Fine-Grained ϵ-Margin Closed-Form Stabilization of Parametric Hawkes Processes

Rafael Lima[*]
Samsung R&D Institute Brazil
Av Cambacicas, 1200, Campinas-SP, Brazil
rafael.goncalves.lima@gmail.com

Abstract

Hawkes Processes have undergone increasing popularity as default tools for modeling self- and mutually exciting interactions of discrete events in continuous-time event streams. A Maximum Likelihood Estimation (MLE) unconstrained optimization procedure over parametrically assumed forms of the triggering kernels of the corresponding intensity function are a widespread cost-effective modeling strategy, particularly suitable for data with few and/or short sequences. However, the MLE optimization lacks guarantees, except for strong assumptions on the parameters of the triggering kernels, and may lead to instability of the resulting parameters . In the present work, we show how a simple stabilization procedure improves the performance of the MLE optimization without these overly restrictive assumptions. This stabilized version of the MLE is shown to outperform traditional methods over sequences of several different lengths.

1 Introduction

Temporal Point Processes are mathematical objects for modeling occurrences of discrete events in continuous-time event streams. Hawkes Processes (HP) [12], a particular form of point process, are particularly suited for capturing self- and mutual-excitation among events, i.e., when the arrival of one event makes future events more likely. HPs have recently undergone increasing popularization in both natural and social domains, such as Finance [5, 10, 2], Social Networks [27, 41], Earthquake Occurrences [23, 14], Criminal Activities [22, 19], Epidemiology [28] and Paper Citation counts [35].

In HPs, the self- and mutual-excitation is represented by the addition of a term in the expected rate of event arrivals: the *triggering kernel*. The ubiquity of applicability of HPs has given rise to a myriad of modeling approaches for this triggering effect, inspired by parametric functions (e.g., Exponential, Power-Law, Gaussian) [26, 36, 20] and Gaussian Processes [39]. For data hungry applications, choices such as non-parametric methods [37, 6, 17, 40], Neural Networks (NN) [21, 9, 34, 25, 30], Attention Models [38, 42], have been proposed. Furthermore, improvements regarding real-world data limitations and challenges are developed in [31, 32, 29].

For datasets with few and/or short sequences, a parametric choice of the triggering function is the default modeling choice [8], as a way of making use of domain knowledge to meaningfully capture the main aspects of the triggering effect. In this approach, several forms have been proposed, such as Exponential [12], Power-Law [41, 4], Rayleigh [18], Tsallis Q-Exponential [18] and Gaussian [33]. An additional, recently introduced, parametric modeling choice, the Mittag-Leffler function [8], which is not feasible for explicit gradient-based optimization, is left for future work.

[*] This work was not supported by any organization.

After the choice of the function, an unconstrained optimization is carried out for obtaining the MLE parameters. Unfortunately, both the non-convexity of the implicitly defined likelihood function, as well as eventual ill-conditioning of the initialization of the to-be-optimized self-triggering function parameters,may lead to a consequent instability of the resulting model, which is to say that the resulting HP may asymptotically result in an infinite number of events arriving in a finite time interval.

A recent work [13] proposes a Convex Conjugate approach to tackle the non-convexity of the likelihood function. Regarding the stability of parameters, [16] performs a theoretical analysis of unstable HPs, while [15, 3] point to HPs with nearly unstable parameter as suitable for some application domains. Nevertheless, [15] restricts itself to the normalization of exponential kernel by its amplitude parameter, while [3] concerns itself uniquely with asymptotic results, i.e., when the sequence of events is observed for a time window $[0, T]$, with $(T \to \infty)$. The work closest to ours is the one in [18], which can be shown to be a particular case of our algorithm, for a specific value of the Stabilization Resolution (SR) parameter (M=2).

In the following sections, we:

- Describe the MLE for Parametric HPs, along with the proposed choices for the triggering function;
- Introduce out Fine-Grained Stabilization MLE procedure (FGS-MLE);
- Validate the stabilization procedure on sets of several different lengths, also accounting for misspecification of each parametric form, with respect to one another.

2 Parametric Maximum Likelihood Estimation of Hawkes Processes

In the following, we restrict ourselves to the Univariate HP case. A realization of a Temporal Point Process over $[0, T]$ may be represented by a Counting Process N_t such that $dN_t = 1$, if there is an event at time t, or $dN_t = 0$, otherwise. Alternatively, it may be represented by a vector $\mathcal{S} = \{t_1, t_2, ..., t_N\}$, which contains all the time coordinates of the event arrivals considered in the realization of the process.

A Conditional Intensity Function (CIF) $\lambda(t)$, may be defined over N_t as:

$$\lambda(t)dt = \mathbb{E}\{dN_t = 1|\mathcal{H}_t\}, \tag{1}$$

where $\mathcal{H}_t = \{\forall t_i \in \mathcal{S} | t_i < t\}$.

The CIF of a Hawkes Process is defined as:

$$\lambda_{HP}(t) = \underbrace{\mu}_{\text{background rate}} + \underbrace{\sum_{t_i<t} \phi(t - t_i)}_{\text{excitation function}}, \tag{2}$$

in which $\mu \in \mathbb{R}^+$ is the *background rate* or *exogenous intensity*, here taken as a constant, while $\phi(t) \in \mathbb{R}^+$ is the *excitation function* or *triggering kernel* (TK), which captures the influence of past events in the current value of the CIF.

In the present work, we will deal with the following five types of parametric TKs:

Exponential (EXP(α,β)). The exponential function is hereby defined as:

$$EXP(\alpha, \beta) = \alpha e^{-\beta t} \tag{3}$$

It is a default choice for modeling quick monotonic decay of the triggering effect, found to be suitable for domains such as high-frequency finance. Its markovian property allows to compute the influence of all past events in the intensity function by considering only its value at the most recent event. **Power-Law (PWL(K,c,p))**. The power-law function is hereby defined as:

$$PWL(K, c, p) = \frac{K}{(t + c)^p} \tag{4}$$

It has been adopted for representing a slow monotonic type of decay, which has found applications in domains such as Social Network analysis [41] and Earthquake Modeling [24].

Tsallis Q-Exponential (QEXP(a,q)). The Tsallis Q-Exponential is hereby defined as:

$$QEXP(a, q) =$$

$$\begin{cases} ae^{-t} & q = 1 \\ a\left[1 + (q-1)t\right]^{\frac{1}{(1-q)}} & q \neq 0 \text{ and } 1 + (1-q)t > 0 \\ 0 & q \neq 0 \text{ and } 1 + (1-q)t \leq 0 \end{cases} \tag{5}$$

It was proposed first in domains such as Quantum Optics and Atomic Physics. Here, it is intended to be a continuous deformation among the exponential and power-law types of influences, specially in those situations in which these are not easily distinguishable [7].

Rayleigh (RAY(γ,η)). The rayleigh kernel is hereby defined as:

$$RAY(\gamma, \eta) = \gamma t e^{-\eta t^2} \tag{6}$$

It was previously used for diffusion modeling over networks [11]. Here, it is intended to represent a non-monotonically decaying and skewed type of influence, which reaches its peak some time after the event occurrence itself. Domains such as online forums' response rates were shown to not adjust well to strictly decaying effects, such as those of exponential and power-law [9].

Gaussian (GSS(γ,η)). The gaussian kernel [33] is hereby defined as:

$$GSS(\kappa, \tau, \sigma) = \kappa \mathrm{e}^{-\dfrac{(t-\tau)^2}{\sigma}} \tag{7}$$

It models a non-strictly decaying effect, such as that of the Rayleigh, but in a symmetric way.

The Maximum Likelihood Estimation procedure for parametric HPs consists in, given a choice of triggering kernels, along with its implicitly defined form of CIF, finding a tuple of optimal parameters $(\boldsymbol{\mu}^*, \theta^*)$ such that:

$$(\boldsymbol{\mu}^*, \theta^*) = \underset{\mu \in \mathbb{R}^+, \theta \in \Omega}{\operatorname{argmax}} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\mu}, \theta), \tag{8}$$

where the loglikelihood function $\mathcal{L}_{\mathcal{S}}(\boldsymbol{\mu}, \boldsymbol{\theta})$ over a sequence $\mathcal{S}$ is defined by:

$$\mathcal{L}_{\mathcal{S}}(\boldsymbol{\mu}, \boldsymbol{\theta}) = \sum_{i \in N} \log \lambda_{HP}(t_i) - \underbrace{\int_0^T \lambda_{HP}(t) dt}_{\text{Compensator}} \tag{9}$$

In practice, this optimization is done through gradient-based updates, with a learning rate δ, over μ and each of the parameters:

$$\mu^{i+1} = \mu^i + \delta \nabla_{\mu^i} \mathcal{L}_{\mathcal{S}}(\mu^i, \boldsymbol{\theta^i}) \qquad \theta^{i+1} = \theta^i + \delta \nabla_{\theta^i} \mathcal{L}_{\mathcal{S}}(\mu^i, \boldsymbol{\theta^i}) \tag{10}$$

up to a last iterate I, which is implicitly defined by a stopping criteria (e.g., Optimality Tolerance or Step Tolerance).

This procedure, first introduced in [26], lacks theoretical guarantees with respect to the convexity of $\mathcal{L}_{\mathcal{S}}(\boldsymbol{\mu}, \boldsymbol{\theta})$, over finite sequences, for a more general choice of $\phi(t)$. In practice, very restrictive assumptions are made, such as fixing $\alpha = \beta$ in the EXP(α,β) kernel, and constraining the initial values of the parameters, in advance, to be very close to the optimal values. Both of these are unfeasible for practical applications, when the true values of the parameters are hardly known. Prescinding from these often lead to the resulting final values of parameters to be very far from the optimal, and frequently corresponding to unstable configurations, i.e., when:

$$\int_{0^-}^{\infty} \phi(t) dt > 1 \tag{11}$$

This condition leads the underlying HP to be asymptotically unstable, generating an infinite number of events over a finite time interval as the length of the sequence $\mathcal{S}$ extends up to infinity.

In the next section, we will present a stabilization method, which preserves the stability of the corresponding $\phi(t)$ for each of the five introduced parametric forms of triggering kernels, without constraining their originally considered parameter space Ω. This stabilization is performed after the last iterate value of the parameters, through closed-form computations.

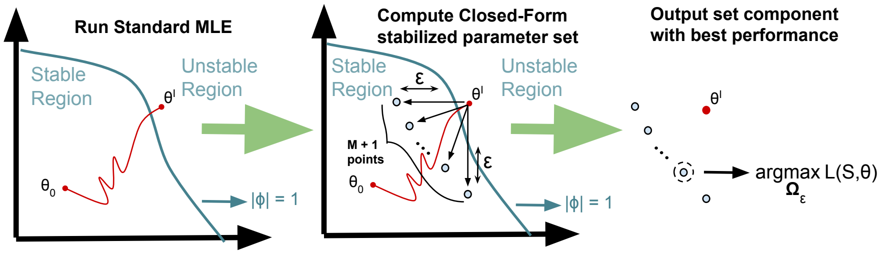

Figure 1: Diagram of the two-dimensional stabilization procedure. Given values for M and ϵ, the algorithm consists in finding (M+1) points inside the stable region for the two-dimensional parameter vector θ, each at a distance ϵ from the boundary. The (M+1) points represent a gradual change of the relevance of each component of θ to the its final stabilized value. The procedure for higher-dimensional θ follows analogously.

3 Fine-Grained ϵ-Margin Closed-Form Stabilization of Parametric Hawkes Processes

In this section, we describe the stabilization procedure for the MLE of parametric HPs. It essentially consists of finding stable configurations for both μ and $\phi(t)$ in a closed-form way, with a pre-defined values for stability margin and resolution. An intuitive graphical description of the stabilization method is shown in Figure 1

Definition 3.1. Stabilized Hawkes Process

Given a Hawkes Process implicitly defined by its background rate $\mu \in \mathbb{R}^+$ and its triggering kernel $\phi(t) : \Omega \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$, its corresponding Stabilized Hawkes Process is found by obtaining new parameter values $\mu_\epsilon \in \mathbb{R}^+_\epsilon$ and $\phi_\epsilon(t) : \Omega_\epsilon \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ such that:

$$|\phi_\epsilon| = \int_{0^-}^{\infty} \phi_\epsilon(t) = \frac{1}{1+\epsilon}, \quad \mu_\epsilon = \frac{\Lambda}{1+|\phi_\epsilon|} \quad \rightarrow \quad \boldsymbol{\theta_\epsilon} = \underset{\theta\in\Omega_\epsilon}{\operatorname{argmax}} \mathcal{L}_{\mathcal{S}}(\mu_\epsilon, \boldsymbol{\theta}) \tag{12}$$

where Ω_ϵ is a $(M \times \dim(\theta))$-sized set defined in Definition 3.2

Definition 3.2. ϵ-Margin Fine-Grained Parameter Set Ω_ϵ

Given last iterate parameter vector $\theta^I \in \Omega$ (with $D = \dim(\Omega)$), a margin parameter $\epsilon \in (0, 1)$, a Stabilization Resolution parameter $M \geq 3$, and two subindexes, j and k ($j \neq k$), from the θ^I parameter vector, the ϵ-Margin Fine-Grained Parameter Set Ω_ϵ is defined as a $(M+1) \times \dim(\Omega)$-sized tuple such that:

$$\Omega_\epsilon = \{\theta_I, \theta_0, \theta_1, ..., \theta_M\}, \tag{13}$$

with each $\theta_i = \{\theta_i^d\}_{d=1}^D$ ($i \in \{0, 1, ..., M\}$) computed such that

$$\int_{0^-}^{\infty} \phi_{\theta_i^j}(t)dt = 1/(1+\epsilon)^{\frac{i}{M}} \qquad \int_{0^-}^{\infty} \phi_{\theta_i^k}(t)dt = 1/(1+\epsilon)^{\frac{M-i}{M}}, \tag{14}$$

where $\phi_{\theta_i^j}(t)$ denotes the original $\phi(t)$ with only the j-th parameter replaced by a value which leads $\phi(t)$ to satisfy the given stabilization condition.

3.1 Closed-Form Computation of Ω_ϵ for Parametric Hawkes Processes

In this subsection, we detail the closed-form calculations of the stabilized parameter set Ω_ϵ for each of the parametric forms for $\phi(t)$ detailed at Section 2.

3.1.1 Example 1: Closed-Form Stabilization of EXP(α,β)

For $\phi(t) = EXP(\alpha, \beta)$, we have $\theta^I = (\alpha^I, \beta^I) \in \mathbb{R}^+_2$. By defining M, ϵ, j = 1 and k = 2, with $\alpha = \theta^1$, $\beta = \theta^2$ and $|\phi| = \int_{0^-}^{\infty} \phi(t)$, we can readily find:

$$\Omega_\epsilon = \{(\alpha^I, \beta^I)\} \cup \{(\alpha^I/(|\phi|(1+\epsilon))^{\frac{i}{M}}), \beta^I(|\phi|(1+\epsilon))^{\frac{M-i}{M}})\}_{i=0}^M \tag{15}$$

3.1.2 Example 2: Closed-Form Stabilization of PWL(K,c,p)

For $\phi(t) = PWL(K,c,p)$, we have $\theta^I = (K^I, c^I, p^I) \in \mathbb{R}_3^+$. By defining M, ϵ, j = 1 and k = 2, with $K = \theta^1$ and $c = \theta^2$, we can readily find:

$$\Omega_\epsilon = \{(K^I, c^I, p^I)\} \cup \{(K^I/(|\phi|(1+\epsilon))^{\frac{i}{M}}, c^I((|\phi|(1+\epsilon))^{\frac{M-i}{M}})^{\frac{1}{p^I-1}}, p^I)\}_{i=0}^{M} \tag{16}$$

By defining j = 1 and k = 3, with $K = \theta^1$ and $p = \theta^3$, we can readily find:

$$\Omega_\epsilon = \{(K^I, c^I, p^I)\} \cup (K^I/(|\phi|(1+\epsilon)), c^I, p^I) \cup \{(K^I/(|\phi|(1+\epsilon))^{\frac{i}{M}}, c^I, 1 + \frac{W(\Delta^i \log c^I)}{\log c^I})\}_{i=0}^{M-1} \tag{17}$$

where

$$\Delta^i = (|\phi|(1+\epsilon))^{\frac{M-i}{M}} (p^I - 1) c^{I(p^I-1)} \tag{18}$$

and W is a standard function, taken as the analytical continuation of branch 0 of the product log (Lambert-W) function.

3.1.3 Example 3: Closed-Form Stabilization of QEXP(a,q)

For $\phi(t) = QEXP(a,q)$, we have $\theta^I = (a^I, q^I) \in \mathbb{R}_2^+$. By defining M, ϵ, j = 1 and k = 2, with $a = \theta^1$, $q = \theta^2$ and $|\phi| = \int_{0^-}^{\infty} \phi(t)$, we can readily find:

$$\Omega_\epsilon = \{(a^I, q^I)\} \cup \{((\frac{a}{(|\phi|(1+\epsilon)})^{\frac{i}{M}}, 2 - (2-q)(|\phi|(1+\epsilon))^{\frac{M-i}{M}})\}_{i=0}^{M} \tag{19}$$

3.1.4 Example 4: Closed-Form Stabilization of RAY(γ,η)

For $\phi(t) = RAY(\gamma, \eta)$, we have $\theta^I = (\gamma^I, \eta^I) \in \mathbb{R}_2^+$. By defining M, ϵ, j = 1 and k = 2, with $\gamma = \theta^1$, $\eta = \theta^2$ and $|\phi| = \int_{0^-}^{\infty} \phi(t)$, we can readily find:

$$\Omega_\epsilon = \{(\gamma^I, \eta^I)\} \cup \{(\gamma^I/(|\phi|(1+\epsilon))^{\frac{i}{M}}), \eta^I(|\phi|(1+\epsilon))^{\frac{M-i}{M}})\}_{i=0}^{M} \tag{20}$$

3.1.5 Example 5: Closed-Form Stabilization of GSS(κ,τ,σ)

For $\phi(t) = GSS(\kappa, \tau, \sigma)$, we have $\theta^I = (\kappa^I, \tau^I) \in \mathbb{R}_2^+$. By defining M, ϵ, j = 1 and k = 2, with $\kappa = \theta^1$, $\tau = \theta^2$ and $|\phi| = \int_{0^-}^{\infty} \phi(t)$, we can readily find:

$$\begin{aligned}\Omega_\epsilon = & \{(\kappa^I, \tau^I, \sigma^I)\} \cup \{(\kappa^I/(|\phi|(1+\epsilon))^{\frac{i}{M}}, \tau^I \frac{\operatorname{erfinv}(2/\kappa\sqrt{\pi\sigma}(1+\epsilon)^{\frac{M-i}{M}} - 1)}{\operatorname{erfinv}(2|\phi|^{\frac{M-i}{M}}/(\kappa\sqrt{\pi}\sqrt{\sigma}) - 1)}, \sigma^I)\}_{i=0}^{M-1} \\ & \cup \{(\kappa^I/(|\phi|(1+\epsilon)), \tau^I, \sigma^I)\},\end{aligned} \tag{21}$$

where *erfinv* is the inverse of the error function ($erf(z) = \frac{2}{\sqrt{\pi}} \int_0^z \mathrm{e}^{-x^2} dx$). In practice, the arguments of *erfinv* functions are constrained to be less than 1.

4 Experiments

For empirical validation of the proposed stabilization strategy, we performed experiments with synthetic data generated from each of the five parametric forms, with μ set as 0.5, and parameter values:

$$EXP(1.0, 1.1) \quad PWL(0.9, 1.0, 2.0) \quad QEXP(0.8, 1.1) \quad RAY(1.2, 1.0) \quad GSS(0.5, 0.5, 1.0), \tag{22}$$

Since last iterate $\theta^I \in \Omega_\epsilon$, then clearly $\max_{\theta \in \Omega_\epsilon} \mathcal{L}(\theta, \mathcal{S}) \geq \mathcal{L}(\theta^I, \mathcal{S})$. Our goal is, then, verifying the rate with which the stabilization algorithm output strictly outperforms θ^I, including those cases of model misassignment (e.g., an exponential kernel is used to fit a sequence generated by a gaussian kernel, and so on). In the following, we investigate the performance of the stabilization algorithm over sequences of several different horizons T, and also verify the influence of margin parameter ϵ and stabilization resolution M on the success rate of stabilized kernels. The MLE optimization procedure is carried out with the standard Nelder-Mead method implementation from the Python *SciPy* library. As an example, a success rate of 0.9 for EXP kernel fitted over sequences generated with GSS kernels mean that the likelihood of the parameters returned by the stabilization method is strictly higher than the original MLE last iterate θ^I in 90 % of the corresponding sequences.

4.1 Influence of horizon T

We considered the horizon values $T \in \{10, 50, 100, 500\}$. For each value of T, we verified the performance of the stabilized search from each kernel type in 10 sequences of each kernel type, thus also accounting for misspecification of the parametric form. We fixed $\epsilon = 0.1$ and $M = 4$. The results are shown in Figure 2.

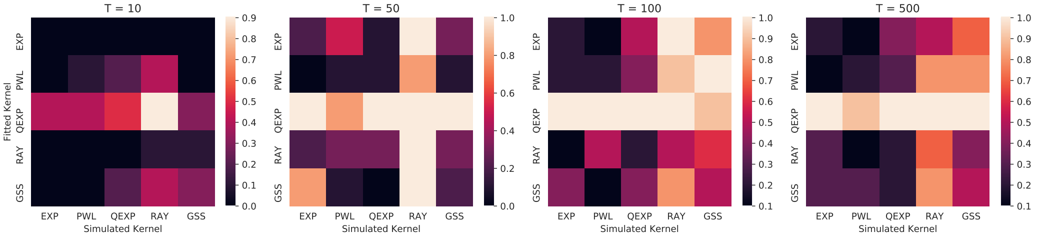

Figure 2: Influence of time horizon T in the performance of Stabilization Procedure.

Overall, success rates of the stabilized parameters outperforming the original last iterates tend to be higher at the midsize and larger sequences ($T \in \{50, 100, 500\}$), than at the smallest ones ($T = 10$).

4.2 Influence of Stabilization Resolution parameter M

For verifying the effect of the Stabilization Resolution parameter M, we fixed $\epsilon = 0.1$ and $T = 100$ while considering $M \in \{4, 8, 10, 50\}$.

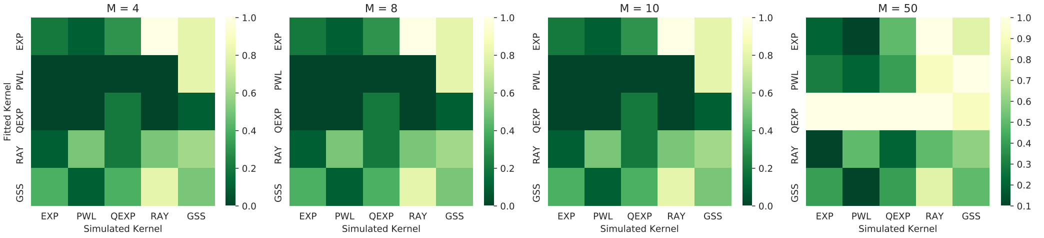

Figure 3: Influence of Stabilization Resolution M in the performance of Stabilization Procedure.

With the increasing of M above the lowest chosen value of 4, the success rates of the stabilized parameters show some slight increase some choices of fitting kernels, as shown in Figure 3. Since the stabilized set is computed in closed-form, larger values of M do not significantly affect the complexity of the algorithm, as long as it is kept low with respect to the average number of iterations of the initial MLE optimization procedure. Thus, setting larger values for M still results in a slight improvement over the performance of the algorithm.

4.3 Influence of Margin parameter ϵ

For verifying the effect of the margin criterion ϵ, we fixed $T = 100$ and $M = 6$, and analyzed the performance of the stabilization for $\epsilon \in \{0.1, 0.01, 0.001, 0.0001\}$

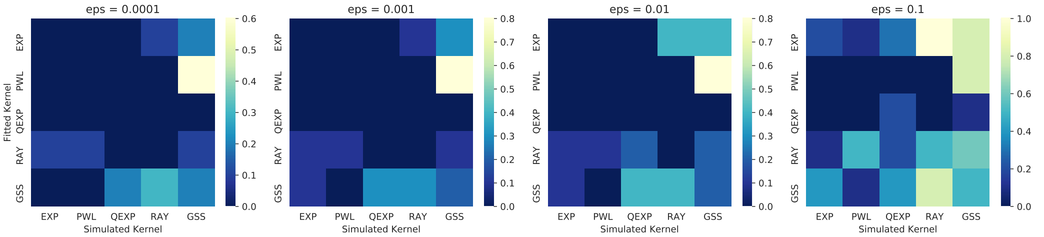

Figure 4: Influence of Margin Parameter ϵ in the performance of Stabilization Procedure.

Increasing ϵ intuitively means picking projected points which fall farther away from the instability boundary. Figure 4 shows that a larger value of ϵ increases the overall success rate of the stabilization procedure.

4.4 Performance across different optimization methods

For verifying the performance of the stabilization algorithms beyond the default choice of Nelder-Mead method, we computed the average loglikelihood improvement for each kernel type across the following optimization methods: **Nelder-Mead**, **Conjugate Gradient (CG)**, **BFGS**, **L-BFGS-B**, **TNC**, **COBYLA** and **SLSQP**.

All the methods were run with the default *SciPy* implementations, and the routines resulting in overflow were ignored. The parameters used were: T = 100, ϵ = 0.1 and M = 6. The results are shown in Table 1. The stabilization method shows increase in performance across all methods.

Method	Nelder-Mead	CG	BFGS	L-BFGS-B	TNC	COBYLA	SLSQP
$\Delta\mathcal{L}$	+99.9	+82.31	+64.38	+82.56	+91.19	+97.35	+86.09

Table 1: Comparison of the stabilization method among different optimization methods. $\Delta\mathcal{L}$ is the average difference in loglikelihood among the stabilized kernel and the original last iterate kernel across all sequences.

The kernel types with largest increase in performance were EXP and QEXP.

4.5 Baseline Comparisons on Real and Synthetic Data

For testing the quality of our stabilization algorithms, hereby denoted **MLE- STAB**, we performed experiments with real data. The chosen datasets were "Hawkes","Retweet" and "Missing".Their description and content can be found in [21]. For those datasets with more than one fold, only the first fold was used. Besides, only sequences with more than 20 events were considered. The "Meme" dataset had all sequences with a number of events which disallowed the fitting of all baselines, while "Mimic" dataset resulted in only one sequence being properly fitted for comparison. Thus, both these datasets were excluded from our comparison. The baselines chosen were those which allowed for a more flexible modeling of the triggering function. They are the implementations of **HawkesEM**, **HawkesBasisKernels** and **HawkesConditionalLaw**, publicly available in the *tick* library [1].

We intend to show here that, for most real-world applications, the number of events associated with a given self-triggering dynamics is usually small. This fact is advantageous to simple parametric functions, such as the ones we presented here. And the improvements proposed by the MLE-STAB algorithm works on improving the predictive power of these simpler parametric models. The models are fitted in the events up to 70% of T, here taken as the time of the last event. And then have their loglikelihood evaluated in the final 30%. The loglikelihood of each sequence is then normalized by the number of events in the test portion. Unfortunately, the very small number of events was prohibitive for the **HawkesBasisKernels** and **HawkesConditionalLaw** models, thus only results for **HawkesEM** and **MLE-STAB** are displayed. The final average across all sequences of the dataset is shown in Table.

Dataset /Method	MLE-STAB	HawkesEM
Hawkes	**1.0**	0.039
Missing	**1.0**	-0.06
Retweet	**0.99**	-20.68

Table 2: Comparison of the stabilization method with baseline over standardized datasets. The result shown corresponds to the average, across all sequences, of the loglikelihood of the method on the test portion of the given sequence (final 30% of the event horizon $[0, T]$). The loglikelihood is normalized by the number of events contained in this given test portion.

5 Conclusions

Hawkes Processes are particularly suited for modeling self- and mutually exciting interactions of discrete events in continuous-time event streams, what corroborates the increasing popularity they have enjoyed across many domains, such as Social Network analysis and High-Frequency Finance. A Maximum Likelihood Estimation (MLE) unconstrained optimization procedure over parametrically assumed forms of the triggering kernels of the corresponding intensity function are a widespread cost-effective modeling strategy, particularly suitable for data with few and/or short sequences. However, the MLE optimization lacks guarantees, except for strong assumptions on the parameters of the triggering kernels, and may lead to instability of the resulting parameters . In the present work, we show how a simple stabilization procedure improves the performance of the MLE optimization without these overly restrictive assumptions, while also accounting for model misassignment cases. This stabilized version of the MLE is shown to outperform the standard method over sequences of several different lengths. The effects of resolution and margin parameters are also evaluated, as well as baseline comparisons over standardized datasets.

References

[1] E. Bacry, M. Bompaire, S. Gaïffas, and S. Poulsen. tick: a Python library for statistical learning, with a particular emphasis on time-dependent modeling. *ArXiv e-prints*, July 2017.

[2] E. Bacry, K. Dayri, and J. F. Muzy. Non-parametric kernel estimation for symmetric hawkes processes. application to high frequency financial data. *The European Physical Journal B*, 85(5):1–12, 2012.

[3] Emmanuel Bacry, Sylvain Delattre, Marc Hoffmann, and J.F. Muzy. Scaling limits for hawkes processes and application to financial statistics. 123, 02 2012.

[4] Emmanuel Bacry, Thibault Jaisson, and Jean–François Muzy. Estimation of slowly decreasing hawkes kernels: application to high-frequency order book dynamics. *Quantitative Finance*, 16(8):1179–1201, 2016.

[5] Emmanuel Bacry, Iacopo Mastromatteo, and Jean-François Muzy. Hawkes processes in finance. *arXiv*, 2015.

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

[7] Thierry Bochud and Damien Challet. Optimal approximations of power-laws with exponentials. 5 2006.

[8] Jing Chen, Alan G. Hawkes, and Enrico Scalas. A fractional hawkes process, 2020.

[9] Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. 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, San Francisco, CA, USA, August 13-17, 2016*, 2016.

[10] J. Etesami, N. Kiyavash, K. Zhang, and K. Singhal. Learning network of multivariate hawkes processes: A time series approach. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2016.

[11] Manuel Gomez-Rodriguez. *Structure and Dynamics of Diffusion Networks*. PhD thesis, Stanford University, 2013.

[12] Alan G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, (1):201–213, 1971.

[13] Niao He, Zaid Harchaoui, Yichen Wang, and Le Song. Point process estimation with mirror prox algorithms. *Applied Mathematics & Optimization*, 2019.

[14] A. Helmstetter and D. Sornette. Diffusion of epicenters of earthquake aftershocks, omori's law, and generalized continuous-time random walk models. *Phys. Rev. E*, 66:061104, 2002.

[15] Thibault Jaisson and Mathieu Rosenbaum. Limit theorems for nearly unstable hawkes processes. *The Annals of Applied Probability*, 25(2):600–631, 04 2015.

[16] Dmytro Karabash. On stability of hawkes process. 1 2013.

[17] Erik Lewis and George Mohler. A nonparametric em algorithm for multiscale hawkes processes. *Journal of Nonparametric Statistics*, 1(1):1–20, 2011.

[18] Rafael Lima and Jaesik Choi. Hawkes process kernel structure parametric search with renormalization factors, 2018.

[19] Scott W. Linderman and Ryan P. Adams. Discovering latent network structure in point process data. In *Proceedings of the International Conference on Machine Learning*, pages 1413–1421, 2014.

[20] Yanchi Liu, Tan Yan, and Haifeng Chen. Exploiting graph regularized multi-dimensional hawkes processes for modeling events with spatio-temporal characteristics. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, page 2475–2482. AAAI Press, 2018.

[21] Hongyuan Mei and Jason Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, 2017.

[22] George O. Mohler, Martin B. Short, P. Jeffrey Brantingham, Frederic P. Schoenberg, and George E. Tita. Self-exciting point process modelling of crime. *Journal of the American Statistical Association*, 106(493):100–108, 2012.

[23] Yosihiko Ogata. On lewis' simulation method for point processes. *IEEE Transactions on Information Theory*, 27(1):23–31, 1981.

[24] Yosihiko Ogata. Seismicity analysis through point-process modelling: A review. *Pure and Applied Geophysics*, 155(5):471–507, 1999.

[25] Takahiro Omi, Naonori Ueda, and Kazuyuki Aihara. Fully neural network based model for general temporal point processes. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 2120–2129, 2019.

[26] T. Ozaki. Maximum likelihood estimation of hawkes' self-exciting point processes. *Annals of the Institute of Statistical Mathematics*, (31):145–155, 1979.

[27] Marian-Andrei Rizoiu, Young Lee, and Swapnil Mishra. Hawkes processes for events in social media. In *Frontiers of Multimedia Research*, pages 191–218. 2018.

[28] Marian-Andrei Rizoiu, Swapnil Mishra, Quyu Kong, Mark James Carman, and Lexing Xie. Sir-hawkes: Linking epidemic models and hawkes processes to model diffusions in finite populations. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, pages 419–428. ACM, 2018.

[29] Farnood Salehi, William Trouleau, Matthias Grossglauser, and Patrick Thiran. Learning hawkes processes from a handful of events. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 12694–12704, 2019.

[30] Jin Shang and Mingxuan Sun. Geometric hawkes processes with graph convolutional recurrent neural networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 4878–4885. AAAI Press, 2019.

[31] Christian Shelton, Zhen Qin, and Chandini Shetty. Hawkes process inference with missing data, 2018.

[32] William Trouleau, Jalal Etesami, Matthias Grossglauser, Negar Kiyavash, and Patrick Thiran. Learning hawkes processes under synchronization noise. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 6325–6334. PMLR, 2019.

[33] Haoyun Wang, Liyan Xie, Alex Cuozzo, Simon Mak, and Yao Xie. Uncertainty quantification for inferring hawkes networks, 2020.

[34] Shuai Xiao, Hongteng Xu, Junchi Yan, Mehrdad Farajtabar, Xiaokang Yang, Le Song, and Hongyuan Zha. Learning conditional generative models for temporal point processes. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 6302–6310, 2018.

[35] Shuai Xiao, Junchi Yan, Changsheng Li, Bo Jin, Xiangfeng Wang, Xiaokang Yang, Stephen M. Chu, and Hongyuan Zha. On modeling and predicting individual paper citation count over time. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2676–2682, 2016.

[36] Hongteng Xu, Mehrdad Farajtabar, and Hongyuan Zha. Learning granger causality for hawkes processes. In *Proceedings of the International Conference on Machine Learning*, pages 1717–1726, 2016.

[37] Yingxiang Yang, Jalal Etesami, Niao He, and Negar Kiyavash. Online learning for multivariate hawkes processes. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 4944–4953, 2017.

[38] Qiang Zhang, Aldo Lipani, Ömer Kirnap, and Emine Yilmaz. Self-attentive hawkes processes. *CoRR*, abs/1907.07561, 2019.

[39] Rui Zhang, Christian J. Walder, and Marian-Andrei Rizoiu. Sparse gaussian process modulated hawkes process. *CoRR*, abs/1905.10496, 2019.

[40] Rui Zhang, Christian J. Walder, Marian-Andrei Rizoiu, and Lexing Xie. Efficient non-parametric bayesian hawkes processes. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 4299–4305. ijcai.org, 2019.

[41] Qingyuan Zhao, Murat A. Erdogdu, Hera Y. He, Anand Rajaraman, and Jure Leskovec. Seismic: A self-exciting point process model for predicting tweet popularity. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1513–1522, 2015.

[42] Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer hawkes process. *CoRR*, abs/2002.09291, 2020.