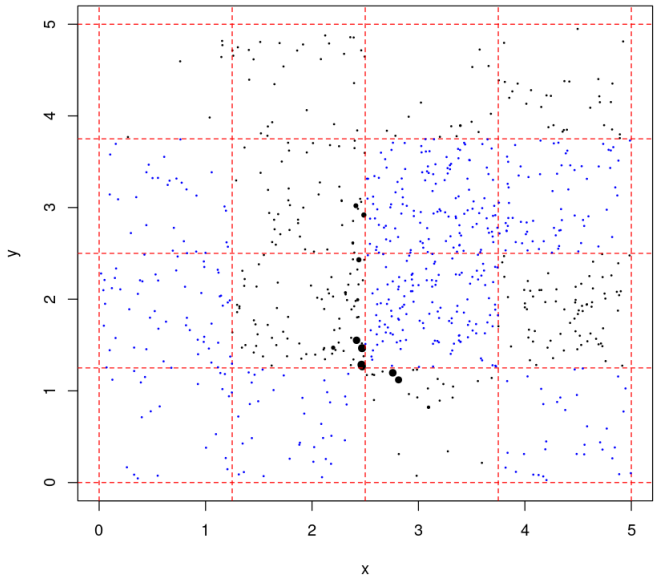


Figure 4.14: *Thinning of the treatment process. The blue points represent the control process, while the black points represent the treatment process. The black points are sized according to the probability that they will be thinned-meaning that the larger points are more likely to be thinned. Notice that this primarily occurs in the region with the highest background rate, which is reasonable as there would be more points in general in that area. The points that might be thinned are also at the edges of the grid sections bordering control grid sections.*

and other methods that are involved with structural equations methods to be used. While conditioning on certain variables to attempt to prove causality is often not theoretically sound, it is possible that conditioning on variables would provide at least a better estimate of a causal effect than if no variables can be conditioned on.

Further study using this framework could also involve looking at other types of point process frameworks beyond just a Hawkes and Poisson model. These models are used fairly often in practice in the point process literature, but other models could be explored. Similar methods to the Hawkes process method could be used for any models in which there is any level of triggering that goes forward in time as well as spatially out of certain points to

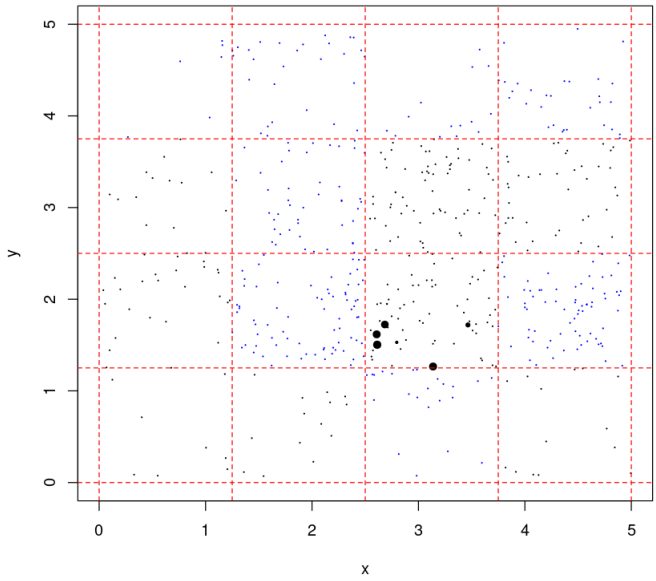


Figure 4.15: *Thinning of the control process. The blue points represent the treatment process, while the black points represent the control process. The black points are sized according to the probability that they will be thinned-meaning that the larger points are more likely to be thinned. Notice that this primarily occurs in the region with the highest background rate, which is reasonable as there would be more points in general in that area. The points that might be thinned are also at the edges of the grid sections bordering treatment grid sections.*

deal with the spill-over effects, and the thinning method would simply have to be slightly adjusted to deal with different types of models.

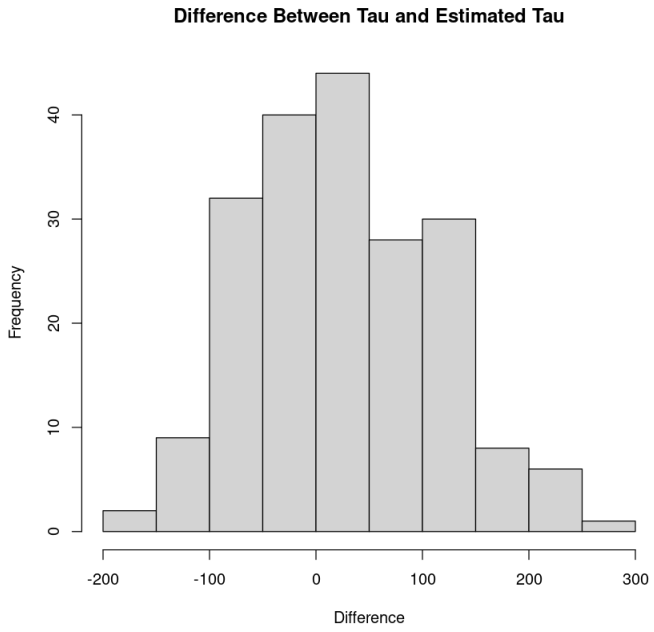


Figure 4.16: *Estimation of τ with a non-constant background rate*

CHAPTER 5

Conclusion

The combination of causal analysis and point processes is a field in which there is relatively little prior research-the applications to each other are fairly common, however, usually these two fields are not looked at in compatible frameworks and methods. This dissertation shows that there are many applications and methods that can help bridge this gap and provide a clearer definition and detection for causality in point process data.

The hypothesis test introduced in this dissertation to find evidence of causal clustering rather than inhomogeneity performed well on the simulated data in the spatio-temporal context. This was then applied to crime data to investigate the contagion theory of crime, which found some evidence to suggest this is reasonable to look at under certain circumstances. In addition, this hypothesis test was applied to temporal data of different infectious diseases, where the results were in line with the expected results. In addition, the contagion theory for adolescent suicide could not be confirmed using this method, although it was not proved negatively by any means.

The potential outcome framework for point processes worked well on simulations and was able to be extended to deal with covariates if available. This framework was proved to provide accurate estimations of the parameters for the treatment and control processes under certain conditions, and to provide a way to deal with spill-over effects in Hawkes processes.

The future of this research should apply not only the methods discussed directly in this dissertation but also to the general philosophy of the stringent proof needed to show causality in the point processes framework. The study of causal inference has become relatively

advanced over the short time it has been looked at in more rigorous ways, so this level of advancement should also be applied and utilized for point process data.

CHAPTER 6

Appendix

6.1 Code for Spatio-temporal Hypothesis Test

```
loglhawk_norm_2 = function(theta,draw=0){
  #print('theta = ')
  #print(theta)
  mu = theta[1]; K = theta[2]; sigma_t = theta[3]; sigma_xy = theta[4]
  #cat("\n mu = ",m3(mu),", K = ",m3(K),", alpha = ",m3(alpha),", beta
    ↪ = ",m3(beta),".\n")
  if(min(mu,K,sigma_t,sigma_xy)<0.000000001) return(99999)
  if(K>.99999) return(99999)
  if(draw){
    r = seq(0,3,length=100)
    t = alpha/pi * exp(-alpha * r^2)
    lines(r,t,col="orange",lty=2)
  }
  sumlog = log(mu/X1/Y1)
  intlam = mu*T + K*z$n
  const = K*1/(sqrt(2*pi)*sigma_t)*(1/(2*pi))*(1/sigma_xy)
  for(j in 2:(z$n)){
    #print(j)
    gij = 0
    for(i in 1:(j-1)){
      #print(i)
      #print(j)
      gij = gij + 1/((1 - pnorm((0 - z$t[i])/sigma_t)))*exp(-1/(2*sigma
        ↪ _t^2)*(z$t[i] - z$t[j])^2 + -1/(2*sigma_xy)*(z$lat[i] - z$
        ↪ lat[j])^2 + -1/(2*sigma_xy)*(z$lon[i] - z$lon[j])^2)
    }
    lamj = mu / X1 / Y1 + const*gij
```

```
    if(is.na(lamj)){
      print('lamj is NA')
      return(99999)
    }
    if(lamj < 0){
      cat("lambda ",j," is less than 0.")
      return(99999)
    }
    sumlog = sumlog + log(lamj)
  }
  loglik = sumlog - intlam
  #cat("loglike is ", loglik, ". sumlog = ", sumlog,". integral = ",
    ↪ intlam,".\n")
  if(draw) lines(r,t,col="white",lty=2)
  return(-1.0*loglik)
}

power_checking_function <- function(mu_in, K_in, theta_t_in, theta_xy_
  ↪ in, x_in, y_in, t_in, M){
  theta_sigma_t <<- theta_t_in
  theta_sigma_xy <<- theta_xy_in
  T <- t_in
  mu <<- mu_in; theta_K <<- K_in; theta_alpha<<-3.5; theta_beta<<-3.5;
    ↪ theta_b<<-1; theta_m0<<-3; theta_a<<-1
  X1 <- x_in
  Y1 <- y_in
  test_hawk_norm <- list('n' = 3)
  while(test_hawk_norm$n < 6){
```

```
  test_hawk_norm = simhawk(x1 = x_in, y1 = y_in, T=t_in ,mdensity =
    ↪ pointmag, gt=normgt, gxy = normxy, gmi = pointprod)
}
#print('we got out of while loop!')
z <<- test_hawk_norm
scatter3D(z$lon, z$lat, z$t)
for_optim = optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_
  ↪ norm_2 )
#print(for_optim$par)
#print(for_optim$value)
z$lon <<- rev(z$lon)
z$lat <<- rev(z$lat)
z$t <<- rev(z$t)
z$t <<- -(z$t - z$t[1])
back_optim = optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_
  ↪ norm_2 )
#print(back_optim$par)
#print(back_optim$value)
z <<- test_hawk_norm
aa <<- Mclust(data = cbind(z$lon, z$lat, z$t), modelNames = 'EEI')

num_clust <- ncol(aa$parameters$mean)
avg_size <- mean(table(aa$classification))
sigma_mat <- aa$parameters$variance$Sigma
#library(mvtnorm)
true_val <- (-for_optim$value - (-back_optim$value))/(test_hawk_norm$
  ↪ n)
#print(num_clust)
#print(avg_size)
```

```
#print(sigma_mat)
sampling_distrib_1 <- c()
for(m in 1:M){
  if(m%%10 == 0) print(m)
  sim_ns <- matrix(data = NA, nrow = 3, ncol = 1)
  #sim_ns <- cbind(sim_ns, t(temp))
  for(i in 1:num_clust){
    temp <- rmvnorm(max(1,rpois(1, avg_size)), c(runif(1, 0, X1),
      ↪ runif(1, 0, Y1), runif(1, 0, T)), sigma = sigma_mat)
    sim_ns <- cbind(sim_ns, t(temp))
  }
  sim_ns <- sim_ns[,-1]

  z$lon <<- sim_ns[1,]
  z$lat <<- sim_ns[2,]
  z$t <<- sim_ns[3,]
  z$n <<- nrow(sim_ns)

  z$lon <<- z$lon[order(z$t)]
  z$lat <<- z$lat[order(z$t)]
  z$t <<- z$t[order(z$t)]

  z$lon <<- z$lon[z$t >0]
  z$lat <<- z$lat[z$t > 0]
  z$t <<- z$t[z$t > 0]
  z$n <<- length(z$t)


  b3 <- optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_norm_2
```

```
      ↪  )
    #print(b3$value)
    #print(b3$par)
    z$lon <<- rev(z$lon)
    z$lat <<- rev(z$lat)
    z$t <<- rev(z$t)
    z$t <<- -(z$t - z$t[1])
    #z$t
    b4 = optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_norm_2
      ↪ )
    #print(b4$value)
    #print(b2$par)
    sampling_distrib_1[m] <- (-(b3$value) - (-b4$value))/(z$n)
  }
  return(list('true_value' = true_val, 'sampling_dist' = sampling_
    ↪ distrib_1))
}

## General ETAS simulator where the user can input densities.
## Have all the parameters defined externally!!!!

simhawk = function(x1=1, y1=1, T=100, rho=unifrho, gt=powergt, gxy=
  ↪ powerxy,
                   gmi=expprod, mdensity=expmag, sor=1, keep=1){
  ##### THIS IS FOR SIMULATING A HAWKES PROCESS WITH
  ##### lambda(t,x,y) = mu rho(x,y) +
  ##### SUM gmi(m_i) gt(t-t_i) gxy(x-xi,y-yi; mi),
  ##### on a space S = [0,x1] x [0,y1] (km), in time [0,T],
  ##### background temporal rate mu and spatial density rho(x,y),
```

```
##### triggering density gt(t-t_i) gxy(x-xi, y-yi; mi),
##### productivity gmi(m_i),
##### and magnitude density mdensity(m).
##### sor = 1 outputs the points in chronological order.
##### keep = 1 means only keep the ones within the space time window.
##### Both gt and gxy must be densities, so that if mu = 1/(x1y1),
##### then the integral of lambda over the space time region = mu T +
    ↪ SUM gmi(m_i).
##### Thus the ETAS parameter K is included in gmi.
##### If no magnitudes are desired, just let gmi = K.
##### mu should be defined externally, along with other parameters
    ↪ used in the functions.
y = bgpts(x1,y1,T,rho, mdensity) ## lay down the background points.
cat(y$n,"mainshocks.\n")
calcbr = 0
calcbr = mean(gmi(mdensity(1000000))) ## calculate branching ratio.
    ↪ Stop if br > 1.
cat("branching ratio is", calcbr,"\n")
if(calcbr>1.0){
  cat("error, branching ratio =", calcbr, "-> 1.")
  return(0)
}
stop1 = 0
if(y$n < 0.5) stop1 = 2
cat("aftershocks by generation\n")
w = y
while(stop1 < 1){
  z = aft(w,x1,y1,T,gt,gxy,gmi,mdensity) ## place aftershocks down
      ↪ around y.
```

```
    cat(z$n,"-")
    if(z$n > 0.5){
      y = combine1(y,z)
      w = z
      if(min(z$t) > T) stop1 = 2
    }
    if(z$n < 0.5) stop1 = 2
  }
  if(keep==1) y = keep1(y,x1,y1,T) ## to keep just the pts in the
     ↪ window.
  if(sor==1) y = sort1(y) ## to have the points sorted chronologically.
  y
}

## br = INT gmi(m) mdensity(m) dm, from m = m0 to infinity.

normgt = function(n){
  ## normal triggering in time with mean gmean and gsd defined
     ↪ externally!
  rnorm(n,mean=gmean,sd=gsd)
}

bgpts = function(x1,y1,T,rho,mdensity){
  ## define mu externally!
  z1 = list()
  n = rpois(1,mu*T)
  z1$n = n
  xy = rho(n,x1,y1)
  z1$lon = xy[,1]
```

```
  z1$lat = xy[,2]
  z1$t = sort(runif(n)*T)
  z1$m = mdensity(n)
  z1$ztimes = c()
  z1
}

aft = function(y,x1,y1,T,gt,gxy,gmi,mdensity){
  ## place aftershocks around y.
  z1 = list()
  z1$t = c()
  z1$n = 0
  z1$m = c()
  z1$lat = c()
  z1$lon = c()
  z1$ztimes = c()
  n2 = gmi(y$m) ## vector of number of aftershocks for each mainshock.
  for(i in 1:length(n2)){
    if(n2[i] > 0.5){
      b1 = gt(n2[i])
      z1$ztimes = c(z1$ztimes, b1)
      z1$t = c(z1$t, b1 + y$t[i])
      xy = gxy(n2[i], y$m[i])
      z1$lon = c(z1$lon, xy[,1] + y$lon[i])
      z1$lat = c(z1$lat, xy[,2] + y$lat[i])
      z1$m = c(z1$m, mdensity(n2[i]))
    }
  }
  z1$n = sum(n2)
```

```
  z1
}

combine1 = function(y,z){
  z1 = list()
  z1$t = c(y$t,z$t)
  z1$n = y$n + z$n
  z1$m = c(y$m,z$m)
  z1$lat = c(y$lat,z$lat)
  z1$lon = c(y$lon,z$lon)
  z1$ztimes = c(y$ztimes, z$ztimes)
  z1
}

keep1 = function(y,x1,y1,T){
  ## keep only the pts of y that are within the space time window [0,x1
     ↪ ] x [0,y1] x [0,T].
  keeps = c(1:length(y$t))[(y$t<T)&(y$lon<x1)&(y$lat<y1)&(y$lon>0)&(y$
     ↪ lat>0)]
  y$t = y$t[keeps]
  y$m = y$m[keeps]
  y$lon = y$lon[keeps]
  y$lat = y$lat[keeps]
  y$n = length(keeps)
  y
}

sort1 = function(y){
  ## sort the pts chronologically.
```

```
  ord2 = order(y$t)
  y$t = y$t[ord2]
  y$m = y$m[ord2]
  y$lon = y$lon[ord2]
  y$lat = y$lat[ord2]
  y
}

## rho takes an integer n and x1 and y1 and outputs a matrix of n
   ↪ locations of mainshocks.
## gt takes an integer n and outputs a vector of n nonnegative times
   ↪ since mainshock.
## gxy takes an integer n and magnitude m and outputs a matrix of n
   ↪ locs from mainshock.
## gmi takes a vector of mags m and outputs a vector of number of
   ↪ aftershocks per mainshock.
## mdensity takes an integer n and lower mag threshold m0 and outputs a
   ↪  vector of n magnitudes.

## Below are examples of functions rho, gt, gxy, gmi, and mdensity.

unifrho = function(n,x1,y1){
  ## Uniform spatial background density rho on [0,x1] x [0,y1].
  x = runif(n,min=0,max=x1)
  y = runif(n,min=0,max=y1)
  cbind(x,y)
}

## density = b e ^ -bm. cdf = 1 - e^-bm. m means m - m0.
```

```
## u = unif(0,1). F(x) = u. 1 - e^-b(m-m0) = u.
## Solve for m.
## e^-b(m-m0) = 1-u.
## -b(m-m0) = log(1-u).
## m - m0 = log(1-u)/-b.
## m = -log(1-u)/b + m0.

expmag = function(n){  ## need theta_b and theta_m0 defined externally!
  -log(1-runif(n))/theta_b + theta_m0
}

pointmag = function(n) rep(0,n)

## expmag = function(n,theta, m0=3.5){
##    ## exponential magnitude density mdensity with minimum m0 and
   ↪ mean m0+b1.
## THIS IS WRONG!!! rexp(n,rate=1/theta$b) + m0
## }

expgt = function(n){ ## need theta_beta defined externally!
  ## exponential triggering function in time gt, with mean beta.
  ## f(u) = beta e^(-beta u).
  rexp(n,rate=theta_beta)
}

normgt = function(n){
  abs(rnorm(n, sd = theta_sigma_t))
}
```

```
powergt = function(n){
  ## power law triggering function in time gt. Define theta_c and theta
     ↪ _p externally!
  ## f(u) = (p-1) c^(p-1) (u+c)^-p.
  v = runif(n)
  theta_c*(1-v)^(1/(1-theta_p)) - theta_c
}

## Notes for powergt.
## if v = runif(1), then new time is found by letting v = F(t) and
   ↪ solving for t.
## F(t) = INT from 0 to t of f(u) du = (p-1) c^(p-1) (u+c)^(1-p) / (1-p
   ↪ )
## from u = 0 to t
## = -c^(p-1) (t+c)^(1-p) + c^(p-1) c^(1-p) = 1 - c^(p-1) (t+c)^(1-p).
## Setting v = 1 - c^(p-1) (t+c)^(1-p) and solving for t, we get
## c^(p-1) (t+c)^(1-p) = 1-v.
## (t+c)^(1-p) = (1-v) c^(1-p).
## t+c = c (1-v)^{1/(1-p)}.
## t = c (1-v)^{1/(1-p)} - c.

powerxy = function(n,m){
  ## define theta_d and theta_q externally!
  ## power law triggering in space according to ETAS (2.3), gxy, of
  ## Ogata (1998). See http://wildfire.stat.ucla.edu/pdflibrary/ogata98
     ↪ .pdf .
  ## Here the density does not depend on magnitude of the mainshock.
  ##   f (x,y)dxdy = 1 =  h (r)rdrd  = 2  h  (r)rdr.
  ## h(r) = c (r^2 + d)^(-q).
```

```
  ##      h(r)rdr = c(r^2+d)^(1-q)/(2-2q), r=0 to  . For q > 1, this is
      ↪ 0+cd^(1-q)/(2q-2).
  ## So c = (q-1)d^(q-1)/  .
  v = runif(n)
  dist1 = sqrt(theta_d*(1-v)^(1/(1-theta_q))-theta_d)
  thet1 = runif(n)*2*pi
  x = cos(thet1)*dist1
  y = sin(thet1)*dist1
  cbind(x,y)
}

pointxy = function(n,m) matrix(0,ncol=2,nrow=n)

expxy = function(n,m){
  ## define theta_alpha externally!
  ## exponential triggering in space. f(r) = alpha/pi exp(-alpha r^2).
  ## Here the density does not depend on magnitude of the mainshock.
  ## To see that this is a density,
  ##  f (x,y)dxdy =  f (r)rdrd  = 2  f  (r)rdr
  ## = 2alpha     exp(-alpha r^2) r dr = -exp(-alpha r^2) , r=0 to  , =
      ↪ 0+1, for alpha>0.
  v = rexp(n,rate=theta_alpha)
  dist1 = sqrt(v)
  thet1 = runif(n)*2*pi
  x = cos(thet1)*dist1
  y = sin(thet1)*dist1
  cbind(x,y)
}
```

```
normxy = function(n,m){
  #theta_xy
  rmvnorm(n, sigma = diag(rep(theta_sigma_xy,2)))
}

expprod = function(m){
  ## define m0, theta_K, and theta_a externally!
  ## exponential productivity with parameters K and a for gmi.
  rpois(length(m),theta_K*exp(theta_a*(m-theta_m0)))
}

## expect Kexp(am) =      Kexp(am) bexp(-bm)dm =      Kb exp(am-bm)dm =
    ↪ Kb/(b-a).
## This is the branching ratio.
#y = mdensity(1000)
#z = gmi(y)
#mean(z)

pointprod = function(m) rpois(length(m),theta_K) ## Here each point has
    ↪  productivity theta_K.
## Define theta_K externally!

unifgt = function(n) sort(runif(n,max=unif_t_length))
## Define unif_t_length externally!

unifxy = function(n,m){
  ## define unif_xy_rad externally!
  ## generate 3n candidate points on the unit square, keep the ones in
      ↪ the unit circle, and rescale.
```

```
  numcand = max(100,3*n)
  candx = runif(numcand)*2-1
  candy = runif(numcand)*2-1
  keep = (candx^2 + candy^2 < 1)
  if(sum(keep)<n) cat("\n\n\n Error! \n\n\n")
  x2 = candx[keep>0]
  y2 = candy[keep>0]
  x = (x2[1:n])*unif_xy_rad
  y = (y2[1:n])*unif_xy_rad
  cbind(x,y)
}

power_checking_function_NS <- function(mu_in, K_in, theta_t_in, theta_
    ↪ xy_in, x_in, y_in, t_in, M){
  theta_sigma_t <<- theta_t_in
  theta_sigma_xy <<- theta_xy_in
  T <- t_in
  mu <<- mu_in; theta_K <<- K_in; theta_alpha<<-3.5; theta_beta<<-3.5;
     ↪ theta_b<<-1; theta_m0<<-3; theta_a<<-1
  X1 <- x_in
  Y1 <- y_in
  test_hawk_norm <- list('n' = 3)
  while(test_hawk_norm$n < 6){

    test_hawk_norm = simhawk(x1 = x_in, y1 = y_in, T=t_in ,mdensity =
       ↪ pointmag, gt=normgt, gxy = normxy, gmi = pointprod)
  }
  #print('we got out of while loop!')
  z <<- test_hawk_norm
```

```
scatter3D(z$lon, z$lat, z$t)
for_optim = optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_
    ↪ norm_2 )
print('for_optim')
print(for_optim$par)
print(for_optim$value)
#z$lon <<- rev(z$lon)
#z$lat <<- rev(z$lat)
#z$t <<- rev(z$t)
#z$t <<- -(z$t - z$t[1])
#back_optim = optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_
    ↪ norm_2 )
#print('back_optim')
#print(back_optim$par)
#print(back_optim$value)
#z <<- test_hawk_norm
aa <<- Mclust(data = cbind(test_hawk_norm$lon, test_hawk_norm$lat,
    ↪ test_hawk_norm$t), modelNames = 'EEI', verbose = FALSE)


num_clust <- mu_in
avg_size <- 1/(1-K_in)
sigma_mat <- diag(c(theta_xy_in, theta_xy_in, theta_t_in))
ns_true_val <- ns_likelihood_func(cbind(test_hawk_norm$lon, test_hawk
    ↪ _norm$lat, test_hawk_norm$t))
print('ns-lik')
print(ns_true_val)
#library(mvtnorm)
true_val <- (-for_optim$value - (ns_true_val))/(test_hawk_norm$n)
```

```
#print(num_clust)
#print(avg_size)
#print(sigma_mat)
sampling_distrib_1 <- c()
for(m in 1:M){
  if(m%%10 == 0) print(m)
  sim_ns <- matrix(data = NA, nrow = 3, ncol = 1)
  #sim_ns <- cbind(sim_ns, t(temp))
  for(i in 1:num_clust){
    nsamp <- rpois(1, avg_size)
    #print('nsamp')
    #print(nsamp)
    if(length(nsamp) > 0 & nsamp > 0){
      temp <- rmvnorm(nsamp, c(runif(1, 0, x_in), runif(1, 0, y_in),
        ↪ runif(1, 0, t_in)), sigma = sigma_mat)
      sim_ns <- cbind(sim_ns, t(temp))
    }

  }
  sim_ns <- sim_ns[,-1]

  z$lon <<- sim_ns[1,]
  z$lat <<- sim_ns[2,]
  z$t <<- sim_ns[3,]
  z$n <<- nrow(sim_ns)

  z$lon <<- z$lon[order(z$t)]
  z$lat <<- z$lat[order(z$t)]
  z$t <<- z$t[order(z$t)]
```

```
z$lon <<- z$lon[z$t >0]
z$lat <<- z$lat[z$t > 0]
z$t <<- z$t[z$t > 0]
z$n <<- length(z$t)
ggi <- cbind(z$lon, z$lat, z$t)

scatter3D(z$lon, z$lat, z$t, main = 'Simulated')
b3 <- optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_norm_2
    ↪ )
print('b3')
print(b3$par)
print(b3$value)

#print(b3$value)
#print(b3$par)
#z$lon <<- rev(z$lon)
#z$lat <<- rev(z$lat)
#z$t <<- rev(z$t)
#z$t <<- -(z$t - z$t[1])
#z$t
#b4 = optim(c(mu_in,K_in,theta_t_in,theta_xy_in)/2,loglhawk_norm_2
    ↪ )
#print(b4$value)
#print(b2$par)
print('ns_lik')
print(ns_likelihood_func(ggi))
sampling_distrib_1[m] <- (-(b3$value) - (ns_likelihood_func(ggi)))/
    ↪ (z$n)
```

```
  }
  return(list('true_value' = true_val, 'sampling_dist' = sampling_
    ↪ distrib_1))
}


ns_likelihood_func <- function(data_ns){
  kk <- Mclust(data_ns, modelNames = 'EEI', verbose = FALSE )
  num_clust <- ncol(kk$parameters$mean)
  avg_size <- mean(table(kk$classification))
  sigma_mat <- kk$parameters$variance$Sigma
  loglik <- 0
  for(j in 1:nrow(data_ns)){
    loglik_j <- 0
    for(k in 1:ncol(kk$parameters$mean)){
      loglik_j <- loglik_j +  avg_size*dnorm(data_ns[j,1], mean = kk$
        ↪ parameters$mean[1,k], sd = sqrt(sigma_mat[1,1]))*dnorm(data
        ↪ _ns[j,2], mean = kk$parameters$mean[2,k], sd = sqrt(sigma_
        ↪ mat[2,2]))*dnorm(data_ns[j,3], mean = kk$parameters$mean[3,
        ↪ k], sd = sqrt(sigma_mat[3,3]))
    }
    loglik <- loglik + log(loglik_j)
  }
  loglik <- loglik - avg_size*num_clust
  loglik
}
```

6.2 Code for Temporal Hypothesis Test

```
hypothesis_test_NS <- function(data_in, theta_in, K){
  #fit intput data

  out_matrix <- matrix(NA, nrow = sum(data_in), ncol = 2)
  #print(nrow(out_matrix))
  count <- 0
  print(length(data_in))
  ns_lik <- logl_ns(data_in)
  for(k in 1:length(data_in)){
    for(g in 1:data_in[k]){
      #print(k)
      if(data_in[k] > 0){

        count <- count + 1
        #print(count)
        out_matrix[count, 1] <- k
        out_matrix[count, 2] <- 1
      }
    }
  }

  clust <- Mclust(out_matrix[,1], modelNames = 'E')
  #return number of clusters, average size of clusters, and sigma
     ↪ values
  num_clust <- length(clust$parameters$mean)
  avg_size <- mean(table(clust$classification))
  sigma <- sqrt(clust$parameters$variance$sigmasq)
  sampling_dist <- c()
  for(m in 1:K){
```

```
  if(m%%100 ==0){print(m)}
  sim_ns <- c()
  mean_vals <- runif(num_clust, min = 1, max = length(data_in))
  for(i in 1:num_clust){
    sim_ns <- c(sim_ns, mean_vals[i] + rnorm(rpois(1,avg_size), 0, sd
        ↪  = sigma))
  }
  sim_ns <- round(sim_ns)
  sim_ns <- sim_ns[order(sim_ns)]
  sim_ns <- sim_ns[sim_ns > 0]
  sim_ns_hawk <- c()
  for(i in 1:max(sim_ns)){
    sim_ns_hawk[i] <- sum(sim_ns == i)
  }
  ns_lik_sim <- logl_ns(sim_ns_hawk)
  #print(sim_ns_hawk)
  forward_optim <- optim(theta_in,logl2, input_data = sim_ns_hawk)
  forward_optim <- optim(forward_optim$par, logl2, input_data = sim_
      ↪ ns_hawk)
  #print(forward_optim$value)
  #back_optim <- optim(theta_in, logl2, input_data = rev(sim_ns_hawk)
      ↪ )
  #back_optim <- optim(back_optim$par, logl2, input_data = rev(sim_ns
      ↪ _hawk))
  #print(back_optim$value)
  sampling_dist[m] <- (-forward_optim$value + ns_lik_sim )/(length(
      ↪ sim_ns))
  #print(sampling_dist)
}
```

```
  forward_optim <- optim(theta_in,logl2, input_data = data_in)
  forward_optim <- optim(forward_optim$par, logl2, input_data = data_in
    ↪ )
  return(list('val' = (-forward_optim$value + ns_lik)/(sum(data_in)),
    ↪ 'dist' = sampling_dist) )
}

logl_ns <- function(input_data){
  out_matrix <- matrix(NA, nrow = sum(input_data), ncol = 2)
  count <- 0
  for(i in 1:length(input_data)){
    if(input_data[i] > 0){
      for(j in 1:input_data[i]){
        count <- count + 1
        out_matrix[count, 1] <- i
        out_matrix[count, 2] <- 1
      }
    }
  }
  clust <- Mclust(out_matrix[,1], modelNames = 'E')
  loglik <- 0
  for(i in 1:nrow(out_matrix)){
    for(j in 1:length(clust$parameters$mean))
      loglik <- loglik + log(dnorm(out_matrix[i,1], mean = clust$
        ↪ parameters$mean[j], sd = sqrt(clust$parameters$variance$
        ↪ sigmasq) ))
  }
  loglik = mean(table(clust$classification))*loglik
  return(-loglik)
```

```
}

logl2 <- function(theta, input_data){
  if(min(theta) < 0.00000001) return(9e20)
  mu = theta[1]
  K = theta[2]
  p = theta[3]
  if(p > .999999) return(9e20)
  if(K > .999999) return(9e20)
  n <- length(input_data)
  loglik = input_data[1]*log(mu) - (mu)
  for(i in 2:n){
    lami = 0
    for(j in 1:(i-1)){
      lamij <- input_data[j]*dgeom(x = (i-j) - 1, prob = p)
      lami = lami + lamij
    }
    loglik = loglik + input_data[i]*log(mu + K*lami) - (mu + K*lami)
  }
  return(-loglik)
}
library(mclust)
Chl_Data_Week$x
ll <- hypothesis_test_NS(Chl_Data_Week$x, c(.1,.9, .5), 500)
ll$val
ll$dist
```

6.3 Simulation for Potential Outcomes Framework

```
    ## General ETAS simulator where the user can input densities.
## Have all the parameters defined externally!!!!

simhawk = function(x1=1, y1=1, T_in=100, rho=unifrho, gt=powergt, gxy=
    ↪ powerxy,
                   gmi=expprod, mdensity=expmag, sor=1, keep=1){
  ##### THIS IS FOR SIMULATING A HAWKES PROCESS WITH
  ##### lambda(t,x,y) = mu rho(x,y) +
  ##### SUM gmi(m_i) gt(t-t_i) gxy(x-xi,y-yi; mi),
  ##### on a space S = [0,x1] x [0,y1] (km), in time [0,T],
  ##### background temporal rate mu and spatial density rho(x,y),
  ##### triggering density gt(t-t_i) gxy(x-xi, y-yi; mi),
  ##### productivity gmi(m_i),
  ##### and magnitude density mdensity(m).
  ##### sor = 1 outputs the points in chronological order.
  ##### keep = 1 means only keep the ones within the space time window.
  ##### Both gt and gxy must be densities, so that if mu = 1/(x1y1),
  ##### then the integral of lambda over the space time region = mu T +
      ↪  SUM gmi(m_i).
  ##### Thus the ETAS parameter K is included in gmi.
  ##### If no magnitudes are desired, just let gmi = K.
  ##### mu should be defined externally, along with other parameters
      ↪ used in the functions.
  y = bgpts(x1,y1,T_in,rho, mdensity) ## lay down the background points
      ↪ .
  cat(y$n,"mainshocks.\n")
  calcbr = 0
```

```
  calcbr = mean(gmi(mdensity(1000000))) ## calculate branching ratio.
    ↪ Stop if br > 1.
  cat("branching ratio is ", calcbr ,"\n")
  if(calcbr >1.0){
    cat("error , branching ratio = ", calcbr , " -> 1.")
    return(0)
  }
  stop1 = 0
  if(y$n < 0.5) stop1 = 2
  cat("aftershocks by generation\n")
  w = y
  while(stop1 < 1){
    z = aft(w,x1,y1,T_in,gt,gxy,gmi,mdensity) ## place aftershocks down
      ↪ around y.
    cat(z$n," ")
    if(z$n > 0.5){
      y = combine1(y,z)
      w = z
      if(min(z$t) > T_in) stop1 = 2
    }
    if(z$n < 0.5) stop1 = 2
  }
  if(keep==1) y = keep1(y,x1,y1,T_in) ## to keep just the pts in the
    ↪ window.
  if(sor==1) y = sort1(y) ## to have the points sorted chronologically.
  y
}

## br = INT gmi(m) mdensity(m) dm, from m = m0 to infinity.
```

```
normgt = function(n){
  ## normal triggering in time with mean gmean and gsd defined
     ↪ externally!
  rnorm(n,mean=gmean,sd=gsd)
}

bgpts = function(x1,y1,T_in,rho,mdensity){
  ## define mu externally!
  z1 = list()
  n = rpois(1,mu*T_in)
  z1$n = n
  xy = rho(n,x1,y1)
  z1$lon = xy[,1]
  z1$lat = xy[,2]
  z1$t = sort(runif(n)*T_in)
  z1$m = mdensity(n)
  z1$ztimes = c()
  z1
}

aft = function(y,x1,y1,T_in,gt,gxy,gmi,mdensity){
  ## place aftershocks around y.
  z1 = list()
  z1$t = c()
  z1$n = 0
  z1$m = c()
  z1$lat = c()
  z1$lon = c()
```

```
  z1$ztimes = c()
  n2 = gmi(y$m) ## vector of number of aftershocks for each mainshock.
  for(i in 1:length(n2)){
    if(n2[i] > 0.5){
      b1 = gt(n2[i])
      z1$ztimes = c(z1$ztimes, b1)
      z1$t = c(z1$t, b1 + y$t[i])
      xy = gxy(n2[i], y$m[i])
      z1$lon = c(z1$lon, xy[,1] + y$lon[i])
      z1$lat = c(z1$lat, xy[,2] + y$lat[i])
      z1$m = c(z1$m, mdensity(n2[i]))
    }
  }
  z1$n = sum(n2)
  z1
}

combine1 = function(y,z){
  z1 = list()
  z1$t = c(y$t,z$t)
  z1$n = y$n + z$n
  z1$m = c(y$m,z$m)
  z1$lat = c(y$lat,z$lat)
  z1$lon = c(y$lon,z$lon)
  z1$ztimes = c(y$ztimes, z$ztimes)
  z1
}

keep1 = function(y,x1,y1,T_in){
```

```
  ## keep only the pts of y that are within the space time window [0,x1
     ↪ ] x [0,y1] x [0,T].
  keeps = c(1:length(y$t))[(y$t<T_in)&(y$lon<x1)&(y$lat<y1)&(y$lon>0)&(
     ↪ y$lat>0)]
  y$t = y$t[keeps]
  y$m = y$m[keeps]
  y$lon = y$lon[keeps]
  y$lat = y$lat[keeps]
  y$n = length(keeps)
  y
}

sort1 = function(y){
  ## sort the pts chronologically.
  ord2 = order(y$t)
  y$t = y$t[ord2]
  y$m = y$m[ord2]
  y$lon = y$lon[ord2]
  y$lat = y$lat[ord2]
  y
}

## rho takes an integer n and x1 and y1 and outputs a matrix of n
   ↪ locations of mainshocks.
## gt takes an integer n and outputs a vector of n nonnegative times
   ↪ since mainshock.
## gxy takes an integer n and magnitude m and outputs a matrix of n
   ↪ locs from mainshock.
## gmi takes a vector of mags m and outputs a vector of number of
```

```
    ↪ aftershocks per mainshock.
## mdensity takes an integer n and lower mag threshold m0 and outputs a
    ↪  vector of n magnitudes.

## Below are examples of functions rho, gt, gxy, gmi, and mdensity.

unifrho = function(n,x1,y1){
  ## Uniform spatial background density rho on [0,x1] x [0,y1].
  x = runif(n,min=0,max=x1)
  y = runif(n,min=0,max=y1)
  cbind(x,y)
}
unifrho_boundry = function(n, x1, y1){
  x = runif(n, min = .2, max = x1-.2)
  y = runif(n, min = .2, max = y1-.2)
  cbind(x,y)
}

## density = b e ^ -bm. cdf = 1 - e^-bm. m means m - m0.
## u = unif(0,1). F(x) = u. 1 - e^-b(m-m0) = u.
## Solve for m.
## e^-b(m-m0) = 1-u.
## -b(m-m0) = log(1-u).
## m - m0 = log(1-u)/-b.
## m = -log(1-u)/b + m0.

expmag = function(n){  ## need theta_b and theta_m0 defined externally!
  -log(1-runif(n))/theta_b + theta_m0
}
```

```
pointmag = function(n) rep(0,n)

## expmag = function(n,theta, m0=3.5){
##    ## exponential magnitude density mdensity with minimum m0 and
    ↪ mean m0+b1.
## THIS IS WRONG!!! rexp(n,rate=1/theta$b) + m0
## }

expgt = function(n){ ## need theta_beta defined externally!
  ## exponential triggering function in time gt, with mean beta.
  ## f(u) = beta e^(-beta u).
  rexp(n,rate=theta_beta)
}

powergt = function(n){
  ## power law triggering function in time gt. Define theta_c and theta
     ↪ _p externally!
  ## f(u) = (p-1) c^(p-1) (u+c)^-p.
  v = runif(n)
  theta_c*(1-v)^(1/(1-theta_p)) - theta_c
}

## Notes for powergt.
## if v = runif(1), then new time is found by letting v = F(t) and
    ↪ solving for t.
## F(t) = INT from 0 to t of f(u) du = (p-1) c^(p-1) (u+c)^(1-p) / (1-p
    ↪ )
## from u = 0 to t
```

```
## = -c^(p-1) (t+c)^(1-p) + c^(p-1) c^(1-p) = 1 - c^(p-1) (t+c)^(1-p).
## Setting v = 1 - c^(p-1) (t+c)^(1-p) and solving for t, we get
## c^(p-1) (t+c)^(1-p) = 1-v.
## (t+c)^(1-p) = (1-v) c^(1-p).
## t+c = c (1-v)^{1/(1-p)}.
## t = c (1-v)^{1/(1-p)} - c.

powerxy = function(n,m){
  ## define theta_d and theta_q externally!
  ## power law triggering in space according to ETAS (2.3), gxy, of
  ## Ogata (1998). See http://wildfire.stat.ucla.edu/pdflibrary/ogata98
     ↪ .pdf .
  ## Here the density does not depend on magnitude of the mainshock.
  ##   f (x,y)dxdy = 1 =  h (r)rdrd  = 2  h  (r)rdr.
  ## h(r) = c (r^2 + d)^(-q).
  ##      h(r)rdr = c(r^2+d)^(1-q)/(2-2q),r=0 t o  . For q > 1, this is
     ↪ 0+cd^(1-q)/(2q-2).
  ## So c = (q-1)d^(q-1)/  .
  v = runif(n)
  dist1 = sqrt(theta_d*(1-v)^(1/(1-theta_q))-theta_d)
  thet1 = runif(n)*2*pi
  x = cos(thet1)*dist1
  y = sin(thet1)*dist1
  cbind(x,y)
}

pointxy = function(n,m) matrix(0,ncol=2,nrow=n)

expxy = function(n,m){
```

```
  ## define theta_alpha externally!
  ## exponential triggering in space. f(r) = alpha/pi exp(-alpha r^2).
  ## Here the density does not depend on magnitude of the mainshock.
  ## To see that this is a density,
  ##  f(x,y)dxdy =  f(r)rdrd  = 2  f  (r)rdr
  ## = 2alpha      exp(-alpha r^2) r dr = -exp(-alpha r^2) ,r=0 to , =
     ↪ 0+1, for alpha>0.
  v = rexp(n,rate=theta_alpha)
  dist1 = sqrt(v)
  thet1 = runif(n)*2*pi
  x = cos(thet1)*dist1
  y = sin(thet1)*dist1
  cbind(x,y)
}

expprod = function(m){
  ## define m0, theta_K, and theta_a externally!
  ## exponential productivity with parameters K and a for gmi.
  rpois(length(m),theta_K*exp(theta_a*(m-theta_m0)))
}

## expect Kexp(am) =      Kexp(am) bexp(-bm)dm =      Kb exp(am-bm)dm =
   ↪ Kb/(b-a).
## This is the branching ratio.
#y = mdensity(1000)
#z = gmi(y)
#mean(z)

pointprod = function(m) rpois(length(m),theta_K) ## Here each point has
```

```
    ↪ productivity theta_K.
## Define theta_K externally!

unifgt = function(n) sort(runif(n,max=unif_t_length))
## Define unif_t_length externally!

unifxy = function(n,m){
  ## define unif_xy_rad externally!
  ## generate 3n candidate points on the unit square, keep the ones in
     ↪ the unit circle, and rescale.
  numcand = max(100,3*n)
  candx = runif(numcand)*2-1
  candy = runif(numcand)*2-1
  keep = (candx^2 + candy^2 < 1)
  if(sum(keep)<n) cat("\n\n\n Error! \n\n\n")
  x2 = candx[keep>0]
  y2 = candy[keep>0]
  x = (x2[1:n])*unif_xy_rad
  y = (y2[1:n])*unif_xy_rad
  cbind(x,y)
}

## Make sure the data are stored in z, and you define T,X1,Y1, and M0
    ↪ externally.
## First we will write the loglikelihood function in R.
loglhawk = function(theta,draw=0){
  mu = theta[1]; K = theta[2]; alpha = theta[3]; beta = theta[4]
  #cat("\n mu = ",m3(mu),", K = ",m3(K),", alpha = ",m3(alpha),", beta
     ↪ = ",m3(beta),".\n")
```

```
  if(min(mu,K,alpha,beta)<0.000000001) return(99999)
  if(K>.99999) return(99999)
  if(draw){
    r = seq(0,3,length=100)
    t = alpha/pi * exp(-alpha * r^2)
    lines(r,t,col="orange",lty=2)
  }
  sumlog = log(mu/X1/Y1)
  intlam = mu*T_in + K*z$n
  const = K*alpha/pi*beta
  for(j in 2:(z$n)){
    gij = 0
    for(i in 1:(j-1)){
      r2 = (z$lon[j]-z$lon[i])^2+(z$lat[j]-z$lat[i])^2
      gij = gij + exp(-beta*(z$t[j]-z$t[i])-alpha*r2)
    }
    lamj = mu / X1 / Y1 + const*gij
    if(lamj < 0){
      cat("lambda-",j,"-is-less-than-0.")
      return(99999)
    }
    sumlog = sumlog + log(lamj)
  }
  loglik = sumlog - intlam
  #cat("loglike is ", loglik, ". sumlog = ", sumlog,". integral = ",
    ↪ intlam,".\n")
  if(draw) lines(r,t,col="white",lty=2)
  return(-1.0*loglik)
}
```

```
loglhawk = function(theta,draw=0){
  mu = theta[1]; K = theta[2]; alpha = theta[3]; beta = theta[4]
  #cat("\n mu = ",m3(mu),", K = ",m3(K),", alpha = ",m3(alpha),", beta
    ↪ = ",m3(beta),".\n")
  if(min(mu,K,alpha,beta)<0.000000001) return(99999)
  if(K>.99999) return(99999)
  if(draw){
    r = seq(0,3,length=100)
    t = alpha/pi * exp(-alpha * r^2)
    lines(r,t,col="orange",lty=2)
  }
  sumlog = log(mu/X1/Y1)
  intlam = mu*T_in + K*z$n
  const = K*alpha/pi*beta
  for(j in 2:(z$n)){
    gij = 0
    for(i in 1:(j-1)){
      r2 = (z$lon[j]-z$lon[i])^2+(z$lat[j]-z$lat[i])^2
      gij = gij + exp(-beta*(z$t[j]-z$t[i])-alpha*r2)
    }
    lamj = mu / X1 / Y1 + const*gij
    if(lamj < 0){
      cat("lambda-",j,"-is-less-than-0.")
      return(99999)
    }
    sumlog = sumlog + log(lamj)
  }
  loglik = sumlog - intlam
```

```
  #cat("loglike is ", loglik, ". sumlog = ", sumlog,". integral = ",
      ↪ intlam,".\n")
  if(draw) lines(r,t,col="white",lty=2)
  return(-1.0*loglik)
}


demonstration_function <- function(x_v, y_v, T_v,mu_control, theta_K_
    ↪ control, theta_alpha_control, theta_beta_control, theta_b_control
    ↪ , theta_a_control, mu_treatment, theta_K_treatment, theta_alpha_
    ↪ treatment, theta_beta_treatment, theta_b_treatment, theta_a_
    ↪ treatment ){
  mu <<- mu_control; theta_K <<- theta_K_control; theta_alpha<<-theta_
      ↪ alpha_control; theta_beta<<-theta_beta_control; theta_b<<-theta
      ↪ _b_control; theta_m0<<-3; theta_a<<-theta_a_control

  T_in <<- T_v
  X1 <<- x_v
  Y1 <<- y_v
  test_sim_control = simhawk(x1 = x_v, y1 = y_v, T_in =T_v,mdensity =
      ↪ pointmag, gt=expgt, gxy = expxy, gmi = pointprod, keep = 1, rho
      ↪ =unifrho_boundry)
  mu<<-mu_treatment; theta_K <<- theta_K_treatment; theta_alpha<<-theta
      ↪ _alpha_treatment; theta_beta<<-theta_beta_treatment; theta_b<<-
      ↪ theta_b_treatment; theta_m0<<-3; theta_a<<-theta_a_treatment
  test_sim_treatment = simhawk(x1 = x_v, y1 = y_v, T_in=T_v,mdensity =
      ↪ pointmag, gt=expgt, gxy = expxy, gmi = pointprod, rho=unifrho_
      ↪ boundry, keep = 0)
  lat_max <- max(test_sim_control$lat, test_sim_treatment$lat)
```

```
lon_max <- max(test_sim_control$lon, test_sim_treatment$lon)
lat_min <- min(test_sim_control$lat, test_sim_treatment$lat)
lon_min <- min(test_sim_control$lon, test_sim_treatment$lon)
t_max <- max(test_sim_control$t, test_sim_treatment$t)
t_min <- min(test_sim_control$t, test_sim_treatment$t)


test_pp3_control <- pp3(test_sim_control$lon, test_sim_control$lat,
    ↪ test_sim_control$t,c(lon_min,lon_max,lat_min,lat_max,t_min,t_
    ↪ max))
test_pp2_control <- ppp(test_sim_control$lon, test_sim_control$lat,
    ↪ xrange = c(lon_min,lon_max), yrange = c(lat_min,lat_max))
treat_assign <- rep(c(0,1), 8)
#treat_assign <- treat_assign[-1]
test_quads <- quadrats(test_pp2_control$window, nx = 4, ny = 4)
#plot(test_pp3_control)
marks(test_quads) <- treat_assign
marked_control <- cut.ppp(test_pp2_control, test_quads)
treat_assign_function <- as.function(test_quads, values = treat_
    ↪ assign)
test_pp2_control$marks <- treat_assign_function(test_pp2_control)
test_pp3_control$marks <- test_pp2_control$marks
#fitting lambda_c on data up to time t*
z <<- list()
t_star <- T_in/2
z$t <<- test_sim_control$t[test_sim_control$t < t_star]
z$lat <<- test_sim_control$lat[test_sim_control$t < t_star]
z$lon <<- test_sim_control$lon[test_sim_control$t < t_star]
z$n <<- sum(test_sim_control$t < t_star)
```

```
theta1 = c(2,.75,.5,.5)/2
T_in <<- t_star
X1 <<- max(test_sim_control$lat) - min(test_sim_control$lat)
Y1 <<- max(test_sim_control$lon) - min(test_sim_control$lon)
#print('made it to here!!!!')
#print(theta1)
b1 = optim(theta1,loglhawk)
#print('also mat it here')
fit_lambda_c = optim(b1$par,loglhawk)



fit_lambda_c$par


#subsetting only assigned control units after time t_star
test_pp3_control_sub <- subset(test_pp3_control, test_pp3_control$
    ↪ marks == 0)
control_thinning <- list()
control_thinning$lon <- test_pp3_control_sub$data$x
control_thinning$lat <- test_pp3_control_sub$data$y
control_thinning$t <- test_pp3_control_sub$data$z
control_thinning$n <- length(test_pp3_control_sub$data$y)
test_pp3_control_sub <- subset(test_pp3_control_sub, test_pp3_control
    ↪ _sub$data$z > t_star)



#simulating treatment process
```

```
mu<<-mu_treatment; theta_K <<- theta_K_treatment; theta_alpha<<-theta
    ↪ _alpha_treatment; theta_beta<<-theta_beta_treatment; theta_b<<-
    ↪ theta_b_treatment; theta_m0<<-3; theta_a<<-theta_a_treatment
#test_sim_treatment = simhawk(x1 = x_v, y1 = y_v, T_in=T_v, mdensity =
    ↪ pointmag, gt=expgt, gxy = expxy, gmi = pointprod, keep = 0, rho
    ↪  = unifrho_boundry)
#T_treat_max <- max(test_sim_treatment$t)
test_pp3_treatment <- pp3(test_sim_treatment$lon, test_sim_treatment$
    ↪ lat, test_sim_treatment$t,c(lon_min,lon_max,lat_min,lat_max,t_
    ↪ min,t_max))
test_pp2_treatment <- ppp(test_sim_treatment$lon, test_sim_treatment$
    ↪ lat,xrange = c(lon_min,lon_max), yrange = c(lat_min, lat_max))

test_pp2_treatment$marks <- treat_assign_function(test_pp2_treatment)
test_pp3_treatment$marks <- test_pp2_treatment$marks

test_pp3_treatment_sub <- subset(test_pp3_treatment, test_pp3_
    ↪ treatment$marks == 1)
test_pp3_treatment_sub <- subset(test_pp3_treatment_sub, test_pp3_
    ↪ treatment_sub$data$z > t_star)
#thinning
#print('test1')
print(fit_lambda_c$par)
prob <- c()
for(k in 1:nrow(test_pp3_treatment_sub$data)){
  prob[k] <- min(1, 1/hawkes_intensity_function(x = test_pp3_
      ↪ treatment_sub$data$x[k],y = test_pp3_treatment_sub$data$y[k],
      ↪  z = test_pp3_treatment_sub$data$z[k], theta = fit_lambda_c$
      ↪ par,pp_thin=  control_thinning))
```

```
}
#print('test2')
prob_t <- prob
test_pp3_treatment_sub <- rthin(test_pp3_treatment_sub, prob)
#print('test3')
#fitting theta_hat_treatment
z$lon <<- test_pp3_treatment_sub$data$x
z$lat <<- test_pp3_treatment_sub$data$y
z$t <<- test_pp3_treatment_sub$data$z
z$n <<- length(test_pp3_treatment_sub$data$z)
test_sim_treatment_post <- z
z$t <<- z$t - t_star



theta1 = c(2,.75,.5,.5)/2
T_in <<- max(z$t) - t_star
X1 <<- (max(test_sim_treatment$lat) - min(test_sim_treatment$lat))/2
Y1 <<- max(test_sim_treatment$lon) - min(test_sim_treatment$lon)
#print('made it to here?')
b1 = optim(theta1,loglhawk)
fit_lambda_t = optim(b1$par,loglhawk)
print(fit_lambda_t$par)


#thinning on control
prob_c <- c()
for(k in 1:nrow(test_pp3_control_sub$data)){
  prob_c[k] <- min(1, 1/hawkes_intensity_function(x = test_pp3_
    ↪ control_sub$data$x[k],y = test_pp3_control_sub$data$y[k], z =
    ↪  test_pp3_control_sub$data$z[k], theta = fit_lambda_t$par,pp_
```

```
      ↪ thin= test_sim_treatment_post))
  }
  test_pp3_control_sub <- rthin(test_pp3_control_sub, prob_c)
  T_in <<- T_v
  T_calc <- max(nrow(test_pp3_treatment_sub$data) - nrow(test_pp3_
     ↪ control_sub$data))
  tau <- mu_treatment*(1/(1-theta_K_treatment))*(T_in/2)*(mean(treat_
     ↪ assign)) - mu_control*(1/(1-theta_K_control))*(T_in/2)*(1 -
     ↪ mean(treat_assign))
  list('treat_final' = test_pp3_treatment_sub, 'control_final' = test_
     ↪ pp3_control_sub, 'tau_hat' = nrow(test_pp3_treatment_sub$data)
     ↪ - nrow(test_pp3_control_sub$data), 'tau' = tau, 'prob_t' = prob
     ↪ _t, 'prob_c' = prob_c )
}

#intensity function to use for thinning
hawkes_intensity_function <- function(x,y,z,theta,pp_thin){
  mu = theta[1]; K = theta[2]; alpha = theta[3]; beta = theta[4]
  #print('k' = )
  #print(K)
  #print(alpha)
  const = K*alpha/pi*beta
  gij = 0
  for(i in which(pp_thin$t < z)){
    r2 = (x-pp_thin$lon[i])^2+(y-pp_thin$lat[i])^2
    gij = gij + exp(-beta*(z-pp_thin$t[i])-alpha*r2)
  }
  #lamj = mu / X1 / Y1 + const*gij
  lamj = const*gij
```

```
    lamj

}
```

Bibliography

Akbari, K., Winter, S., and Tomko, M. (2023). Spatial causality: A systematic review on spatial causal inference. *Geographical Analysis*, n/a(n/a).

Ali, M. M., Dwyer, D. S., and Rizzo, J. A. (2011). The social contagion effect of suicidal behavior in adolescents: does it really exist? *The journal of mental health policy and economics*, 14(1):3—12.

Althaus Christian L., e. (2012). Transmission of chlamydia trachomatis through sexual partnerships: a comparison between three individual-based models and empirical data. *R. Soc. Interface*.

Baddeley, A., Davies, T. M., Hazelton, M. L., Rakshit, S., and Turner, R. (2022). Fundamental problems in fitting spatial cluster process models. *Spatial Statistics*, 52:100709.

Bearman, P. S. and Moody, J. (2004). Suicide and friendships among american adolescents. *American Journal of Public Health*, 94(1):89–95. PMID: 14713704.

Browning, R., Sulem, D., Mengersen, K., Rivoirard, V., and Rousseau, J. (2021). Simple discrete-time self-exciting models can describe complex dynamic processes: A case study of covid-19. *PLOS ONE*, 16(4):1–28.

Cheng, Q., Li, H., Silenzio, V., and Caine, E. D. (2014). Suicide contagion: A systematic review of definitions and research utility. *PLOS ONE*, 9(9):1–9.

Chevalier, G. (2018). Lstm cell. [Online; accessed May 13, 2023].

Chiang, W.-H., Liu, X., and Mohler, G. (2022). Hawkes process modeling of covid-19 with mobility leading indicators and spatial covariates. *International Journal of Forecasting*, 89.

Choi, E., Du, N., Chen, R., Song, L., and Sun, J. (2015). Constructing disease network and temporal progression model via context-sensitive hawkes process. In *2015 IEEE International Conference on Data Mining*, pages 721–726.

Christakis, N. A. and Fowler, J. H. (2013). Social contagion theory: examining dynamic social networks and humanbehavior. *Statistics in Medicine*, 32(4):556–577.

Cordi, M., Challet, D., and Toke, I. M. (2017). Testing the causality of hawkes processes with time reversal. *Statistical Science*.

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

Daley, D. and Vere-Jones, D. (2008). *An Introduction to the Theory of Point Processes Volume II: General Theory and Structure*. Springer.

Daley, D. J. and Vere-Jones, D. (2016). Scoring probability forecasts for point processes: the entropy score and information gain. *Journal of Applied Probability*.

Department of Health, N. (2019).

Diggle, P. J. (2014). *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*. CRC Press.

Du, N., Dai, H., Trivedi, R., Upadhyay, U., Gomez-Rodriguez, M., and Song, L. (2016). Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1555–1564, New York, NY, USA. Association for Computing Machinery.

fdeloche (2013). Recurrent neural network unfold. [Online; accessed May 13, 2023].

Gao, B., Wang, J., Stein, A., and Chen, Z. (2022). Causal inference in spatial statistics. *Spatial Statistics*, 50:100621. Special Issue: The Impact of Spatial Statistics.

Granger, C. W. J. (1969a). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438.

Granger, C. W. J. (1969b). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438.

Guttorp, P. (1996). Stochastic modeling of rainfall. In Wheeler, M. F., editor, *Environmental Studies*, pages 171–187, New York, NY. Springer New York.

Harte, D. and Vere-Jones, D. (2005). The entropy score and its uses in earthquake forecasting. *Pure Applied Geophysics*, 162.

Hawkes, A. (1971a). Point spectra of some mutually exciting point processes. *Journal of the Royal Statistical Society*, 33(3):438–443.

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

Holland, P. W. (1986). Statistics and causal inference. *Journal of the American Statistical Association*, 81(396):945–960.

Jarvi, S., Jackson, B., Swenson, L., and Crawford, H. (2013). The impact of social contagion on non-suicidal self-injury: A review of the literature. *Archives of Suicide Research*, 17(1):1–19. PMID: 23387399.

Junhyung Park, Adam W. Chaffee, R. J. H. and Schoenberg, F. P. (2022). A non-parametric hawkes model of the spread of ebola in west africa. *Journal of Applied Statistics*, 49(3):621–637. PMID: 35706773.

Kim, S., Putrino, D., Ghosh, S., and Brown, E. N. (2011a). A granger causality measure for point process models of ensemble neural spiking activity. *PLOS Computational Biology*, 7(3):1–13.

Kim, S., Putrino, D., Ghosh, S., and Brown, E. N. (2011b). A granger causality measure for point process models of ensemble neural spiking activity. *PLOS Computational Biology*, 7(3):1–13.

Laksono, B. M., De Vries, R. D., McQuaid, S., Duprex, W. P., and De Swart, R. L. (2016). Measles virus host invasion and pathogenesis. *Viruses*, 8(8).

Li, S., Xiao, S., Zhu, S., Du, N., Xie, Y., and Song, L. (2020). Learning temporal point processes via reinforcement learning.

Mei, H. and Eisner, J. (2017). The neural hawkes process: A neurally self-modulating multivariate point process.

Moller, J. and Waagepetersen, R. (2004). *Statistical Inference and Simulation for Spatial Point Processes*. Chapman and Hall/CRC.

Neal, B. (2020). Introduction to causal inference from a machine learning perspective.

Neyman, J. and Scott, E. L. (1958). Statistical approach to problems of cosmology. *Journal of the Royal Statistical Society*, 20(1):1–43.

Ogata, Y. (1988). Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association*, 83(401):9–27.

Ogata, Y. and Zhuang, J. (2006). Space–time etas models and an improved extension. *Tectonophysics*, 413(1):13–23. Critical Point Theory and Space-Time Pattern Formation in Precursory Seismicity.

Olinde, J. and Short, M. B. (2020). A self-limiting hawkes process: Interpretation, estimation, and use in crime modeling. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 3212–3219.

Omi, T., Ueda, N., and Aihara, K. (2020). Fully neural network based model for general temporal point processes.

Palmowski, Z. and Puchalska, D. (2020). Modeling social media contagion using hawkes processes.

Papangelou, F. (1972). Integrability of expected increments of point processes and a related random change of scale. *Transactions of the American Mathematical Society*, 165:483–506.

Park, J., Schoenberg, F., Bertozzi, A., and Brantingham, P. (2021). Investigating clustering and violence interruption in gang-related violent crime data using spatial–temporal point processes with covariates. *Journal of the American Statistical Association*, 116:1–32.

Pearl, J. (2009). Causal inference in statistics: An overview. *Statistics Surveys*, 3(none):96 – 146.

Pearl, J. (2010). An introduction to causal inference. *The International Journal of Biostatistics*, 6(2).

Penttinen, A., Stoyan, D., and Henttonen, H. M. (1992). Marked Point Processes in Forest Statistics. *Forest Science*, 38(4):806–824.

Prabhakar, K., Oh, S., Wang, P., Abowd, G. D., and Rehg, J. M. (2010). Temporal causality for the analysis of visual events. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1967–1974.

Reich, B. J., Yang, S., Guan, Y., Giffin, A. B., Miller, M. J., and Rappold, A. (2021). A review of spatial causal inference methods for environmental and epidemiological applications. *International Statistical Review*, 89(3):605–634.

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

Reinhert, A. (2018). A review of self-exciting spatio-temporal point processes and their applications. *Statistical Science*, 33(3).

Rizoiu, M.-A., Mishra, S., Kong, Q., Carman, M., and Xie, L. (2018). Sir-hawkes: Linking epidemic models and hawkes processes to model diffusions in finite populations. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 419–428, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Roome, A., Spathis, R., Hill, L., Darcy, J. M., and Garruto, R. M. (2018). Lyme disease transmission risk: Seasonal variation in the built environment. *Healthcare*, 6(3).

Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461 – 464.

Scrucca, L., Fraley, C., Murphy, T. B., and Raftery, A. E. (2023). *Model-Based Clustering, Classification, and Density Estimation Using mclust in R*. Chapman and Hall/CRC.

Sha, H., Hasan, M. A., Carter, J., and Mohler, G. (2020). Interpretable hawkes process spatial crime forecasting with tv-regularization. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 3228–3236.

Shojaie, A. and Fox, E. (2022). Granger causality: A review and recent advances. *Annual review of statistics and its application*, 9.

Srivastava, S. (1998). *A Course on Borel Sets*. Springer, New YOrk.

Sun, Z., Sun, Z., Dong, W., Shi, J., and Huang, Z. (2021). Towards predictive analysis on disease progression: A variational hawkes process model. *IEEE Journal of Biomedical and Health Informatics*, 25(11):4195–4206.

Tanaka, U., Ogata, Y., and D, S. (2008). Parameter estimation and model selection for neyman-scott point processes. *Biom J*, 50.

van Lieshout, M. (2019). *Theory of Spatial Statistics: A Concise Introduction*. CRC Press.

Van Panhuis, W., Cross, A., and Burke, D. (2018a). Counts of lyme disease reported in united states of america: 1990-2016.

Van Panhuis, W., Cross, A., and Burke, D. (2018b). Counts of measles infection reported in united states of america: 1888-2002.

van Panhuis, W. G. e. a. (2013). Contagious diseases in the united states from 1888 to the present. *The New England journal of medicine*, 369(22).

W., V. P., A., C., and D., B. (2018). Counts of chlamydial infection reported in united states of america: 1995-2016.

Xiao, S., Yan, J., Chu, S. M., Yang, X., and Zha, H. (2017). Modeling the intensity function of point process via recurrent neural networks.

Xu, H., Farajtabar, M., and Zha, H. (2016). Learning granger causality for hawkes processes. In Balcan, M. F. and Weinberger, K. Q., editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1717–1726, New York, New York, USA. PMLR.

Zhuang, J. (2018). Likelihood-based detection of cluster centers for neyman–scott point processes. *Journal of Environmentl Statistics*, 8(3):1–15.