An "Estimate & Score Algorithm" for simultaneous parameter estimation and reconstruction of missing data on social networks

Rachel A. Hegemann*[1] and Erik A. Lewis[1] and Andrea L. Bertozzi[1]

[1]Department of Mathematics, University of California Los Angeles, 520 Portola Plaza, Los Angeles CA, USA

Email: Rachel A. Hegemann *- Rachel.A.Hegemann@gmail.com; Erik A. Lewis - elewis@math.ucla.edu; Andrea L. Bertozzi - Bertozzi@math.ucla.edu;

*Corresponding author

Abstract

Dynamic activity involving social networks often has distinctive temporal patterns that can be exploited in situations involving incomplete information. Gang rivalry networks in particular show a high degree of temporal clustering of activity associated with retaliatory behavior. A recent study of a Los Angeles gang network shows that known gang activity between rivals can be modeled as a self-exciting point process on an edge of the rivalry network. In real-life situations, data is incomplete and law-enforcement agencies may not know which gang is involved. However, even when gang activity is highly stochastic, localized excitations in parts of the known dataset can help identify gangs responsible for unsolved crimes. Previous work successfully incorporated the observed clustering in time of the data to identify gangs responsible unsolved crimes. However, the authors assumed that the parameters of the model are known, when in reality they have to be estimated from the data itself. We propose an iterative method that simultaneously estimates the parameters in the underlying point process and assigns weights to the unknown events with a directly calculable score function. The results of the estimation, weights, error propagation, convergence and runtime are presented.

Key words inferring missing data, social networks, gang rivalries, Hawkes process, self-exciting point processes

1 Introduction

In this work we focus our attention on data sets of violent events involving rival gangs on a social network. Each event in the data set corresponds to a violent crime that occurs at a specified time and involves a pair of rival gangs. A subset of these events are unsolved crimes in which one or both of the rival gangs is not known. The method developed in this paper could be broadly applied to any social network involving activities in time between pairs of nodes on the network. However the interest in the problem came about by examining data from the Hollenbeck Division of the Los Angeles Police Department, home to 29 street gangs with a well-known rivalry network [1, 2, 3].

Unlike other methods used to address missing data relating to social networks [4, 5], the question at hand is not if a rivalry exists, but rather to which rivalry a violent event belongs. This structure of between gang rivalries can be viewed as a social network [6] often embedded in space [1, 2]. Violent events involving gangs tend to be dyadic, and so we can formulate these events as a realization of a stochastic process occurring on the edges of the rivalry network. For each edge in the network there exists a different stochastic process. In our analysis however, we use identical parameters to generate synthetic data. The method does not assume that the underlying parameters generating each process are identical.

The first step to inferring the affiliation of the violent events is to understand the underlying stochastic process. This requires us to capture the behavior of criminal activity through computational means, much like in [7, 8, 9]. Recently methods have been proposed in the literature to mathematically model gang violence. The authors in [10] employ an agent-based model to investigate the geographic influences in the formation of the gang rivalry structure observed in Hollenbeck. These authors consider the long-term structure of the rivalry network embedded in space. In terms of the rivalry violence, a shorter timescale must be considered.

Violence among gangs exhibits retaliatory behavior [11]. In other words, given an event has happened between two gangs, the likelihood that another event will happen shortly after is increased. A problem such as this is modeled naturally by a self-exciting point process. It is interesting to note that these models were first used to analyze earthquakes [12, 13, 14, 15]. Since then, they have been used to model financial contagion in credit markets [16, 17], viral videos on the web [18], terrorist activity in Indonesia [19], and the spread of infectious disease [20].

The authors in [21] and [22] have successfully modeled the pairwise gang violence as a Hawkes process [23]. All of the events are associated with exactly one rivalry, or edge of a social network.

The violence on each edge, k, is assumed to have the conditional intensity

$$\lambda_k(t|H_{\tau,k}) = \mu_k + \alpha_k \sum_{t>t_j} \omega_k e^{-\omega(t-t_j)}. \tag{1}$$

In this Hawkes process, the intensity $\lambda_k(t|H_{\tau,k})$ depends on the history of the process $H_{\tau,k} = \{t_1, t_2, \cdots t_{M_k}\}$, where M_k is the number of events for process, k. In this framework, the window of time, $[0, T]$, observed for each process in the network is the same. However, the number of events in each process, M_k, is stochastic, and therefore varies from process to process. The background rate of the process is defined by the constant μ_k. The expected number of offspring for any event is determined by the constant α_k, and the decay of the intensity back to the background rate is ω_k. Larger values for μ_k and α_k produce more background and offspring events respectively. Larger values of ω_k do not influence the total number of events, but rather the amount of clustering in time.

The authors of [24] produce a mathematical framework to solve the incomplete data problem observed in gang violence data sets. In their work they use an optimization strategy that computes the weights to infer the rivalry affiliation of the incomplete data. In this formulation the authors prove that their optimization has a unique solution under mild constraints. This is substantial contribution in inferring the affiliation of the unknown violent events. However, the authors of [24] assume that the process parameters are known, an assumption that is often not feasible in practice. Further, finding the weights requires solving a computationally expensive optimization problem.

We propose an iterative method that (A) estimates the process parameters assuming the data is generated by the process defined by Equation 1 and (B) infers the process affiliation of simulated data via a direct method of computation. We iterate between (A) and (B) until the estimates for the unknown events converge. We call this the Estimate & Score Algorithm (ESA). The details of the ESA are described in Section 2. The ESA is tested on simulated data in Section 3, with analysis of the estimation of the parameters in the presence of missing data (see Subsection 3.1) and comparison of the proposed score functions with that of the Stomakhin-Short-Bertozzi (SSB) method in [24] (see Subsection 3.2). In Subsection 3.3 there is an analysis of the runtime between the Stomakhin-Short-Bertozzi and the Forward Backward score functions used to update the weights (see Subsection 3.3). Subsection 3.4 contains an analysis of the convergence of the Estimation & Score Algorithm. This method solves the more realistic problem of estimating the process and the weights. Further, the computation for the weight updates is more direct and therefore avoids performing the costly optimization scheme used in [24]. This is a novel piece of work with many exciting extensions. A final discussion of the results and future work is presented in Section 4. As in [24] we do not use field data in this paper, rather we generated point process data using similar

parameters as observed in the field data for Hollenbeck [22]. By using simulated data to test the algorithms we have actual ground truth evaluate the performance of the method.

1.1 Problem Formulation

The data is assumed to lie on a known social network containing K processes, where each of the K processes is a pairwise rivalry between two gangs. From this set of events, there are a total of N events where the time is known, but the processes affiliation is not known. Each of the N unknown events are placed into each of the K processes. Since the process affiliation is not known for all of the events in the network, each event is given an associated weight, $S_{i,k}$. Here $S_{i,k}$ is the ith element of the kth process. If the event is known $S_{i,k} = 1$. If $S_{i,k}$ is unknown then it is assigned a number between 0 and 1 by our algorithm. We enforce the constraint that $\sum_{k=1}^{K} S_{i,k} = 1$.

A simplified representation of our problem formulation can be found in Figure 1. The known events are represented by circles and the unknown event is represented by a triangle. Here we can see that since we do not known the affiliation of the triangle event, it is placed in all of the other processes. We emphasize that this represents our lack of information about which rivalry it belongs to.

As indicated in Figure 1, for each process in the network events $e_{i,k}$ are indexed by increasing time, $t_1 \leq t_2 \leq t_3 \cdots \leq t_{M_k}$. Ordering the events in such a way has the consequence that the first missing element in time, for example, may have different indexes for different processes. In Figure 1 the triangle index in first process is the third event, $e_{1,3}$. However the triangle in the Kth process is the second event, $e_{K,2}$. One can easily keep track of the local index of a unknown event for each process.

2 The Estimation & Score Algorithm (ESA)

The proposed Estimation & Score Algorithm can be broken into three basic stages: initialization, parameter estimation, and updating the weights. This method is succinctly described in Figure 2.

2.1 Initialization

For this paper, there were two ways of initializing the Estimate & Score Algorithm. The first is used to infer rivalry affiliation given field data. After importing the data, the unknown events are identified and placed into each of the of the K processes. The weights, $S_{i,k}$, must also be initialized. If the event is known, then $S_{i,k}$=1. If the event is unknown then $S_{i,k} = \frac{1}{K}$.

An alternate initialization is used simulate data in order to test the components of the Estimate

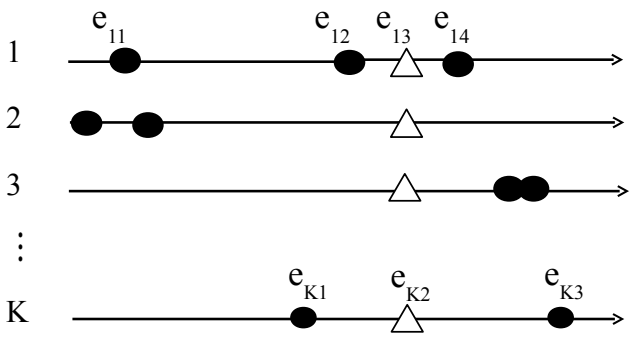


Figure 1: Simplified representation of the rivalry network with known (circle) and unknown (triangle) events.

& Score Algorithm. In this case, data is generated from K independent Hawkes processes with given μ_k, α_k, and ω_k. From these data, choose N events at random from the network to mark as unknown. Place these N unknown events into each of the other processes. Initialize the weights such that for known events $S_{i,k} = 1$ and for unknown events $S_{i,k} = 1/K$. This initialization process is used in this paper to test the method and produce the results in Section 3.

2.2 Parameter Estimation

In the presence of no unknown events, there are both parametric [12] and nonparametric [25, 26, 27, 28] ways to model the underlying stochastic process on each edge of the social network. For this work, we chose a parametric form for the triggering density to validate the model but the results could easily be extended to the nonparametric case. We note that, as is usual with nonparametric estimates, speed would be compromised for the sake of flexibility.

For this paper, the data is assumed to be a realization of Equation 1, where the parameters are estimated using a method similar to the Expectation Maximization (EM) algorithm [29]. An EM-like approach is taken because of the branching structure present in a Hawkes process. In such a process each event can be associated with a background or response event. However, given a realization from this process it is not immediately obvious whether an event is a background or response event. We can view this information as a hidden variable that we must estimate. In this way, every event in each of the K processes is assigned a probability $P^k_{i,j}$. The probability that event i is a background event is denoted $P^k_{i,i}$, and probability that event i caused event j is denoted $P^k_{i,j}$.

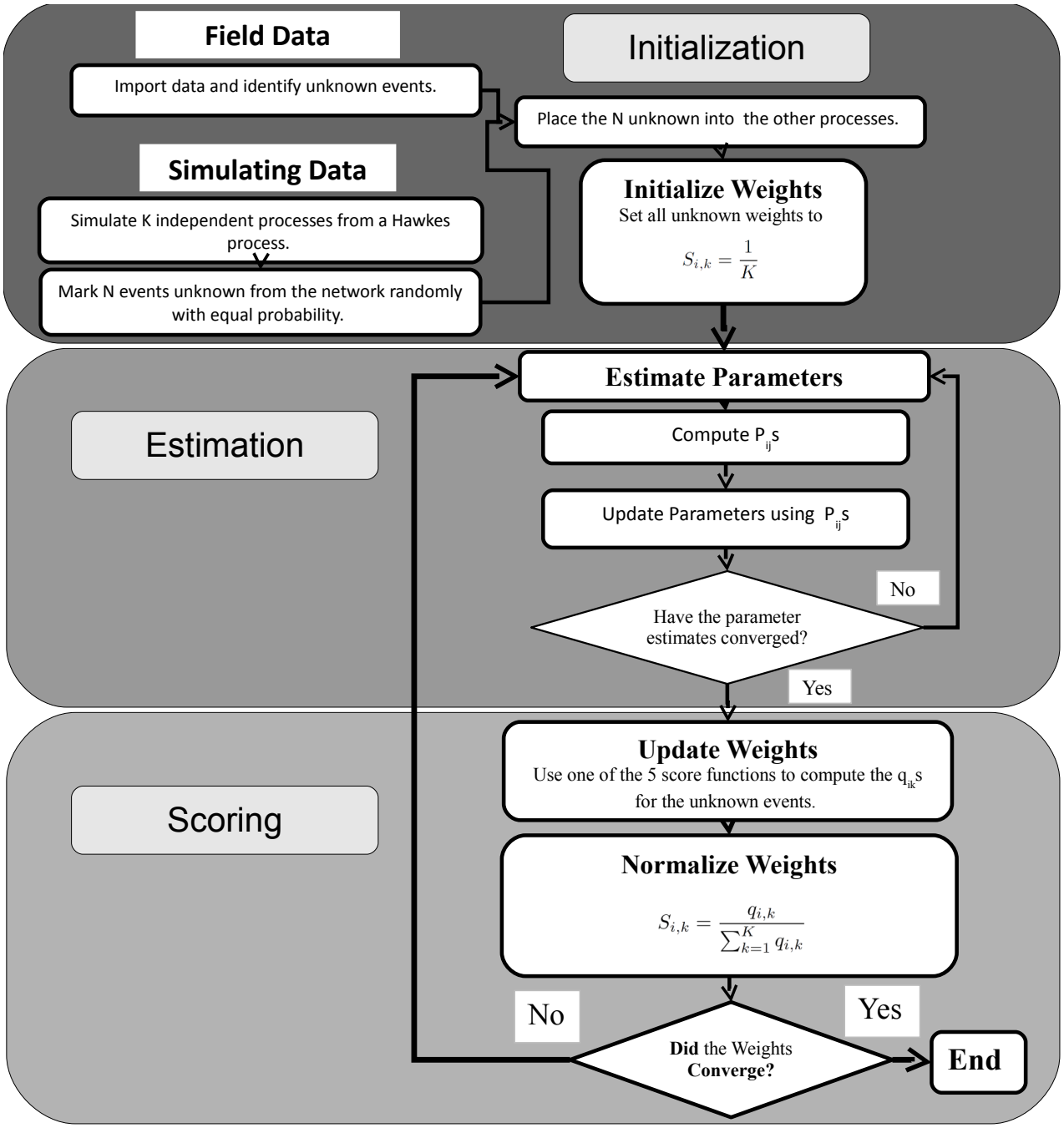


Figure 2: Flow chart of the Estimation & Score Algorithm.

This assumes that $t_i < t_j$. From this EM estimation, the approximation for each of the variables is altered to include the weights for the unknown events. In fact, in the case where all the events are known, the estimation formulas are the same. This section derives the EM estimates when in the presence of missing data.

The classical log-likelihood function $\hat{\ell}_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k)$ for a general point process is

$$\hat{\ell}_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k) = \sum_{i=1}^{M_k} \lambda_k(t_i|H_{\tau,k}) - \int_0^T \lambda_k(t|H_{\tau,k})dt. \tag{2}$$

Incorporating the branching structure into the log-likelihood function, the event association is added as a random variable, $\chi_{i,j}$ such that

$$\chi_{i,j} = \begin{cases} 1 & \text{if event } i \text{ caused event } j \text{ and } i \neq j \\ 1 & \text{if event } i \text{ is a background event and } i = j \\ 0 & \text{else} \end{cases}. \tag{3}$$

This branching allows us to separate those events associated with the background μ_k and the response $g(t) = \alpha_k \omega_k e^{-\omega_k t}$. This leads to the altered log-likelihood function

$$\begin{aligned} \ell_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k) &= \sum_{i=1}^{M_k} \chi_{i,i} \log(\mu_k) - \int_0^T \mu_k dt \\ &+ \sum_{i=1}^{M_k} \left\{ \sum_{j=i+1}^{M_k} \chi_{i,j} \log\left(\alpha_k \omega_k e^{-\omega_k(t_j - t_i)}\right) - \int_0^{T-t_i} \alpha_k \omega_k e^{-\omega_k(s)} ds \right\}. \end{aligned} \tag{4}$$

Taking the expectation of $\ell_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k)$ with respect to $\chi_{i,j}$ results in

$$\begin{aligned} E_\chi[\ell_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k)] &= \sum_{i=1}^{M_k} P_{i,i}^k \log(\mu_k) - \int_0^T \mu_k dt \\ &+ \sum_{i=1}^{M_k} \left\{ \sum_{j=i+1}^{M_k} P_{i,j}^k \log\left(\alpha_k \omega_k e^{-\omega_k(t_j - t_i)}\right) - \int_0^{T-t_i} \alpha_k \omega_k e^{-\omega_k(s)} ds \right\}. \end{aligned} \tag{5}$$

In the EM algorithm, the quantity $E_\chi[\ell_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k)]$ is maximized with respect to each of the variables $\mu_k, \alpha_k, \omega_k$ given the data $H_{\tau,k}$. This leads to the EM estimates

$$\mu_k = \frac{\sum_{i=1}^{M_k} P_{i,i}^k}{T}, \quad \alpha_k = \frac{\sum_{i<j}^{M_k} P_{i,j}^k}{M_k - \sum_{i=1}^{M_k} e^{-\omega_k(T-t_i)}} \tag{6}$$

$$\omega_k = \frac{\sum_{i<j}^{M_k} P_{i,j}^k}{\sum_{i<j}(t_j - t_i) P_{i,j}^k + \alpha_k \sum_{i=1}^{M_k} (T - t_i) e^{-\omega_k(T-t_i)}}. \tag{7}$$

Where $P_{i,j}^k$ is defined by

$$P_{i,j}^k = \frac{\alpha_k \omega_k e^{-\omega_k(t_j - t_i)}}{\lambda_k(t_i|H_{\tau,k})}, \quad P_{i,i}^k = \frac{\mu_k}{\lambda_k(t_i|H_{\tau,k})}, \tag{8}$$

for $t_i < t_j$. The EM algorithm then becomes a matter of iterating between estimating the probabilities and the parameters. It has been proven that this algorithm will converge under mild assumptions [29].

In the presence of events with unknown process affiliation in the network, we assign weights to the contribution of each event to the log-likelihood function. Specifically, each of the unknown events in process k have a weight $S_{i,k}$, such that $\sum_k S_{i,k} = 1$. For the known events $S_{i,k} = 1$. These weights are incorporated for each process via

$$\begin{aligned} L_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k) &= \sum_{i=1}^{M_k} P_{i,i}^k S_{i,k} \log(\mu_k) - \int_0^T \mu_k dt \\ &+ \sum_{i=1}^{M_k-1} \sum_{j=i+1}^{M_k} S_{i,k} S_{j,k} P_{i,j}^k \log\left(\alpha_k \omega_k e^{-\omega_k(t_j - t_i)}\right) \\ &- \sum_{i=1}^{M_k} S_{i,k} \int_0^{T-t_i} \alpha_k \omega_k e^{-\omega_k(s)} ds. \end{aligned} \tag{9}$$

Note that $L_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k)$ is no longer an EM log likelihood in the presence of unknown data. Maximizing $L_k(H_{\tau,k}|\mu_k, \alpha_k, \omega_k)$ with respect to each of the parameters the estimates become

$$\mu_k = \frac{\sum_{i=1}^{M_k} P_{i,i}^k S_{i,k}}{T}, \quad \alpha_k = \frac{\sum_{i<j}^{M_k} P_{i,j}^k S_{i,k} S_{j,k}}{\sum_{i=1}^{M_k} S_{i,k} - \sum_{i=1}^{M_k} S_{i,k} e^{-\omega_k(T-t_i)}} \tag{10}$$

$$\omega_k = \frac{\sum_{i<j}^{M_k} P_{i,j}^k S_{i,k} S_{j,k}}{\sum_{i<j}(t_j - t_i) P_{i,j}^k S_{i,k} S_{j,k} + \alpha_k \sum_{i=1}^{M_k} S_{i,k}(T - t_i) e^{-\omega_k(T-t_i)}}. \tag{11}$$

When all of the events are known, i.e. $S_{i,k} = 1$ when unknown event i, k belongs to process k and is zero otherwise, these estimates become identical to the EM parameter estimates.

2.3 Updating weights

At the start of the Estimation & Score algorithm all of the weights for the unknown events are $S_{i,k} = 1/K$. Once the parameters are estimated using the altered EM algorithm described in Equation 11, the weights, $S_{i,k}$, are updated, see Figure 2. Here we present four different score functions and the Stomakhin-Short-Bertozzi method [24], used to define, $q_{i,k}$, the intermediate process affiliation. Each of these score functions synthesize information from different portions of the data set. Given an event early in the data set, a score function that uses future events would be ideal. On the other hand, for later events a score function using previous events is desired. Similar considerations should be made if there are portion of the data with more incomplete data. After all of these intermediate weights, $q_{i,k}$, have been calculated, they are re-normalized as a probability via $S_{i,k} = \frac{q_{i,k}}{\sum_k q_{i,k}}$. For simplicity we consider a response function of the form, $g_k(t) = \alpha_k \omega_k e^{-\omega_k(t)}$.

2.3.1 Ratio Score Function

The *Ratio* score function considers the ratio of the background rate μ_k and the sum of all the future events, $\sum_{i<j} g_k(t_j - t_i)$. Mathematically the score is determined by

$$q_{i,k}^{Ratio} = \frac{\sum_{i<j} g_k(t_j - t_i)}{\mu_k(t_i)}. \tag{12}$$

2.3.2 Lambda Score Function

The *Lambda* score function uses only previous information by taking the ratio of the intensities evaluated at the unknown event time t_i.

$$q_{i,k}^{Lambda} = \frac{\lambda_k(t_i|H_{\tau,k})}{\sum_{m=1}^{K} \lambda_m(t_i|H_{\tau,k})} \tag{13}$$

2.3.3 Stomakhin-Short-Bertozzi (SSB) method

The method defined in [24] is summarized by

$$\max \left\{ \sum_k \sum_{ij} \delta_{i,j} \mu_k q_{i,k}^{SSB} + \frac{1}{2}(1 - \delta_{ij}) \alpha_k \omega_k e^{-\omega_k |t_i^k - t_j^k|} q_{i,k}^{SSB} q_{j,k}^{SSB} \right\}, \tag{14}$$

subject to

$$\sum_{k=1}^{K} \left(q_{i,k}^{SSB}\right)^2 = 1. \tag{15}$$

This method is motivated by the Hawkes process defined in Equation 1.

2.3.4 Probability Score Function

The *Probability* score function uses the approximation of the branching structure of the underlying process. The idea behind this method is events that are background events with no corresponding response events should not belong in the process. An event that is a background with many response events or an event that is a response to another event should be part of that process.

$$q_{i,k}^{Prob} = \frac{\sum_{t_j > t_i} P_{i,j}^k}{P_{i,i}^k} \tag{16}$$

$$P_{i,i}^k = \frac{\mu_k(t_i)}{\lambda_k(t_i|H_{\tau,k})} \qquad P_{i,j}^k = \frac{g_k(t_j - t_i)}{\lambda_k(t_j|H_{\tau,k})} \tag{17}$$

2.3.5 Forward Backward Response Score Function

This method is the ratio of the summation of the response for the events in the future and the past, $\sum_{i \neq j} g_k(|t_i - t_j|)$ over the background rate μ_k.

$$q_{i,k}^{FB} = \frac{\sum_{i \neq j} g_k(|t_i - t_j|)}{\mu_k} \tag{18}$$

3 Results

The Estimation & Score algorithm is tested for accuracy on simulated data from the Hawkes process defined in Equation 1. An analysis of the parameter estimation method outlined in Subsection 2.2 is conducted in Subsection 3.1. A comparison of the score functions when assuming the true parameters is found in Subsection 3.2. Subsection 3.3 provides a comparison of the runtime between the Forward Backward score function and the Stomakhin-Short-Bertozzi method. A example of convergence of the Estimate & Score Algorithm is provided in Subsection 3.4.

3.1 Estimation analysis

There are many ways we could allow the unknown events to influence our estimates of the underlying parameters for each process. There are two extremes. On the one hand, we could exclude all of the unknown events from the parameter estimation. This would be equivalent to setting the $S_{i,k} = 0$ for all unknown events. On the other hand, we could include all of the unknown events in the estimation of the parameters for each process. This would be equivalent to letting $S_{i,k} = 1$ for all i and k. Another possible estimation method is some combination of these two. We propose this as a way of allowing the unaffiliated events to play some role in the estimation process. The naive choice is allowing each event to play the same role in each process. This amounts to setting $S_{i,k} = 1/K$ for the unknown events. We compare these three choices to the estimations obtained by the Estimate & Score Algorithm (ESA) using the Forward Backward score function. Finally, we want to compare all four of these possible estimation techniques to the best we could possibly do. In this case, that would mean we knew all the affiliations for the events (i.e. there are no unknown events).

Figures 3-5 displays the results for the μ_k, α_k, and ω_k estimates for the five cases: $S_{i,k} = 0$ for unknown events (dash-triangle), $S_{i,k} = 1$ for unknown events (dash-square), $S_{i,k} = 1/K$ for unknown events (dash-x), the results using ESA (dash-circle), and the estimates you get when you know all the affiliations for the unknown events (solid). These results with standard deviations are displayed in Tables 1-3. In each of the three figures, the estimates are plotted vs the number of missing events. Each network has five processes with the true parameters $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$. Different networks are created with 15, 30, 45, 60, and 75 unknown events. Then we estimate the parameters using each of the five methods explained above. This procedure is repeated 100 times with different random seed values and then the average estimate is calculated.

Notice in the estimates for μ_k in Figure 3 and Table 1, the ESA performs the best compared to the true value and has only a slight reduction in accuracy as the number of missing events increases. On average the other three estimates seem to degrade more rapidly as the number of missing events

Table 1: Average and standard deviations for μ_k on 100 networks, true value is $\mu_k = 0.01$.

		15	30	45	60	75
Equal	(Ave)	0.0102	0.0098	0.0100	0.0099	0.0096
Weights	(StDev)	±0.0014	±0.0014	±0.0017	±0.0015	±0.0015
ESA	(Ave)	0.0099	0.0093	0.0093	0.0091	0.0086
	(StDev)	±0.0014	±0.0014	±0.0017	±0.0014	±0.0014
Unknown	(Ave)	0.0098	0.0091	0.0089	0.0085	0.0079
Not Included	(StDev)	±0.0014	±0.0014	±0.0017	±0.0015	±0.0015
Unknown	(Ave)	0.0117	0.0129	0.0143	0.0157	0.0167
Included	(StDev)	±0.0014	±0.0017	±0.0019	±0.0016	±0.0019
No Unknown	(Ave)	0.0100	0.0095	0.0094	0.0093	0.0088
	(StDev)	±0.0014	±0.0014	±0.0017	±0.0015	±0.0015

increases. When $S_{i,k} = 1$, the estimates for μ_k are far above the true value and growing as the number of unknown events increases. This follows from the fact that letting $S_{i,k} = 1$ means we are effectively adding events to the network. Take the case when $K = 2$. Assume that each process has 1000 events, and there are 100 missing from each process. When we do our estimation for the first process, we will use the 900 events we know plus the 200 unknown events from the network. We will get the identical number of events in our estimation for process two. This creates 200 new events and thus biases the estimates for μ_k. This motivated the idea of equal weighting for each unknown event, and that choice is validated by the estimates for μ_k. A similar argument shows why $S_{i,k} = 0$ (i.e. ignoring all the unknown events) has the lowest estimate for μ_k at each level of missing data.

In the estimates for the branching ratio α_k, the ESA on average yields the best estimates and maintains its accuracy in the presence of more incomplete information. It is interesting to note that equal weighting performs worse here than if we let $S_{i,k} = 0$ for all unknown events. Using the ESA overcomes this drawback. Again, setting $S_{i,k} = 1$ for all unknown events performs the worst. This could stem from the fact that most of the unknown events are being labeled background and thus this estimation technique underestimates the branching ratio because fewer events are considered offspring. Notice that the estimate for ESA tracks the best possible estimate (dash-circle) well while the other three start to trail off as more and more information is labeled as missing.

Finally, in Figure 5 and Table 1, it is shown that the ESA estimate (dash-circle) for ω_k tracks the behavior of the best estimate (solid) closer than the other methods. Including all of the unknown events (dash-square) provides the poorest estimate for ω_k. For the other three estimation techniques we see that they are all comparable.

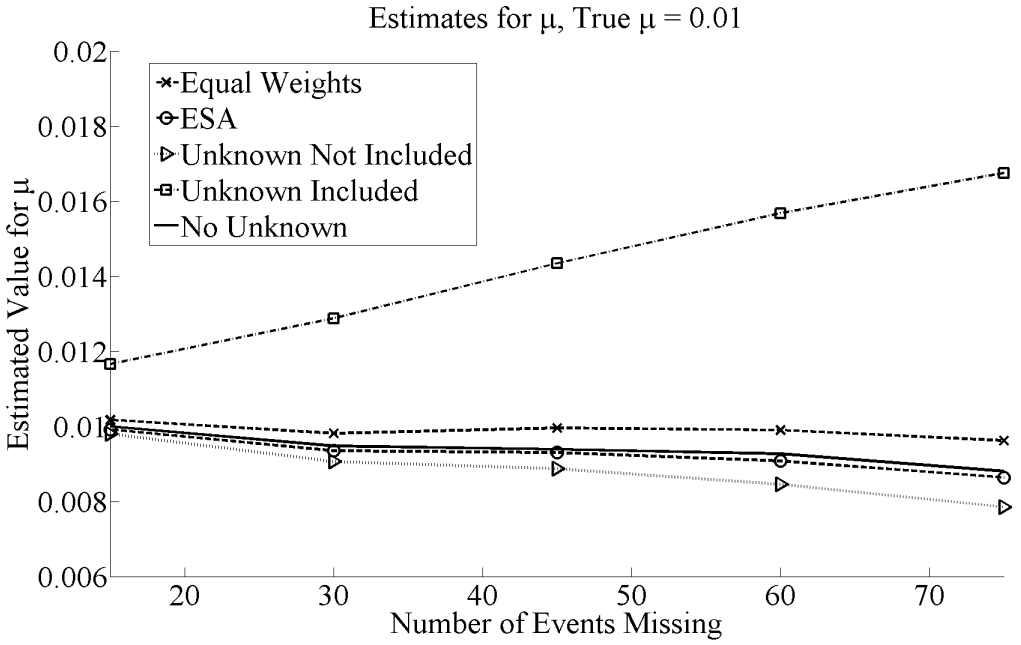


Figure 3: Plot of the parameter estimates for μ_k as the number of missing events increase.

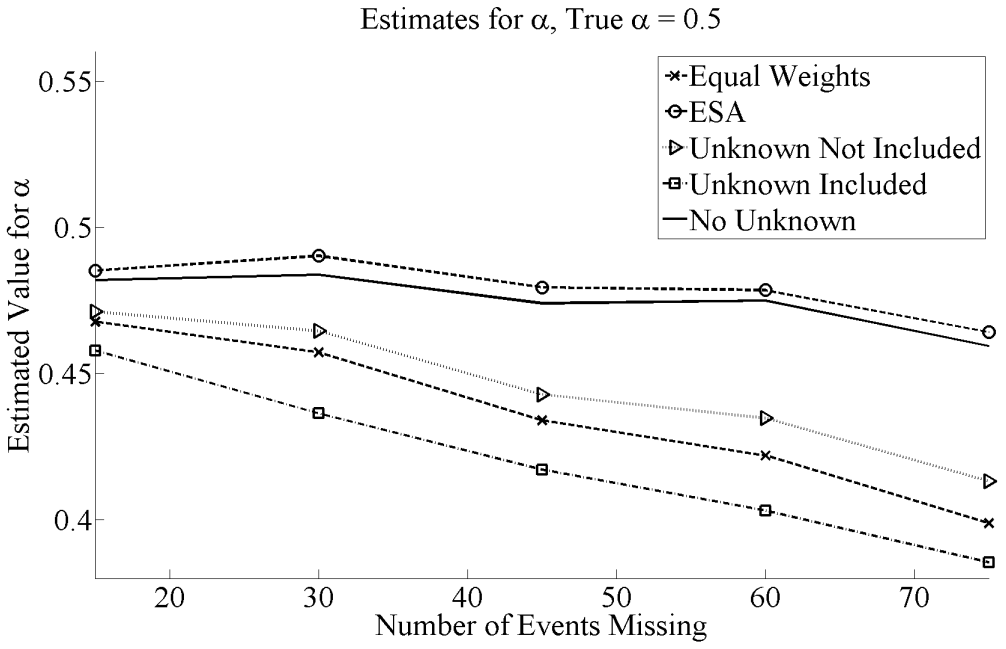


Figure 4: Plot of the parameter estimates for α_k as the number of missing events increase.

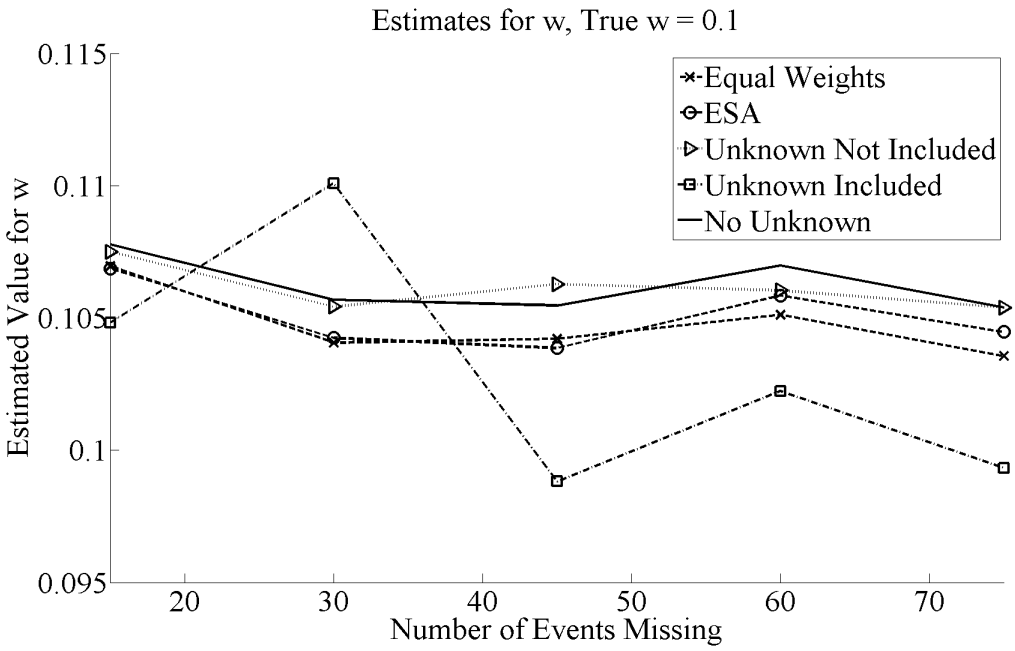


Figure 5: Plot of the parameter estimates for ω_k as the number of missing events increase.

Table 2: Average and standard deviations for α_k on 100 networks, true value is $\alpha_k = 0.5$.

		15	30	45	60	75
Equal	(Ave)	0.4678	0.4573	0.4340	0.4220	0.3989
Weights	(StDev)	±0.0636	±0.0759	±0.0686	±0.0726	±0.0699
ESA	(Ave)	0.4853	0.4903	0.4795	0.4786	0.4642
	(StDev)	±0.0640	±0.0767	±0.0712	±0.0741	±0.0719
Unknown	(Ave)	0.4712	0.4646	0.4429	0.4348	0.4132
Not Included	(StDev)	±0.0638	±0.0779	±0.0700	±0.0737	±0.0702
Unknown	(Ave)	0.4580	0.4364	0.4172	0.4032	0.3855
Included	(StDev)	±0.0668	±0.0822	±0.0705	±0.0799	±0.0818
No Unknown	(Ave)	0.4820	0.4838	0.4741	0.4750	0.4595
	(StDev)	±0.0647	±0.0759	±0.0726	±0.0748	±0.0689

Table 3: Average and standard deviations for ω_k on 100 networks, True Estimate is $\omega_k = 0.1$.

number Missing		15	30	45	60	75
Equal	(Ave)	0.1070	0.1041	0.1042	0.1051	0.10364
Weights	(StDev)	±0.0264	±0.0274	±0.0262	±0.0248	±0.0255
ESA	(Ave)	0.1069	0.1042	0.1039	0.1059	0.1045
	(StDev)	±0.0263	±0.0273	±0.0264	±0.0255	±0.0240
Unknown	(Ave)	0.1075	0.1054	0.1063	0.1060	0.1054
Not Included	(StDev)	±0.0264	±0.0286	±0.0269	±0.0246	±0.0269
Unknown	(Ave)	0.1048	0.1101	0.0988	0.1022	0.0993
Included	(StDev)	±0.0275	±0.1035	±0.0273	±0.0301	±0.0285
No Unknown	(Ave)	0.1078	0.1057	0.1055	0.1070	0.1054
	(StDev)	±0.0265	±0.0277	±0.0256	±0.0241	±0.0230

3.2 Updating Weights analysis

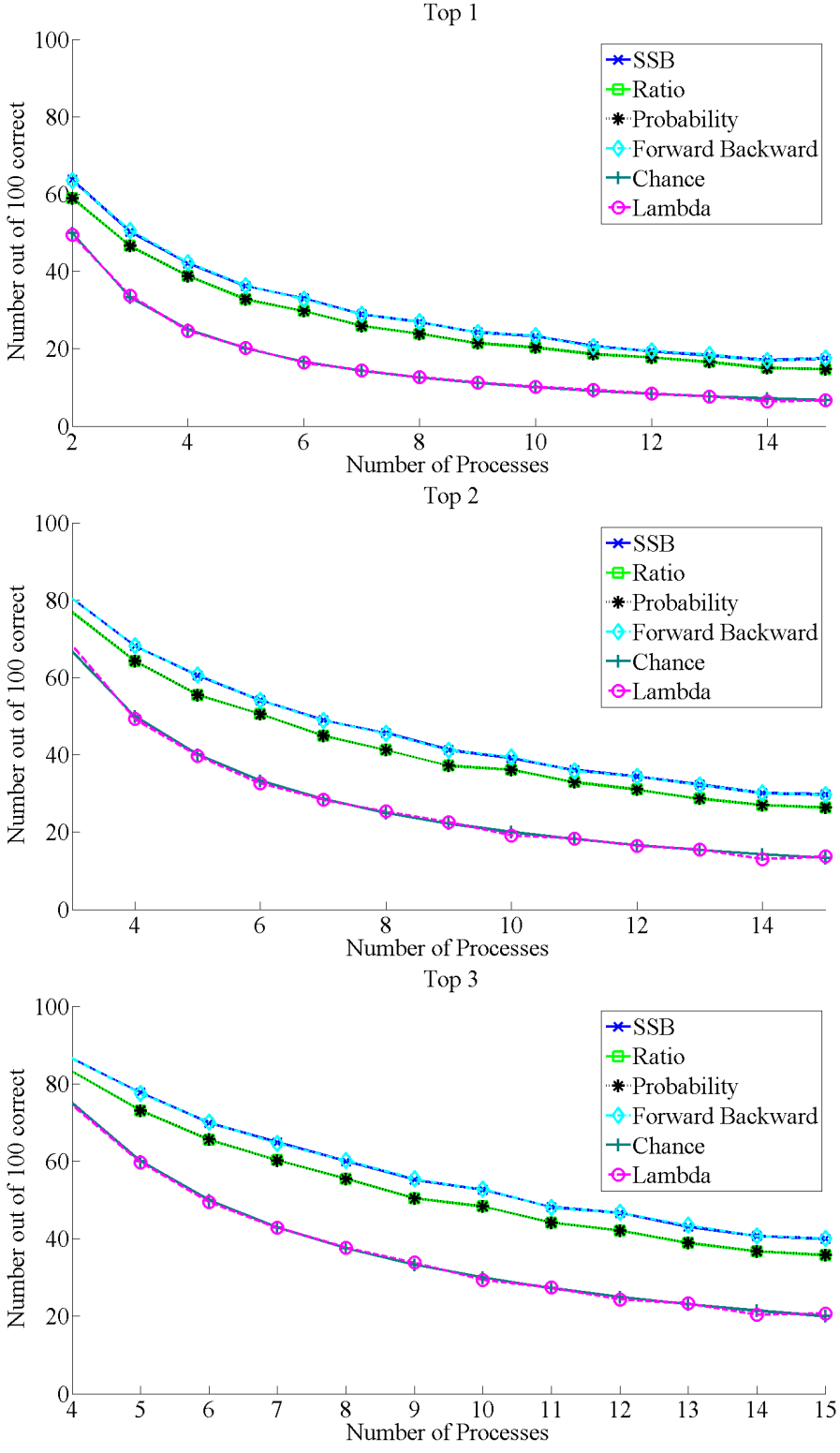
To understand the strengths and weaknesses of each of the five score functions, defined in Subsection 2.3, the score functions were evaluated for 100 missing events using the true values for μ_k, α_k, and ω_k when taking the Top 1, Top 2, and Top 3 best inferences. For comparison to [24], the true parameters were taken to be $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$. Due to the stochastic nature of the processes, for each level of process number 100 random networks were tested. The average results of this analysis are found in Figure 6. The number correctly identified by the each of the score functions is on the vertical axis. The horizontal axis displays the number of processes in the network.

From Figure 6 it is clear that the Stomakhin-Short-Bertozzi score function in solid dark blue, and the Forward Backward score function (cyan dashed diamond) perform nearly identically when looking at the Top 1, Top 2, and Top 3 inferences. These functions look both forward and backward in time from the missing event, and are therefore able to identify clusters of events in time. The Probability (black dashed asterix) and Ratio (solid green square) score functions don't do nearly as well the Stomakhin-Short-Bertozzi and Forward Backwards score functions, but better than the Lambda (magenta dashed circle) score function. The Lambda score function appears to perform close to chance (dark green solid plus) for the Top 1, Top 2, and Top 3 inferred process affiliation. Due to the success of the Forward Backward score function and the Stomakhin-Short-Bertozzi score functions only these functions are used for analysis.

The analysis comparing the score functions assumed that the true parameters were known. However, when applying this method in practice there will be error in the estimated parameters. This estimation error will propagate through to the score functions. To understand how deviations of the estimated parameters influence the score functions pairwise combinations of the parameters were increased and decreased by 90% from the target values $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$ in 10% increments. In particular the Forward Backwards and SSB score functions are computed for pairwise combinations of μ in the range of $[0.001, 0.019]$, ω in the range of $[0.01, 0.19]$, and α in the range of $[0.05, 0.95]$. Further, in these pairwise combinations, the third parameter is kept at the target value. Notice that a 90% change is larger than the errors observed in the parameter estimates in Subsection 3.1.

To examine the propagation of errors of the parameters to the score functions one event from a network with 10 processes is chosen to be missing. The score function $S_{1,true}$ with the target parameters, $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$ for the true process is calculated. Then, on the same

Figure 6: Display of the number of correctly identified missing events out of 100.



network, the parameters are offset by

$$\widehat{\text{parameter}} = \text{parameter} \pm \%\text{change} \cdot \text{parameter}, \tag{19}$$

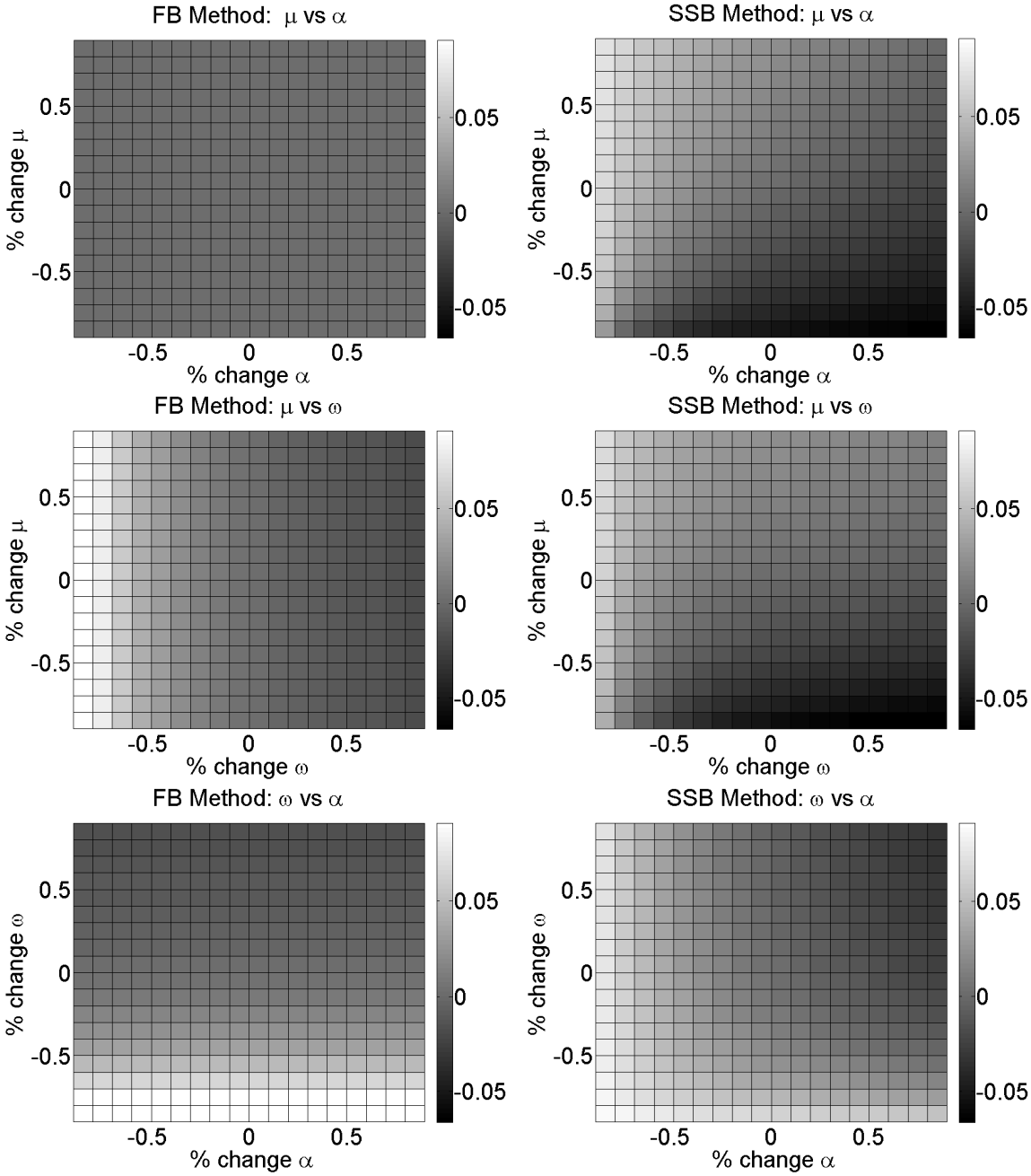
and the offset score function $\hat{S}_{1,true}$ is calculated. The difference between $S_{1,true} - \hat{S}_{1,true}$ is taken for each pairwise combination of parameters. Again, due to the stochastic nature of the processes, each analysis was done for 100 runs and the average difference in score functions is recorded. The results of this analysis are displayed in Figure 7 with those of the Forward Backwards score function (left), and those for the Stomakhin-Short-Bertozzi score function (right). In general the Stomakhin-Short-Bertozzi score function is more sensitive to the changes than the Forward Backward score functions for the μ_k and α_k parameters. Changes in the Forward Backward score functions are minimal for most changes of parameters except for small values of ω_k. As ω_k decreases then the approximated Forward Backwards score function decreases, causing a positive difference. As seen in Subsection 3.1, Figure 5, when estimating ω_k, there is a tendency to over, not under estimate the parameter, and so this does not appear to occur within these parameters. The changes in the Stomakhin-Short-Bertozzi score function depend on all of the pairwise changes of the parameters. As μ_k increases the computed Stomakhin-Short-Bertozzi decreases. On the other hand, as ω_k or α_k increase the score function increases. This analysis shows that though the Stomakhin-Short-Bertozzi method and the Forward Backward score functions preform similarly when the parameters are known exactly, under the influence of estimation error the Stomakhin-Short-Bertozzi score function varies more than the Forward Backward score function.

3.3 Runtime Analysis

Though the Forward Backward score function and the SSB method produce comparable results in terms of accuracy, there is a sizable difference in the time it takes to update the weights using these methods.

The Forward Backward score function is designed to be direct, meaning calculates the weights using available information without need for iteration. The Stomakhin-Short-Bertozzi method, however, determines the weight by solving a optimization problem. A closed form solution for the maximized weights is not known to these authors, so the weights are found by numerically approximating the weights that maximize Equation 14. In the implementation of the Stomakhin-Short-Bertozzi we employ a gradient ascent method which requires 4-11 iterations to reach convergence with a tolerance of 0.001. The direct methods, Forward Backward, Probability, Ratio, and Lambda score functions, are on the same order of operations as one iteration of the gradient ascent used to solve

Figure 7: Error propagation

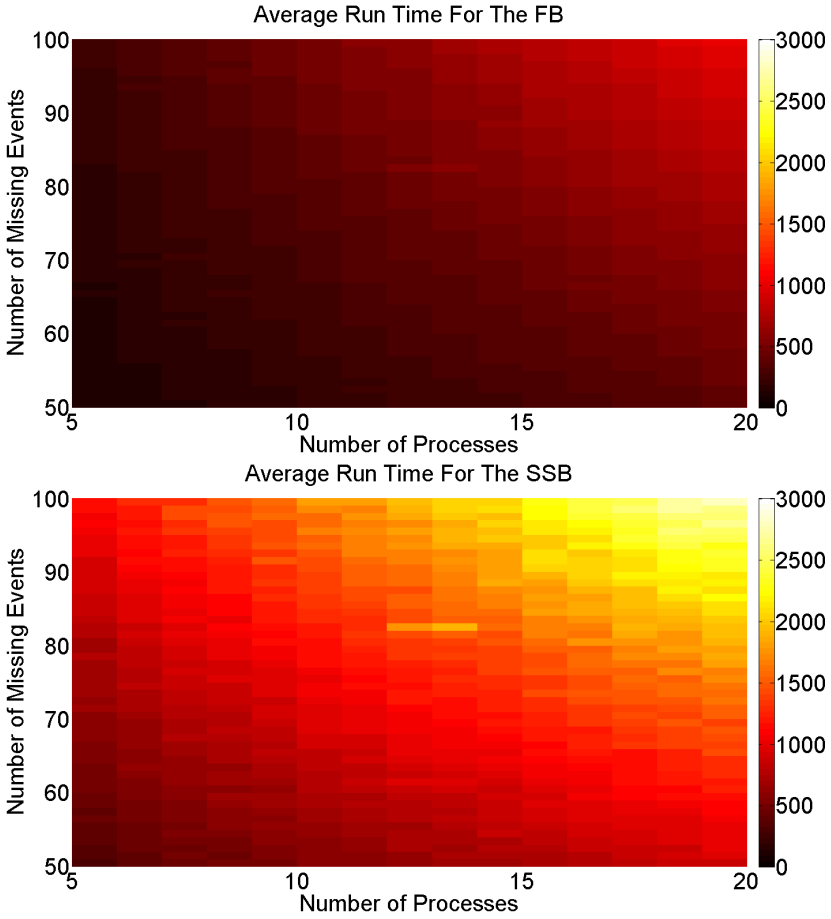


Equation 14. Specifically, one iteration of the gradient ascent method and calculating the direct score functions are $\mathrm{O}\big(N \cdot K \cdot M\big)$ where N is the number of missing events, K is the number of processes and M is the expected number of events in process k. The expected number of events in process K can be further analyzed via,

$$M = E[M_k] = \mu_k \cdot T \cdot \frac{1}{1-\alpha_k} + \frac{K-1}{K} N. \tag{20}$$

The run time of both the Forward Backward function and the Stomakhin-Short-Bertozzi method are empirically examined in Figure 8. Both score functions were calculated with 20 networks for each level of number missing and number of processes with the known parameter values of $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$. All of the run times are calculated in milliseconds. It can be seen that the average run time needed to compute the Forward Backward function at every level of N and K is substantially less than that of the Stomakhin-Short-Bertozzi method. Also, it is clear from this figure that the time needed to calculate both of these methods increases as N and K increase.

Figure 8: Display of the run time of the Forward Backward function and the Stomakhin-Short-Bertozzi method.

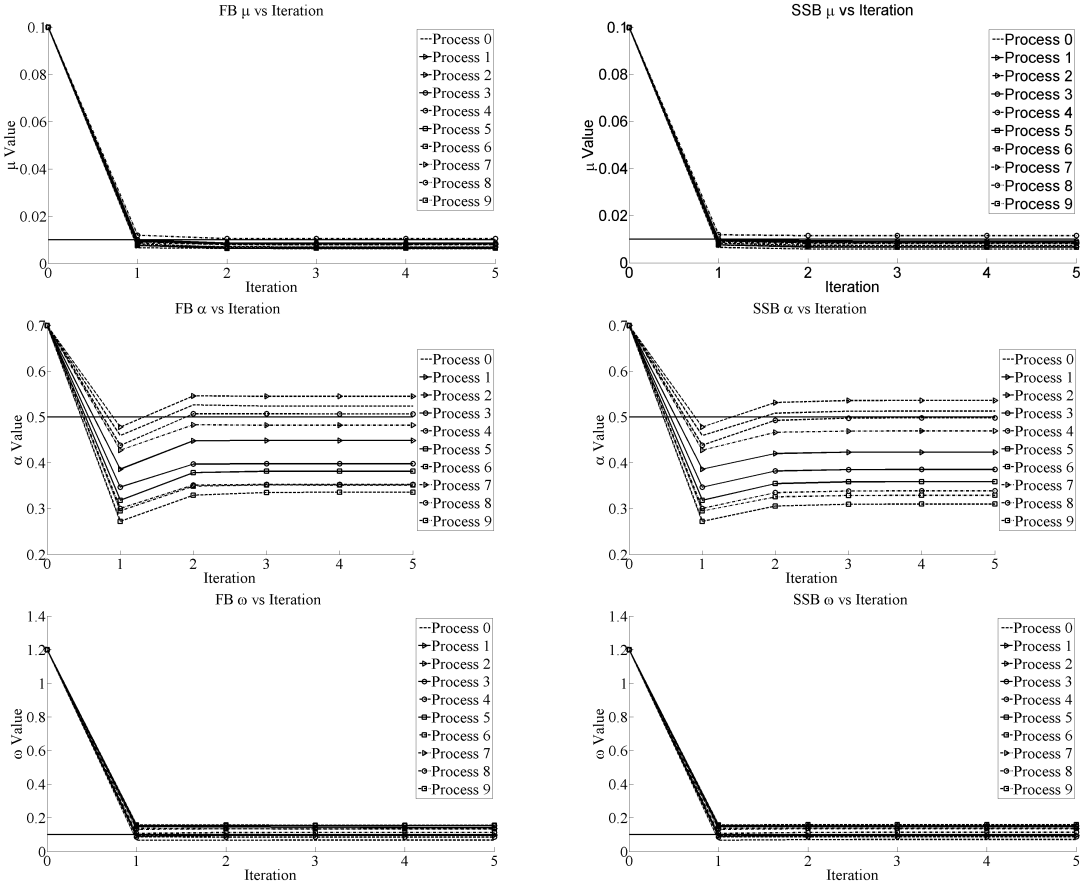


3.4 Convergence Results

The Estimation & Score Algorithm converges quickly when either the Forward Backward score function or Stomakhin-Short-Bertozzi method are used. Figure 10 displays the parameter estimates for a typical run of the Estimation & Score Algorithm for both the Forward Backward (left) and Stomakhin-Short-Bertozzi (right). Both score functions produce qualitatively similar results, and it

appears that the rate of convergence is comparable for both cases. The estimated weights for one missing data event for this typical run versus the iteration for each process are plotted in Figure 10. Here the Forward Backward weight approximations are seen on the left and the Stomakhin-Short-Bertozzi approximations are on the right. It is interesting to note that both methods of weighting choose the same process affiliation. Further tests were conducted with a variable initial weighting. These runs showed similar behavior as initializing the Estimate & Score Algorithm with $S_{i,k} = 1/K$, implying that the Estimate & Score Algorithm is robust to small perturbations of the initial weighting.

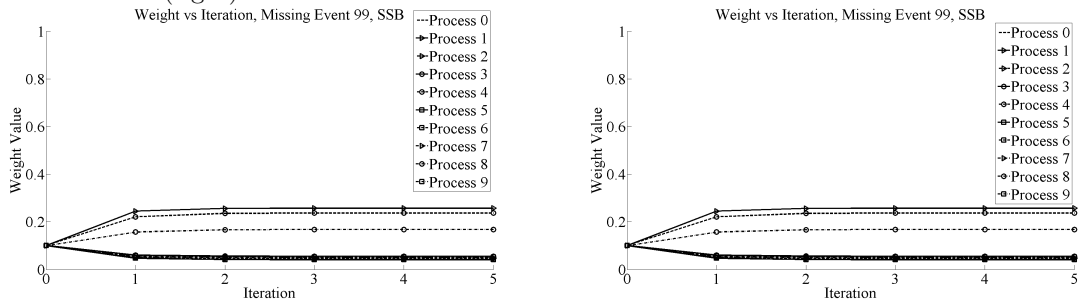
Figure 9: Parameter estimates for the Forward Backward (left) and Stomakhin-Short-Bertozzi (right) methods vs the iteration number.



4 Discussion and Future Work

In this paper we propose an effective method for simultaneously estimating the parameters and assigning process affiliation in case of incomplete field data from self-exciting point processes on a network. This problem comes from the demand for law enforcement agencies to identify gang

Figure 10: Estimated weights for one missing data Forward Backward (left) and Stomakhin-Short-Bertozzi method(right) vs iteration number.



affiliation in the case of unsolved crimes in an area of highly complex gang rivalry activity. We present a new 'Estimate & Score' algorithm for possible application to field data. By testing the method on simulated datasets we can understand its performance features and liabilities. The method is an iterative procedure in which process parameters are estimated alternately with the calculation of network affiliation probabilities. We identify several useful 'score functions' for calculating the network affiliations. We also compare the use of unknown events in the parameter estimation. One upshot of our analysis is that the inclusion of unknown events may increase the accuracy of the parameter estimation. Several score functions are considered and the Forward Backward score function shows the most promise with comparable results to that of the Stomakhin-Short-Bertozzi method of [24] in the parameter regime tested. The score function calculation is a direct method that does not rely on solving a variational problem, and thus is more computationally efficient than [24].

For future work, space often plays a role in understanding criminal activity [8, 30, 31, 32, 33]. Further, criminal behavior has non-random structure and can often be framed in terms of routine activity theory [34, 35]. In the case of gang violence, there is a strong spatial component [36, 1, 10]. One can easily extend the Estimate & Score Algorithm to include space. There is a precedence in the earthquake literature of adding space to self-exciting point processes [13, 15], however, in the case of gang violence, the spatial response may be different. Instead of retaliatory events clustering around prior events, it appears that the data is clustered around regions in space. A spatial model similar to that of [37] could be employed, where the triggering density in space is related to their respective gang set-space, or center of activity [38]. Statistically when modeling spatial point processes one needs to tease out the difference between a hot spots due to risk heterogeneity versus event dependence. The data given will be one realization of the underlying process, however using techniques such as prototyping ,[39], one could potentially reformulate the data into multiple realization of the same

process and distinguish between these two phenomena.

There are other factors in the data that can be fused into the model. For example, in earthquake modeling the magnitude of the earthquake is often included. Such a factor could be further added to the intensity $\lambda_k(t|H_{\tau,k})$ to better infer the gang affiliation. For example tagging, or other low level gang crimes, could be a precursor to more extreme violent interactions between gangs. Including this data could enrich the overall data set allowing for better analysis.

It is important to note that there are other methods to approximate the underlying form of the self exciting process. For example the authors in [28] consider the general form of the intensity function $\lambda_k(t|H_{\tau,k})$ to be

$$\lambda(t|H_\tau) = \mu(t) + \alpha \sum_{t>t_j} g(t - t_j). \tag{21}$$

Using a non-parametric method, they are able to approximated the background function $\mu(t)$ and the response function $g(t)$ for a broader class of functions. In this paper, the data was assumed to come from a Hawkes process with constant background rate and an exponential response to previous events. There are cases where the background rate is not constant [40]. Further it is conceivable that the response function could be of a form other than an exponential decay. In this circumstances, the model for $\lambda(t|H_\tau)$ in Equation 1 would not be appropriate.

Finally, this method has a great potential in the field of policing. Once such a model has been calibrated correctly, the Estimation & Score Algorithm using the quicker Forward Backward score function can be used to infer the gang association in real time, while the investigation is on going. Given an accurate model of the underlying process, such a method could identify rivalries that have heightened activity.

List of abbreviations

- **EM** Expectation Maximization
- **SSB** Stomakhin-Short-Bertozzi
- **ESA** Estimation & Score Algorithm
- **Avg** Average
- **StDev** Standard deviation

Competing interests

The authors declare that they have no competing interests

Author's contribution

RH led in the construction of the Estimation & Score algorithm, coding, analysis of the results and writing of the manuscript. EL contributed to the construction of the Estimation & Score algorithm, coding, analysis of the results, and writing of the manuscript. AB contributed to the analysis of the results, editing of the manuscript, and conception and design of the Estimation & Score algorithm. All authors read and approved the final manuscript.

Acknowledgments

This work was supported by NSF grant DMS-0968309, ONR grant N000141010221, ARO grants W911NF1010472 and W911NF1110332, and AFOSR MURI grant FA9550-10-1-0569. Further, we would like to thank Martin Short, Jeff Brantingham, and George Mohler for their helpful comments and conversations.

References

1. Radil SM, Flint C, Tita GE: **Spatializing Social Networks: Using Social Network Analysis to Investigate Geographies of Gang Rivalry, Territoriality, and Violence in Los Angeles**. *Annals of the Association of American Geographers* 2010, **100**(2):307–326, [http://www.tandfonline.com/doi/abs/10.1080/00045600903550428].

2. Tita G, Radil S: **Spatializing the social networks of gangs to explore patterns of violence**. *Journal of Quantitative Criminology* 2011, :1–25.

3. Tita G, Riley K, Ridgeway G, Grammich C, Abrahamse A, Greenwood P: **Reducing Gun Violence: Results from an Intervention in East Los Angeles**. *Natl. Inst. Justice, RAND* 2003.

4. Hoff P: **Multiplicative latent factor models for description and prediction of social networks**. *Computational & Mathematical Organization Theory* 2009, **15**:261–272, [http://dx.doi.org/10.1007/s10588-008-9040-4]. [10.1007/s10588-008-9040-4].

5. Koskinen JH, Robins GL, Pattison PE: **Analysing exponential random graph (p-star) models with missing data using Bayesian data augmentation**. *Statistical Methodology* 2010, **7**(3):366 – 384, [http://www.sciencedirect.com/science/article/pii/S1572312709000628]. [¡ce:title¿SPECIAL ISSUE ON STATISTICAL METHODS FOR THE SOCIAL SCIENCES Honoring the 10th Anniversary of the Center for Statistics and the Social Sciences at the University of Washington¡/ce:title¿].

6. Short MB, Mohler GO, Brantingham P, Tita GE: **Gang rivalry dynamics via coupled point process networks** 2011.

7. Brantingham P, Glsser U, Jackson P, Vajihollahi M: **Modeling Criminal Activity in Urban Landscapes**. In *Mathematical Methods in Counterterrorism*. Edited by Memon N, David Farley J, Hicks DL, Rosenorn T, Springer Vienna 2009:9–31, [http://dx.doi.org/10.1007/978-3-211-09442-6_2]. [10.1007/978-3-211-09442-6_2].

8. Brantingham P, Glasser U, Kinney B, Singh K, Vajihollahi M: **A computational model for simulating spatial aspects of crime in urban environments**. In *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, *Volume 4* 2005:3667 – 3674 Vol. 4.

9. Brantingham P, Brantingham P: **Computer simulation as a tool for environmental criminologists**. *Security Journal* 2004, **17**:21–30.

10. Hegemann RA, Smith LM, Barbaro AB, Bertozzi AL, Reid SE, Tita GE: **Geographical influences of an emerging network of gang rivalries**. *Physica A: Statistical Mechanics and its Applications* 2011, **390**(21-22):3894 – 3914, [http://www.sciencedirect.com/science/article/pii/S037843711100447X].

11. Decker S: **Collective and normative features of gang violence***. *Justice Quarterly* 1996, **13**(2):243–264, [http://www.ingentaconnect.com/content/routledg/rjqy/1996/00000013/00000002/art00005].

12. Veen A, Schoenberg FP: **Estimation of SpaceTime Branching Process Models in Seismology Using an EM Type Algorithm**. *Journal of the American Statistical Association* 2008, **103**(482):614–624, [http://pubs.amstat.org/doi/abs/10.1198/016214508000000148].

13. Ogata Y: **Space-Time Point-Process Models for Earthquake Occurrences**. *Annals of the Institute of Statistical Mathematics* 1998, **50**:379–402, [http://dx.doi.org/10.1023/A:1003403601725]. [10.1023/A:1003403601725].

14. Ogata Y: **Statistical models for earthquake occurrences and residual analysis for point processes**. *J. Amer. Statist. Assoc.* 1988, **83**(401):9–27.

15. Zhuang J, Ogata Y, Vere-Jones D: **Stochastic declustering of space-time earthquake occurrences**. *Journal of the American Statistical Association* 2002, **97**(458):369–380.

16. Errais E, Giesecke K, Goldberg LR: **Affine Point Processes and Portfolio Credit Risk**. *SIAM Journal on Financial Mathematics* 2010, **1**:642–665, [http://link.aip.org/link/?SJF/1/642/1].

17. Aït-Sahalia Y, Cacho-Diaz J, Laeven R: **Modeling financial contagion using mutually exciting jump processes**. Tech. rep., National Bureau of Economic Research 2010.

18. Crane R, Sornette D: **Robust dynamic classes revealed by measuring the response function of a social system**. *Proceedings of the National Academy of Sciences* 2008, **105**(41):15649–15653, [http://www.pnas.org/content/105/41/15649.abstract].

19. Porter M, White G: **Self-exciting hurdle models for terrorist activity**. *The Annals of Applied Statistics* 2011.

20. Meyer S, Elias J, Höhle M: **A Space–Time Conditional Intensity Model for Invasive Meningococcal Disease Occurrence**. *Biometrics* 2011.

21. Mohler GO, Short MB, Brantingham PJ, Schoenberg FP, Tita GE: **Self-Exciting Point Process Modeling of Crime**. *Journal of the American Statistical Association* 2011, **106**(493):100–108, [http://pubs.amstat.org/doi/abs/10.1198/jasa.2011.ap09546].

22. Egesdal M, Fathauer C, Louie K, Neuman J: **Statistical and Stochastic Modeling of Gang Rivalries in Los Angeles**. *SIAM Undergraduate Research Online* 2010.

23. Hawkes AG: **Spectra of some self-exciting and mutually exciting point processes**. *Biometrika* 1971, **58**:83–90, [http://biomet.oxfordjournals.org/content/58/1/83.abstract].

24. Stomakhin A, Short MB, Bertozzi AL: **Reconstruction of missing data in social networks based on temporal patterns of interactions**. *Inverse Problems* 2011, **27**(11):115013, [http://stacks.iop.org/0266-5611/27/i=11/a=115013].

25. Marsan D, Lengline O: **Extending earthquakes' reach through cascading**. *Science* 2008, **319**(5866):1076.

26. Marsan D, Lengliné O: **A new estimation of the decay of aftershock density with distance to the mainshock**. *Journal of Geophysical Research* 2010, **115**(B9):B09302.

27. Sornette D, Utkin S: **Limits of declustering methods for disentangling exogenous from endogenous events in time series with foreshocks, main shocks, and aftershocks**. *Physical Review E* 2009, **79**(6):61110.

28. Lewis E, Mohler G: **A Nonparametric EM Algorithm for Multiscale Hawkes Processes**. *Preprint* 2011.

29. Dempster AP, Laird NM, Rubin DB: **Maximum likelihood from incomplete data via the EM algorithm**. *J. Roy. Statist. Soc. Ser. B* 1977, **39**:1–38. [With discussion].

30. Brantingham PJ, Brantingham PL: *Environmental Criminology*. Sage Publications, Inc 1981.

31. Herbert DT: *Geography of Urban Crime*. Longman Inc 1982.

32. Eck JE, Chainey S, Leitner JGCM, Wilson RE: *Mapping Crime: Understanding Hot Spots*. National Institute of Justice 2005, [http://www.ojp.usdoj.gov/nij].

33. Brantingham P, Brantingham P: **Criminality of place**. *European Journal on Criminal Policy and Research* 1995, **3**:5–26, [http://dx.doi.org/10.1007/BF02242925]. [10.1007/BF02242925].

34. Brantingham P, Brantingham P: **Environment, routine and situation: Toward a pattern theory of crime**. *Advances in Criminological Theory* 1993, **5**:259–294.

35. Cohen LE, Felson M: **Social Change and Crime Rate Trends: A Routine Activity Approach**. *American Sociological Review* 1979, **44**(4):pp. 588–608, [http://www.jstor.org/stable/2094589].

36. Block R: **Gang Activity and Overall Levels of Crime: A New Mapping Tool for Defining Areas of Gang Activity Using Police Records**. *Journal of Quantitative Criminology* 2000, **16**:369–383, [http://dx.doi.org/10.1023/A:1007579007011]. [10.1023/A:1007579007011].

37. O'Leary M: **Modeling Criminal Distance Decay**. *Cityscape: A Journal of Policy Development and Research* 2011, **13**(3):161–198.

38. Tita G, Cohen J, Engberg J: **An Ecological Study of the Location of Gang "Set Space"**. *Soc. Probl.* 2005, **52**(2):272–299.

39. Nichols K, Schoenberg F, Keeley J, Bray A, Diez D: **The application of prototype point processes for the summary and description of California wildfires**. *Journal of Time Series Analysis* 2011, **32**(4):420–429.

40. Lewis E, Mohler G, Brantingham PJ, Bertozzi A: **Self-exciting point process of Insurgency in Iraq**. *Security Journal* 2011.

Figures

Figure 1 - Simplified representation of the rivalry network with known (circle) and unknown (triangle) events

The known events are depicted with circles and the one unknown event is depicted with a triangle. Note that since we do not know the affiliated process of this event, we place it in all processes. Associated with this event is a weights $S_{i,k} \in [0,1]$ such that $\sum_{k=1}^{K} S_{i,k} = 1$.

Figure 2 - Flow chart of the Estimation & Score Algorithm

There are two ways to implement this method. The first, (left of initialization), is the algorithm used when given an incomplete data set. The second, (right of initialization), is the algorithm used in this paper to simulate the data and test the components of the ESA. The two main phases of the algorithm are the Estimation phase (see Section 3.1) and the Update Weights phase (see Section 2.3).

Figure 3 - Plot of the parameter estimates for μ_k as the number of missing events increase

Plots of the estimates for μ_k for the Unknown Not Included (dash-triangle), Unknown Included (dash-square), Equal Weights (dash-x), ESA (dash-circle), and No Unknown (solid). In each of the three figures, the estimates are plotted vs the number of missing events. Each network has five processes with the true parameters $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$. Each data point presented is the average of the results from 100 simulated networks.

Figure 4 - Plot of the parameter estimates for α_k as the number of missing events increase

Plots of the estimates for α_k for the Unknown Not Included (dash-triangle), Unknown Included (dash-square), Equal Weights (dash-x), ESA (dash-circle), and No Unknown (solid). In each of the three figures, the estimates are plotted vs the number of missing events. Each network has five processes with the true parameters $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$. Each data point presented is the average of the results from 100 simulated networks.

Figure 5 Plot of the parameter estimates for ω_k as the number of missing events increase

Plots of the estimates for ω_k for the Unknown Not Included (dash-triangle), Unknown Included (dash-square), Equal Weights (dash-x), ESA (dash-circle), and No Unknown (solid). In each of the three figures, the estimates are plotted vs the number of missing events. Each network has five processes with the true parameters $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$. Each data point presented is the average of the results from 100 simulated networks.

Figure 6 Display of the number of correctly identified missing events out of 100

Display of the number of correctly identified missing events when the Top 1, Top 2, and Top 3 inferences are taken into consideration. For all score functions, the parameters are $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$, and assumed to be known. The Stomakhin-Short-Bertozzi score function (solid dark blue x) and the Forward Backward score (cyan dashed diamond), the Probability (black dashed asterix) and Ratio (solid green square) score functions, and the Lambda (magenta dashed circle) score function and chance (solid dark green plus) produce comperable results with these parameters.

Figure 7 Error propagation

The average difference of the Forward Backward and Stomakhin-Short-Bertozzi score functions with the parameters varied by $\pm 90\%$ of the target values, $\mu_k = 0.01$, $\omega_k = 0.1$, and $\alpha_k = 0.5$.

Figure 10 - Parameter estimates vs. the iteration number

Plots of the parameter estimates for a typical run of the Estimate & Score Algorithm using the Forward Backward (left) and Stomakhin-Short-Bertozzi (right) methods. Both methods compute nearly identical estimates of the parameters for each of the ten processes. The choice for plotting event 99 was random.

Figure 10 - Estimated weights vs. iteration

Plots of the weights for one missing event computed by a typical run of the Estimate & Score Algorithm using the Forward Backward (left) and Stomakhin-Short-Bertozzi method (right). The choice for plotting event 99 was random.