CentraleSupélec

université PARIS-SACLAY

The Whittle estimation for Hawkes processes

Mathematical modelling of complex systems

Written by

Tomas España
Clément Simon
Yecine Ktari
Ameur Echaabi

Supervising professor

Ioane Muni Toke

Acknowledgements

We are very grateful to Ioane Muni Toke for his careful help, comments and suggestions all along the development of this paper. All remaining errors are our own.

28 mai 2022

Table des matières

Introduction

Abstract

Estimating parameters of a model is a key issue of quantitative finance. In this paper, we focus on the Whittle estimation for Hawkes processes proposed by Félix Cheysson et Gabriel Lang [4] in 2021. Our main goal is to implement a Python code for the Whittle estimation and reach comparable execution times compared with their *hawkesbow* library written in R code. We also studied the speed convergence of this method. Finally, we compared the influence of measurement inaccuracy by randomizing simultaneous events, for both MLE and Whittle estimation. All our code is gathered in the following Gitlab

Introduction

Hawkes processes are self-exciting processes that are used to model many phenomena in all sorts of scientific fields. The strength of this discrete model comes from the fact that the happening of an event is influenced by past events. This is why, originally, Hawkes processes were introduced to model the aftershocks of earthquakes : aftershocks are more likely to occur a short time after the earthquake than a long time after. Hawkes processes are also used in neuroscience, social media, statistical finance and even for predicting terrorist attacks[1]. The R library *hawkesbow* [2] is dedicated to Hawkes processes and also includes two of the most common estimation methods : the Maximum Likelihood Estimation (MLE) and the Whittle estimation. Our objective is to implement these two estimation methods in Python code as a complement to the *tick.hawkes* library[3]. Our interest for the Whittle estimation comes from the fact that it doesn't need the exact location of events in time, unlike MLE. Counting how many events took place in a bin is enough to estimate the parameters of the Hawkes process. This can come in very handy when time precision isn't good enough to dissociate several events very close in time. This is a real issue for quantitative analysts today working with different time frames. Indeed, if the lack of time precision makes several events coincide at the same time when in fact they don't, then the estimates are false. A solution, other than using the Whittle estimation, consists in adding a very little random time of occurrence to each of these coinciding events in order to separate them. In this case, the MLE method can still be used. In this paper, we exclusively focus on exponential kernels for Hawkes processes of dimension 1. In this specific case, MLE estimation has a linear complexity while the Whittle estimation has a $O(nlog(n))$ complexity, where n is the number of bins. However, with other kernels (Gaussian or Powerlaw), complexity becomes quadratic for MLE and remains unchanged for Whittle (see section "Powerlaw" of the Jupyter notebook for further details). Nonetheless, a fast parametric estimation of Hawkes processes is possible by constructing an adaptive stratified sampling estimator of the gradient of the Least Square Error (LSE) method[10]. Other methods that consist in computing the moments of the Hawkes processes [5] can yield fast estimations.

To what extent are Hawkes processes relevant to fit financial data ? The aim of the paper [6] is to discuss the difficulty in obtaining significant fits when testing the explanatory power of Hawkes processes. The author argues that this is due in part to discontinuities in trading activity, such as those that occur when markets open and close. The author suggests that working with data from FX markets, which operate continuously for longer periods, may be more successful in obtaining significant fits. The study found that Hawkes processes can be used to model a large number of events (around 15000 on average) with a high degree of accuracy. The study also found that the fitted kernels are mostly Power-law like, with a few exponentials added in to improve the fit.

Thus, Hawkes processes seem very legitimate to model specific financial data. This paper [8] also suggests this. It discusses the relevance of trades-through in market order model and describes how these can be well-modeled using Hawkes processes, and how this can help to explain periods of high liquidity consumption in order books. In sum, Hawkes processes are very relevant to model financial data and it is their self-exciting aspect that make them so powerful.

Thesis statement

Our work is mainly inspired and based on the paper written by Felix Cheysson, Gabriel Lang [4] which studies the Whittle estimation for Hawkes processes. The goal of this paper is to verify the results from a numerical standpoint. The authors propose a Whittle estimation procedure for stationary Hawkes processes from their count data. They show that this approach has appealing features : it has good asymptotic properties, similar to the one Maximum Likelihood Estimation (MLE) possesses and it is computationally efficient ($O(nlog(n))$ complexity).

This paper attempts to implement this method in Python, build simulations to verify these results, and compare this new method to the known Maximum Likelihood Estimation method.

Chapitre 1

Simulation of a Hawkes process

The expression of the intensity λ of a Hawkes process can be written as :

$$\lambda(t) = \mu + \sum_{ti<t} \alpha\beta e^{-\beta(t-t_i)}$$

1.1 Interpretation of parameters

In order to study the impact of the parameters on the intensity λ, we have simulated several processes thanks to the tick.hawkes library [3], by varying a parameter and keeping the two others constant.

— **Influence of the parameter** μ

$$\lambda(t) = {\color{red}\mu} + \sum_{ti<t} \alpha\beta e^{-\beta(t-t_i)}$$

The parameter μ represents the basic intensity of the process. It is independent of past events. Both the quantity of events and the mean value of intensity peaks rise with this parameter. This is what we observe on the following graphs that illustrate the evolution of the intensity λ with respect to time when μ increases (with $\alpha = 0.6$ and $\beta = 0.8$). We recall that the x-axis represents time and the y-axis the intensity λ.

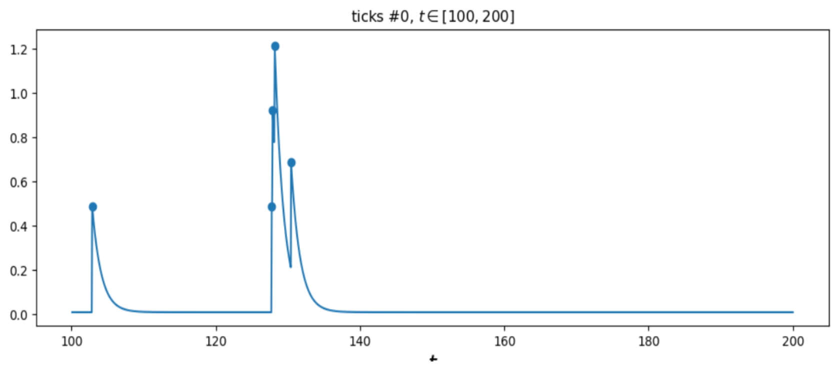

FIGURE 1.1 – $\mu = 0.01$

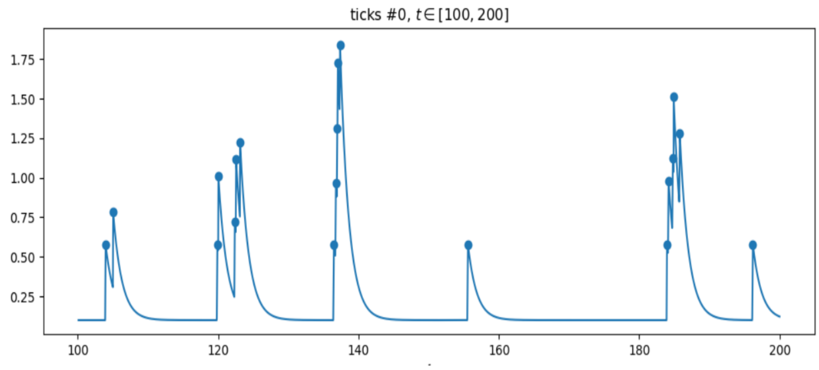

FIGURE 1.2 – $\mu = 0.1$

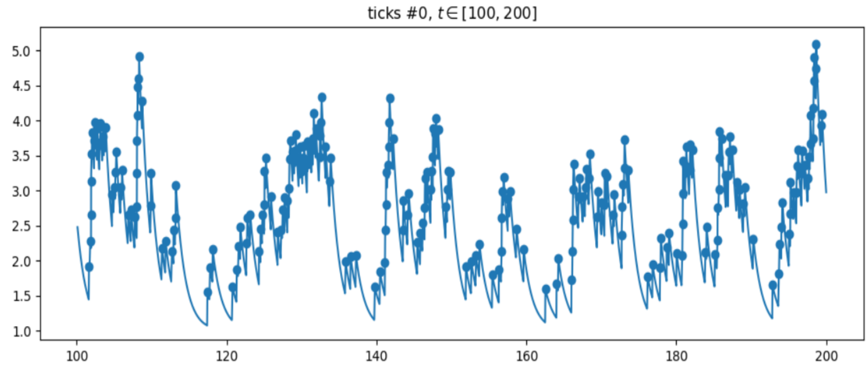

FIGURE 1.3 – $\mu = 1$

— **Influence of the parameter α**

$$\lambda(t) = \mu + \sum_{ti<t} \alpha \beta e^{-\beta(t-t_i)}$$

The parameter α is the multiplicative coefficient in front of the sum that increases the intensity. The height of each peak is proportional to its value. This is what we observe in the following graphs (with $\mu = 0.1$ and $\beta = 1.2$).

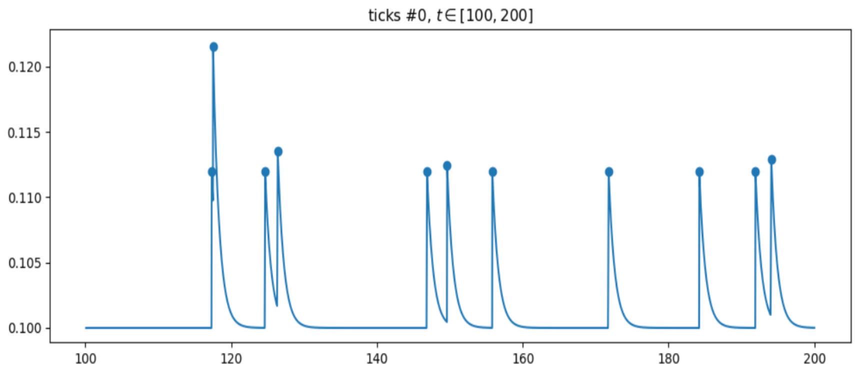

FIGURE 1.4 – $\alpha = 0.01$

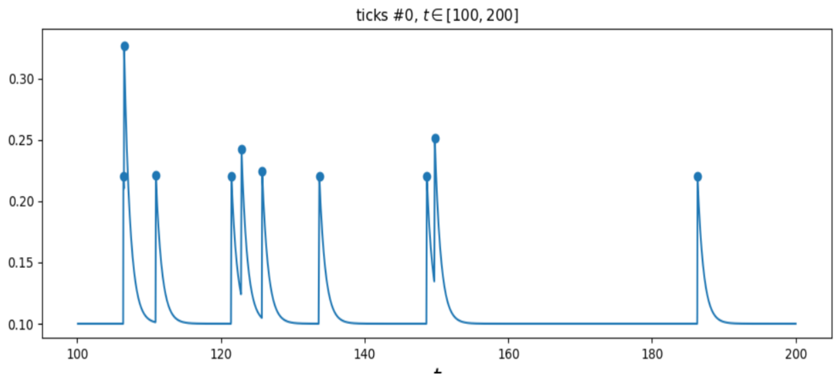

FIGURE 1.5 – $\alpha = 0.1$

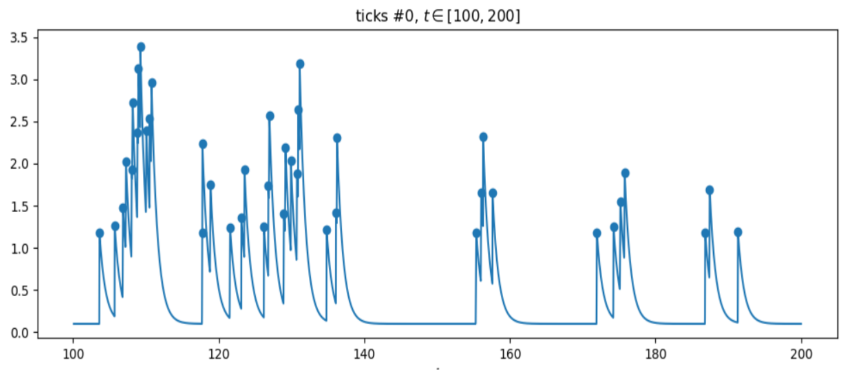

FIGURE 1.6 – $\alpha = 0.9$

— **Influence of the parameter β**

$$\lambda(t) = \mu + \sum_{ti<t} \alpha\beta e^{-\beta(t-t_i)}$$

The parameter β appears in the decreasing exponential like a multiplicative coefficient in front of the difference between the current time and the times of past events. Thus, β modifies the speed of the decreasing of the exponential : the more β increases, the more the intensity λ decreases fast. This is what we observe in the following graphs (with $\mu = 0.1$ and $\alpha = 0.6$).

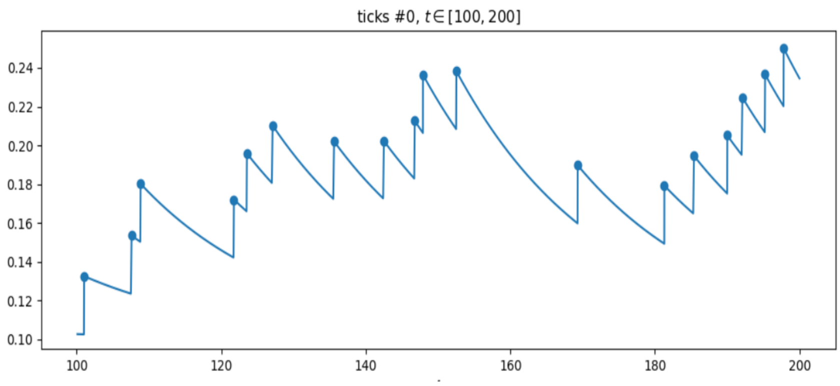

FIGURE 1.7 – $\beta = 0.05$

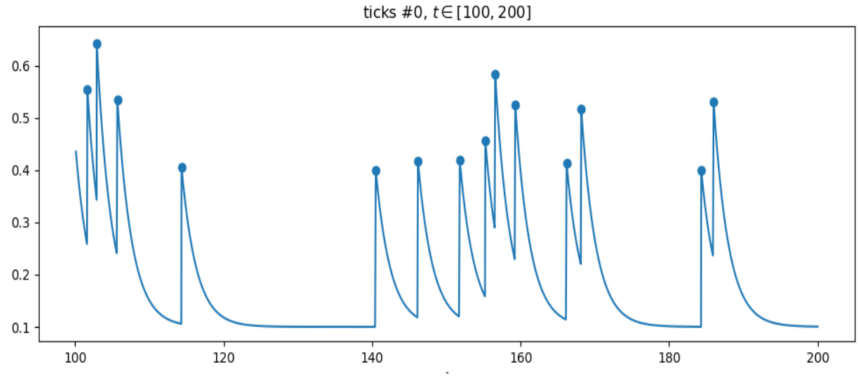

FIGURE 1.8 – $\beta = 0.5$

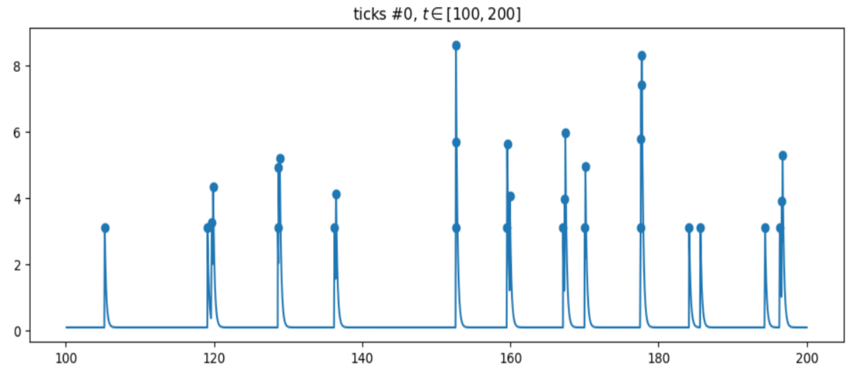

FIGURE 1.9 – $\beta = 5$

Chapitre 2

Estimation methods

In this chapter we explain the mathematical concepts of both MLE and Whittle estimation.

2.1 The Maximum Likelihood Estimation (MLE)

In this section we briefly expose some results of paper [7] written by Patrick Laub, regarding the log-likelihood for Hawkes processes.

Theorem

Let $N()$ be a regular point process on $[0,T]$ for some finite positive T, and let $t_1, ..., t_k$ denote a realisation of $N()$ over $[0,T]$. Then, the likelihood L of $N()$ can be written as :

$$L = \left(\prod_{i=1}^{k} \lambda(t_i) \right) \; exp\left(- \int_0^T \lambda(u)du \right)$$

The log-likelihood l on interval $[0,t_k]$ can thus be written as :

$$l = \sum_{i=1}^{k} log(\lambda(t_i) - \int_0^{t_k} \lambda(u)du$$

If the Hawkes prochess has (μ, α, β) as parameters, then we can transform the expression above to obtain :

$$l = \sum_{i=1}^{k} log(\lambda + \alpha\beta \sum_{j=1}^{i-1} e^{-\beta(t_i - t_j)}) - \lambda t_k + \alpha \sum_{i=1}^{k} (e^{-\beta(t_k - t_i)} - 1)$$

The expression obtained shows that a computational approach to compute l implies a $\mathcal{O}(k^2)$ complexity which would need a lot of time. Hopefully, we can re-write :

$$l = \sum_{i=1}^{k} log(\lambda + \alpha\beta A(i)) - \lambda t_k + \alpha \sum_{i=1}^{k} (e^{-\beta(t_k - t_i)} - 1)$$

where $A(i) = \sum\limits_{j=1}^{i-1} e^{-\beta t_i + \beta t_j}$

The sequence A can be defined recursively as follows (see section 2.3 for further details) :

$$A(i) = e^{-\beta(t_i - t_{i-1})}(1 + A(i-1))$$

Thus, the calculation of l can be achieved in $\mathcal{O}(k)$ complexity when a recursive approach is considered.

Remark

The linear complexity found above comes from the fact that Hawkes processes with exponential decay functions are actually Markov processes. As a consequence, the computation is more efficient. Thus, MLE should turn out to be an effective method for model fitting. However, Filimonov and Sornette [9] found that for some finite sample sizes, the estimator does have significant bias, encounters many local optima, and is highly sensitive to the selection of excitation function. This is the reason why, other estimation methods based on the moments of the point process were elaborated. These are arguably faster according to Da Fonseca and Zaatour [5].

2.2 The Whittle estimation

Whittle estimation is based on the minimization of the log-spectral likelihood of a process. It was introduced by Peter Whittle in 1951. Next, we introduce several mathematical notions to understand the foundations of this method.

Definition

The count series with binsize Δ associated to a point process N is the process $(X_t)_{t\in\mathbb{R}} = (N(t\Delta,$ (t+1)$\Delta])_{t\in\approx}$, generated by the count measure on intervals of size Δ.

Proposition

Let N be a stationary Hawkes process on R and $(X_t)_{t\in\mathbb{R}} = (N(t\Delta,$ (t+1)$\Delta])_{t\in\mathbb{R}})$ the associated count series. Then X_t has a spectral density function given by

$$f_{X_t}(w) = m\Delta sinc^2(\frac{w}{2}) \mid 1 - \tilde{h}(\frac{w}{\Delta}) \mid^2$$

Remark

The symbol $\overset{\sim}{.}$ denotes the Fourier transform :

$$\overset{\sim}{\phi}(w) = \int_{\mathbb{R}} e^{-iws}\phi(s)\,\mathrm{d}s$$

Corollary

Let N be a stationary Hawkes process on R and $(X_k)_{k\in\mathbb{Z}} = (N(k\Delta,$ (k+1)$\Delta])_{k\in\mathbb{Z}})$ the associated count series. Then X_k has a spectral density function given by $f_{X_k}(w) = \sum_{j\in\mathbb{Z}} f_{X_t}(w + 2k\pi)$ where f_{X_t} is the function defined in the previous proposition.

Remark

This corollary shows it is important to distinguish f_{X_k} when $k \in \mathbb{Z}$ and f_{X_t} when $t \in R$.

Understanding the method

The full description of a random process is generally a very complex task, especially if we deal with random vectors. That's why we characterize partially random processes using their first two moments : mean (order 1 moment) and auto-correlation (order 2 moment). For many applications, it is important to characterize a random process in terms of frequency. For instance, telecommunication signals are often modeled by random vectors and it is important to quantify the spectral band used by these signals to optimize transmission channels. The main tool that allows to characterize random stationary processes (such as Hawkes processes) is the *spectral density*. Thanks to the Wiener-Khinchin theorem, we know that the spectral density corresponds to the Fourier transform of the auto-correlation of the process.

Equation of the estimator

Let θ be an unknown parameter vector. In our case, we will have $\theta = (\mu, \alpha, \beta)$ where μ, α and β are the three parameters of our model we want to estimate. We note $I = (0, +\infty) \times (0, 1) \times (0, +\infty)$. For a stationary linear process $(X_k)_{k\in\mathbb{Z}}$ with spectral density f_θ, Peter Whittle proposed the following estimator :

$$\widehat{\theta_n} = \underset{\theta\in I}{argmin}\mathcal{L}_n(\theta)$$

where $\mathcal{L}_n(\theta) = \frac{1}{4\pi}\int_{-\pi}^{\pi}(log\ \mathrm{f}_\theta(w) + \frac{I_n(w)}{f_\theta(w)})\mathrm{d}w$ is the log-spectral likelihood of the process and $I_n(w) = \frac{1}{2\pi n} \mid \sum_{k=1}^{n} X_k e^{-ikw} \mid^2$ is the periodogram of the partial realisation $(X_k)_{1\leq k\leq n}$. We remind that the spectral density τ_X of random stationary process X_n is defined by :
$\forall v \in \mathbb{R},\ \tau_X(v) = \underset{N\to+\infty}{lim}\ \mathbb{E}(\frac{1}{2N+1} \mid \sum_{k=-N}^{N} X_k e^{-ikw} \mid^2)$
Thus, the periodogram can be interpreted as an estimate of the spectral density when $N \to +\infty$.

Remark

We can prove that the estimator $\widehat{\theta_n}$ is asymptotically normal and has a speed convergence in $n^{-\frac{1}{2}}$. This means that the mean square error $\|\ \widehat{\theta_n} - \theta_0\ \|_2^2$ tends to a linear function of slope -1 when $N \to +\infty$ in log-log scale. Note that θ_0 is the real set of parameters of the Hawkes process. We will illustrate this convergence speed with numeric simulations later on.

Conclusion

MLE and Whittle estimation are two powerful methods for estimating the parameters of a Hawkes process. However, we have to keep in mind the main difference between both. On the one hand, MLE uses the exact timestamps of events to perform its estimate. On the other hand, Whittle estimation only needs the count series as input to compute an estimation. Concretely, this comes in very handy when time precision of machines isn't precise enough to differentiate several timestamps that are too close in time. We will study this particularity of the Whittle estimation in the last section of our work.

2.3 Time complexity

The Whittle estimation procedure is computationally interesting, with a complexity in $O(nlogn)$ where n is the number of bins (this comes from the computation of the fast Fourier transform). While the complexity is $O(k^2)$, k the number of events in the process, for the Maximum Likelihood method (except when the kernel is exponential, in which case the complexity is reduced to O(k) with minimal efforts (Ozaki and Ogata, 1979), making it more efficient than the Whittle approach).

Let's focus on simplifications for exponential decay where the Maximum Likelihood method becomes more efficient than the Whittle estimation. We recall that the log-likelihood function used in the Maximum Likelihood estimation is :

$$l = \sum_{i=1}^{k} log(\lambda + \alpha\beta \sum_{j=1}^{i-1} e^{-\beta(t_i - t_j)}) - \lambda t_k + \alpha \sum_{i=1}^{k}(e^{-\beta(t_k - t_i)} - 1)$$

This direct approach has high computation complexity. The term with double summation implies $O(k^2)$ complexity. Fortunately, the structure implies simplifications for exponential decay and allows l to be computed with $O(k)$ complexity Let's note the inner summation, for $i \in 2, ..., k$, as

$$A(i) = \sum_{j=1}^{i-1} e^{-\beta(t_i - t_j)}$$

This can be defined recursively using previous terms :

$$
\begin{aligned}
A(i) &= \sum_{j=1}^{i-1} e^{-\beta t_i + \beta t_j} && (2.1)\\
&= e^{-\beta t_i + \beta t_{i-1}} e^{\beta t_i - \beta t_{i-1}} \sum_{j=1}^{i-1} e^{-\beta t_i + \beta t_j} && (2.2)\\
&= e^{-\beta t_i + \beta t_{i-1}} \sum_{j=1}^{i-1} e^{-\beta t_{i-1} + \beta t_j} && (2.3)\\
&= e^{-\beta (t_i - t_{i-1})} (1 + \sum_{j=1}^{i-2} e^{-\beta (t_{i-1} - t_j)}) && (2.4)\\
&= e^{-\beta (t_i - t_{i-1})} (1 + A(i-1)) && (2.5)
\end{aligned}
$$

This shows that, with other kernels, this simplification would not be possible and the Maximum Likelihood computation complexity would be quadratic. The Whittle estimation, however, always has a $O(nlog(n))$ complexity.

Chapitre 3

Simulation study

In this section, we illustrate the estimation procedure and asymptotic properties of the spectral approach for Hawkes count series in order to proof-read the numerical results from the original article [4]. We will exclusively consider exponential kernels (see expression in Chapter 1).

To do so, we started rewriting the Whittle R library *hawkesbow* [2] into Python. To build such a library in Python, we needed a few modules :

— A Hawkes process generation module (we used the very powerful tick.hawkes library [3]).
— The pre-processing Count-series function (called "discrete")
— A periodogram (using the Fast Fourier Transform module from NumPy library)
— The log-spectral likelihood (as defined in 2.2)
— A global optimizer (L-BFGS)

3.1 Comparison of time execution

In this section, we compare the execution time behavior against two different sets of parameters for the Whittle estimation : Δ and T

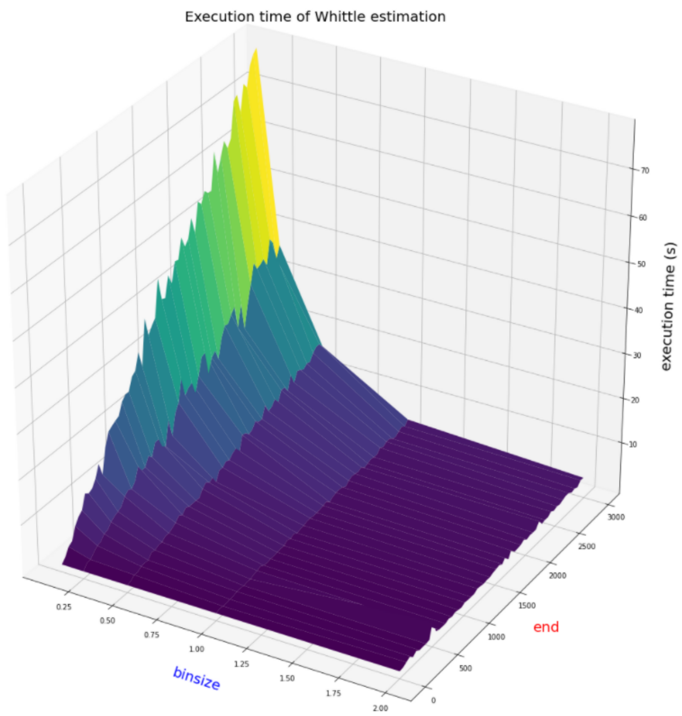

FIGURE 3.1 – Execution time of the Whittle function against Δ and T with the Python library

For more information on the execution time of our different components (Hawkes process generation using *tick.hawkes*, discrete function, periodograms, etc.), we present a few graphs in the Gitlab. We found that their runtimes are negligible compared to the Whittle optimization sequence.

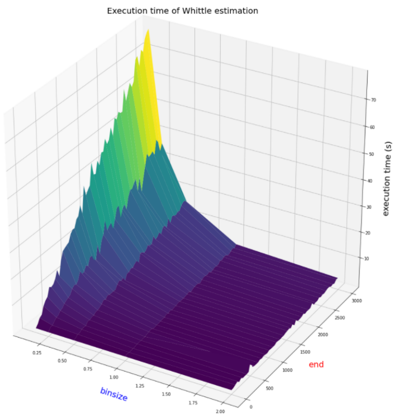

FIGURE 3.2 – Python - *our algorithm*

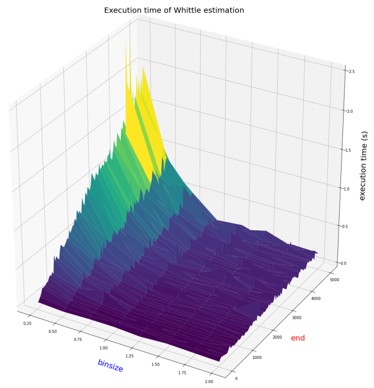

FIGURE 3.3 – R - *hawkesbow*

This graph provides enough evidence that the execution time increases *linearly* as $T \to \infty$ and *exponentially* when $\Delta \to 0$.

If the execution time heavily depends on the machine we used (Intel i5), comparing the different methods and scripts should give more rational insight on how they perform *relative to each other* :

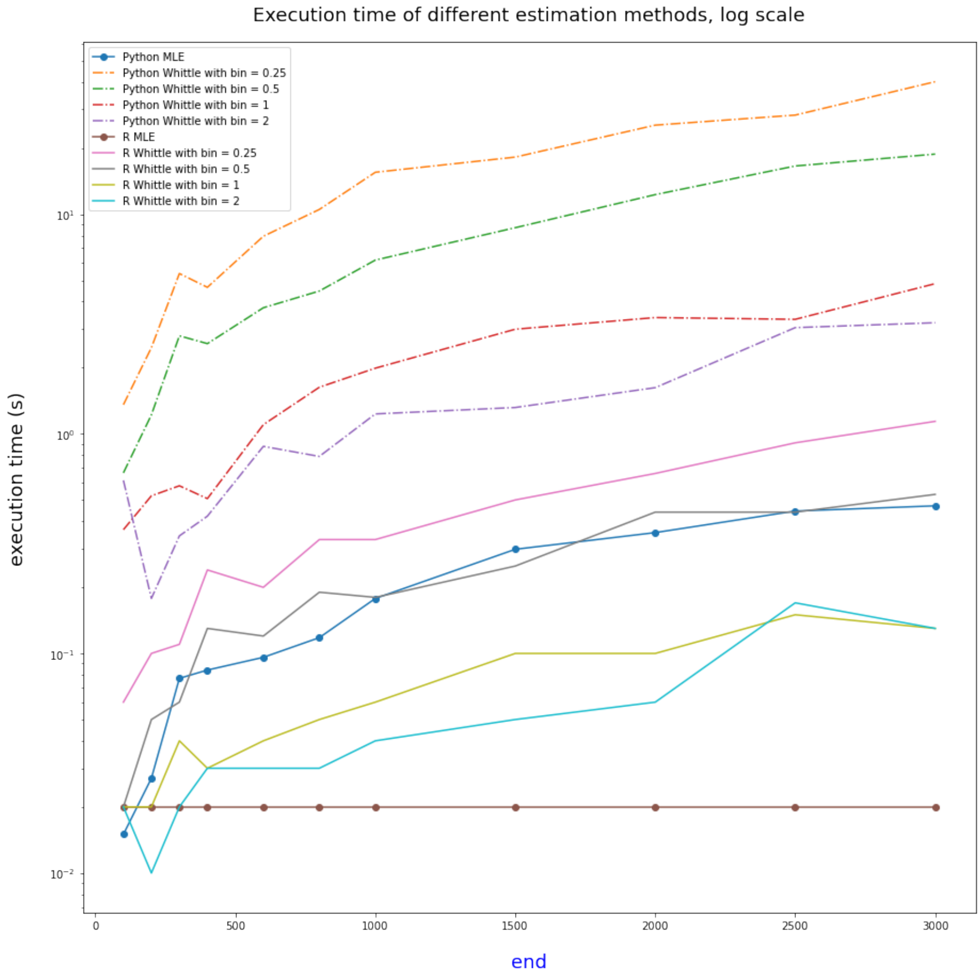

FIGURE 3.4 – Execution time comparison between the different algorithms and libraries

3.2 Comparison of precision

3.2.1 Influence of T (Horizon) on precision

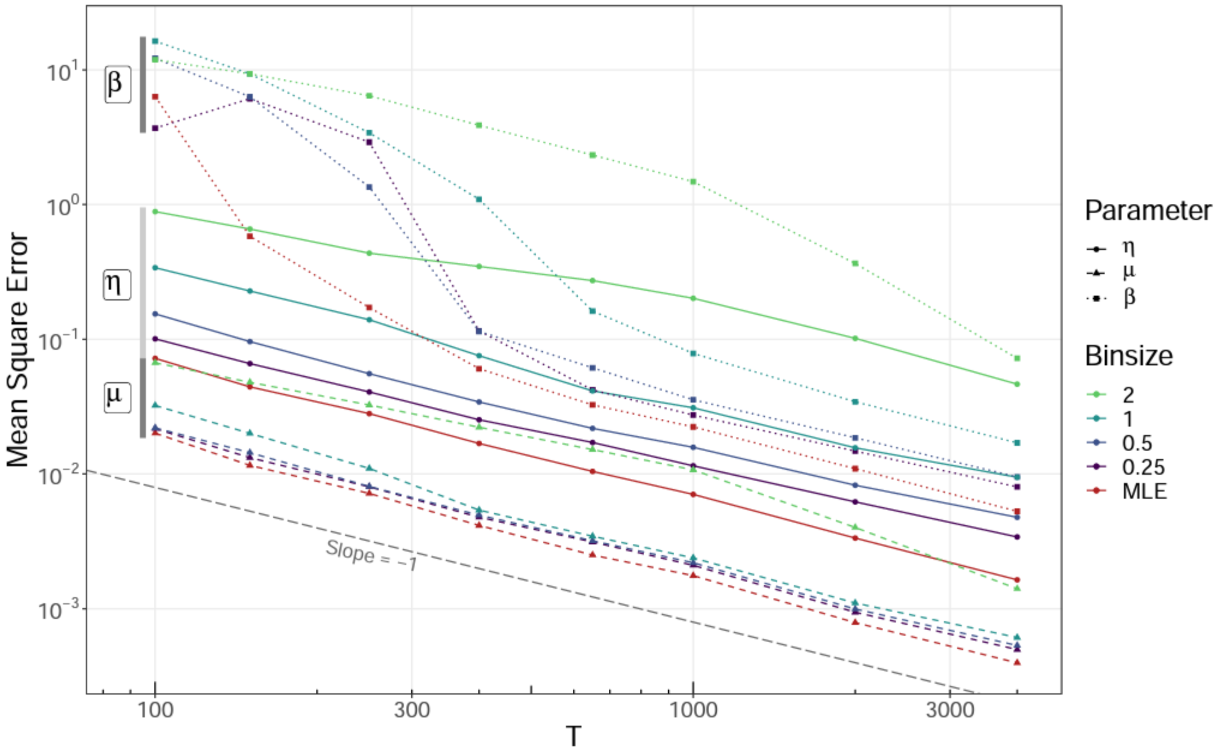

FIGURE 3.5 – **Results from the pre-print** [4] : Mean square error of the estimates of parameters μ, α and β for 1,000 simulations of the stationary Hawkes process with kernel $h(t) = \alpha\beta e^{-\beta t}$ on the interval [0, T], in log-log scale. The thick dashed black line represents the ideal slope of 1, i.e. a rate of convergence of $\mathcal{O}(n^{-1})$.

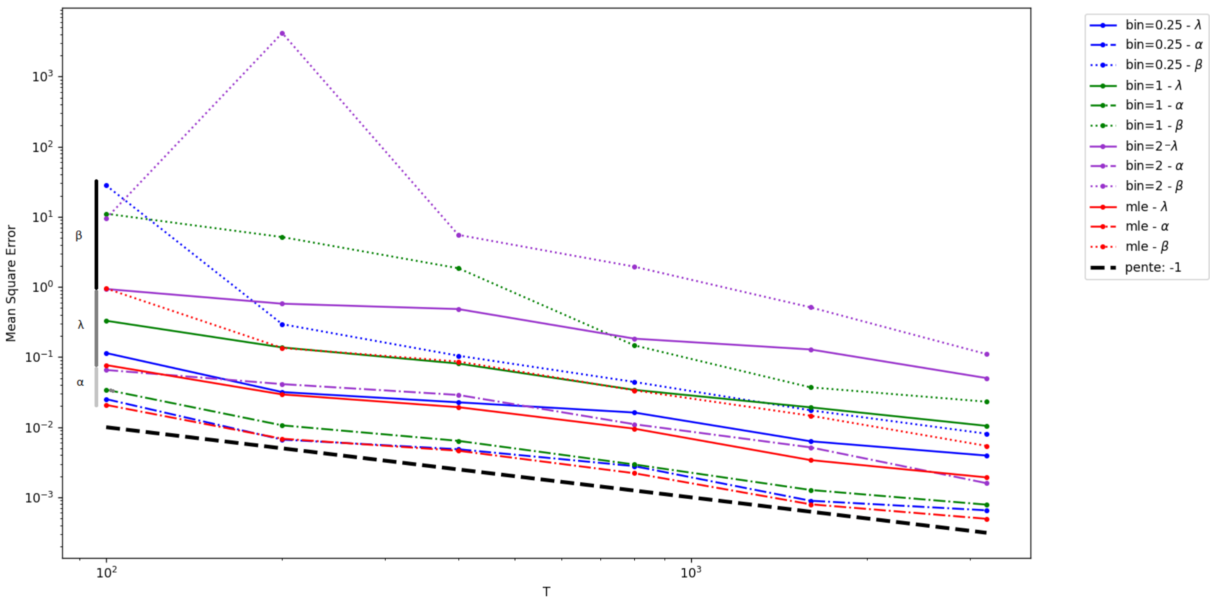

FIGURE 3.6 – **Our results** : Mean square error of the estimates of parameters μ, α and β for 100 simulations of the stationary Hawkes process with kernel $h(t) = \alpha\beta e^{-\beta t}$ on the interval [0, T], in log-log scale. The thick dashed black line represents the ideal slope of 1, i.e. a rate of convergence of $\mathcal{O}(n^{-1})$.

Therefore, **we can confirm the numerical results found in the paper** which assess a rate of convergence of $\mathcal{O}(n^{-1})$. However, increasing the number of simulations should yield more accurate convergence behavior when $T \to \infty$.

3.2.2 Influence of Δ (Binsize) on precision

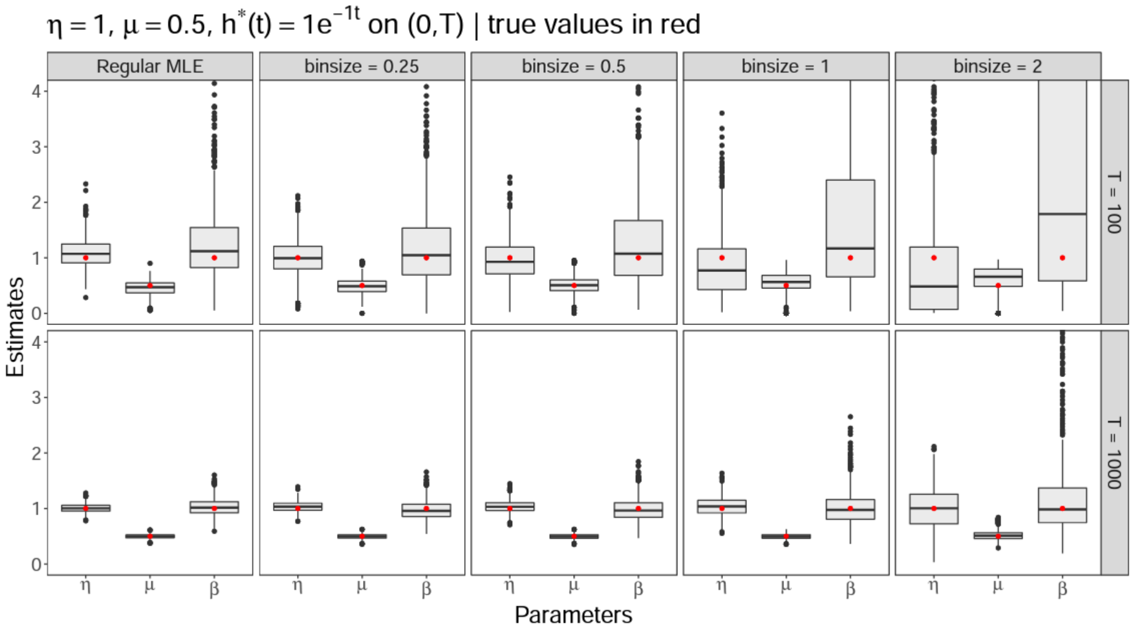

FIGURE 3.7 – **Results from the pre-print** [4] : Boxplot of estimates of parameters μ, α and β for 1,000 simulations of the stationary Hawkes process with kernel $h(t) = \alpha\beta e^{-\beta t}$ on the interval [0, T]. True values (blue crosses) are : $\mu = 1, \alpha = 0.5, \beta = 1$.

We simulated 100 realizations of the Hawkes process on the interval [0,T] with parameter values $\mu = 1$, $\alpha = 0.5$ and $\beta = 1$ and counted the events in bins of size $\Delta = 0.25, 1$ or 2 respectively. We then estimated the parameters μ, α and β. We compared these estimates to the usual maximum likelihood estimates (MLE).

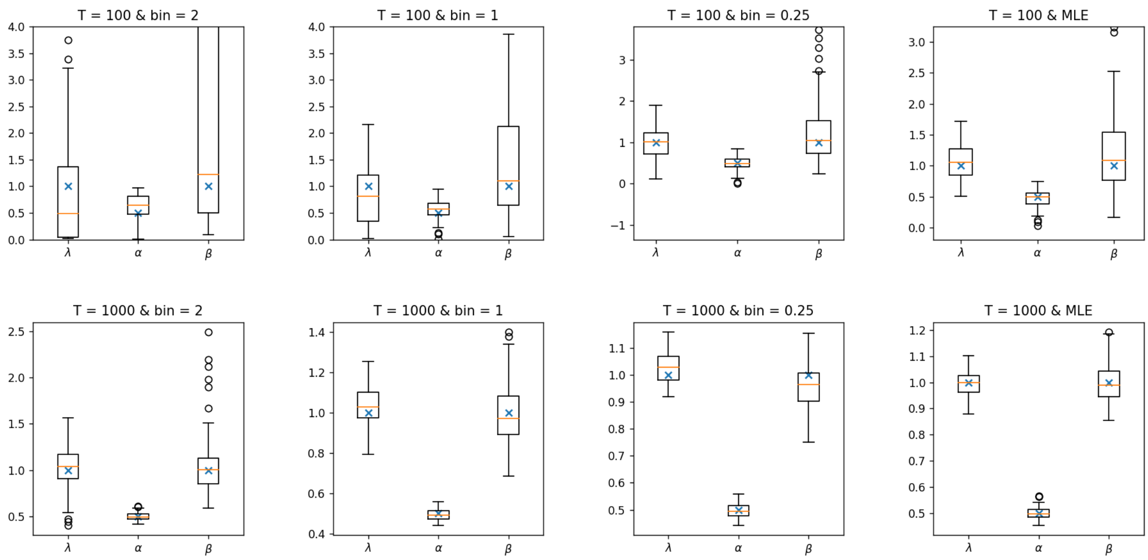

FIGURE 3.8 – **Our results** : Boxplot of estimates of parameters μ, α and β for 100 simulations of the stationary Hawkes process with kernel $h(t) = \alpha\beta e^{-\beta t}$ on the interval [0, T]. True values (blue crosses) are : $\mu = 1, \alpha = 0.5, \beta = 1$.

For small binsizes, the Whittle estimation performs almost as well as the MLE estimation. The accuracy deteriorates massively for higher binsizes, notably for the exponential kernel rate β. This is intuitive, since large binsizes with respect to the kernel scale make it difficult to detect interactions between points. Since the MLE method uses precise information on the location of events, instead of counting the events in a bin for the Whittle estimation, its precision is arguably better that any estimate based on the count series. Furthermore, it provides a best case scenario for the Whittle estimates when the binsize tends to 0. However, as we will see next, the precision of the estimation deteriorates when Δ increases.

To further illustrate the asymptotic properties of the estimation, notably its rate of convergence, we compute the mean square error (MSE) for the estimates of each set of 100 simulations with a given T and binsize Δ (Figure 2.1). For large values of , the slope of the mean square error with respect to T reaches 1 (in log-log scale) for all parameters and almost all binsizes. This illustrates the $\mathcal{O}(n^{-1})$ rate of convergence. For small values of T and for both estimation methods, the estimates of the immigration intensity μ and reproduction mean α have already reached the optimal rate of convergence. The purple peak at small values of T for an estimate of β when $\Delta = 2$ shows that the Whittle estimation isn't reliable and precise enough for big bins and small Ts. Finally note that, for reasonable binsizes ($\Delta \leq 1$), the Whittle estimates of the reproduction mean α have a MSE comparable to those of the maximum likelihood.

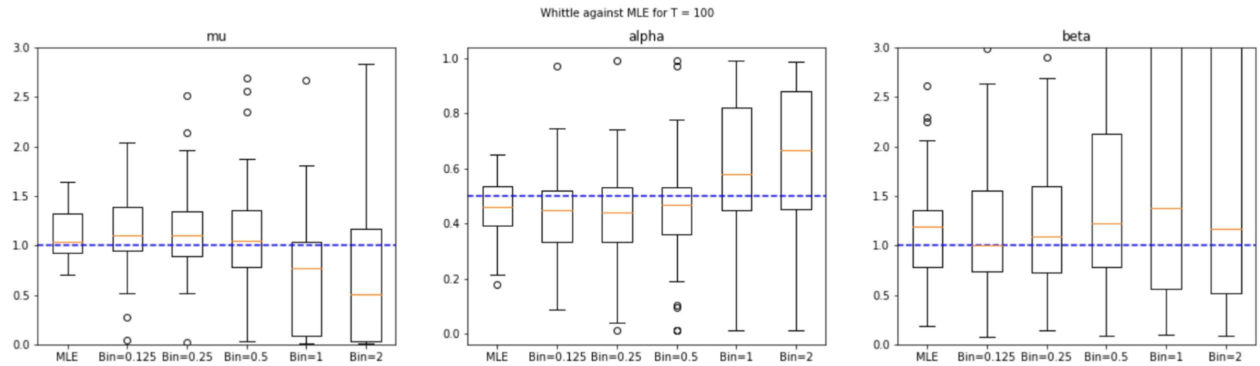

FIGURE 3.9 – **Our results** : Boxplot of estimates of parameters μ, α and β for 50 simulations of the stationary Hawkes process with kernel $h(t) = \alpha\beta e^{-\beta t}$ on the interval [0, 100]. True values (blue dotted line) are : $\mu = 1, \alpha = 0.5, \beta = 1$

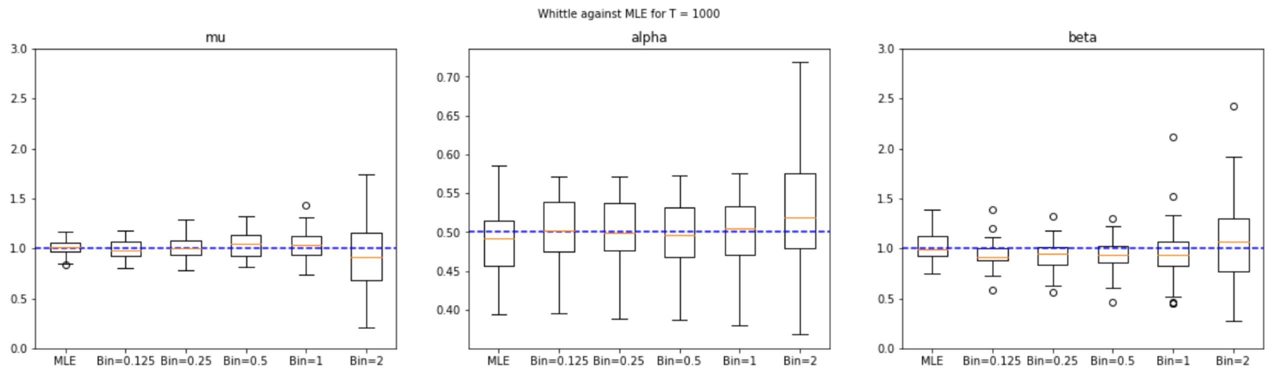

FIGURE 3.10 – **Our results** : Boxplot of estimates of parameters μ, α and β for 50 simulations of the stationary Hawkes process with kernel $h(t) = \alpha\beta e^{-\beta t}$ on the interval [0, 1000]. True values (blue dotted line) are : $\mu = 1, \alpha = 0.5, \beta = 1$

As always, a trade-off has to be found between precision and execution time. Based on our simulations, the value $\Delta = 0.5$ guarantees optimal precision (cf. Q1 and Q3 on the box-plots above).

3.3 Quality of estimation after data degradation

The Maximum Likelihood Estimation method was applied to the Hawkes process by Ozaki (1979) and improved by his recursive version featuring an efficient method for calculating the derivatives and the Hessian matrix. Consistency, asymptotic normality and efficiency of the estimator were proved by Ogata in 1978. On paper, MLE should be a very effective option for model fitting. But, it could produce significant bias, encounter many local optima and is highly sensitive to the selection of excitation function.

While MLE uses the timestamps with the exact order of events occurring to fit the model, the Whittle estimation only needs the count series as an input. Therefore, Whittle estimation can be particularly useful when time precision of machines isn't precise enough to differentiate several timestamps. Simulating this imprecision in the data is not an easy task.

Our strategy used to randomize data and test the ability of each method was the following : set the size of a bin and uniformly distribute events on the bin to take into consideration the minimum time lapse needed to differentiate between two non-simultaneous events.

Given the count series, **our hypothesis is that the Whittle estimation clearly should not be affected by the events getting shifted in time as much as MLE, since it gathers the total number of events happening in a bin**.

The figures on the next page challenge this hypothesis and show a comparison of the mean squared error (MSE) of each estimation as a function of the binsize after randomization of the input data :

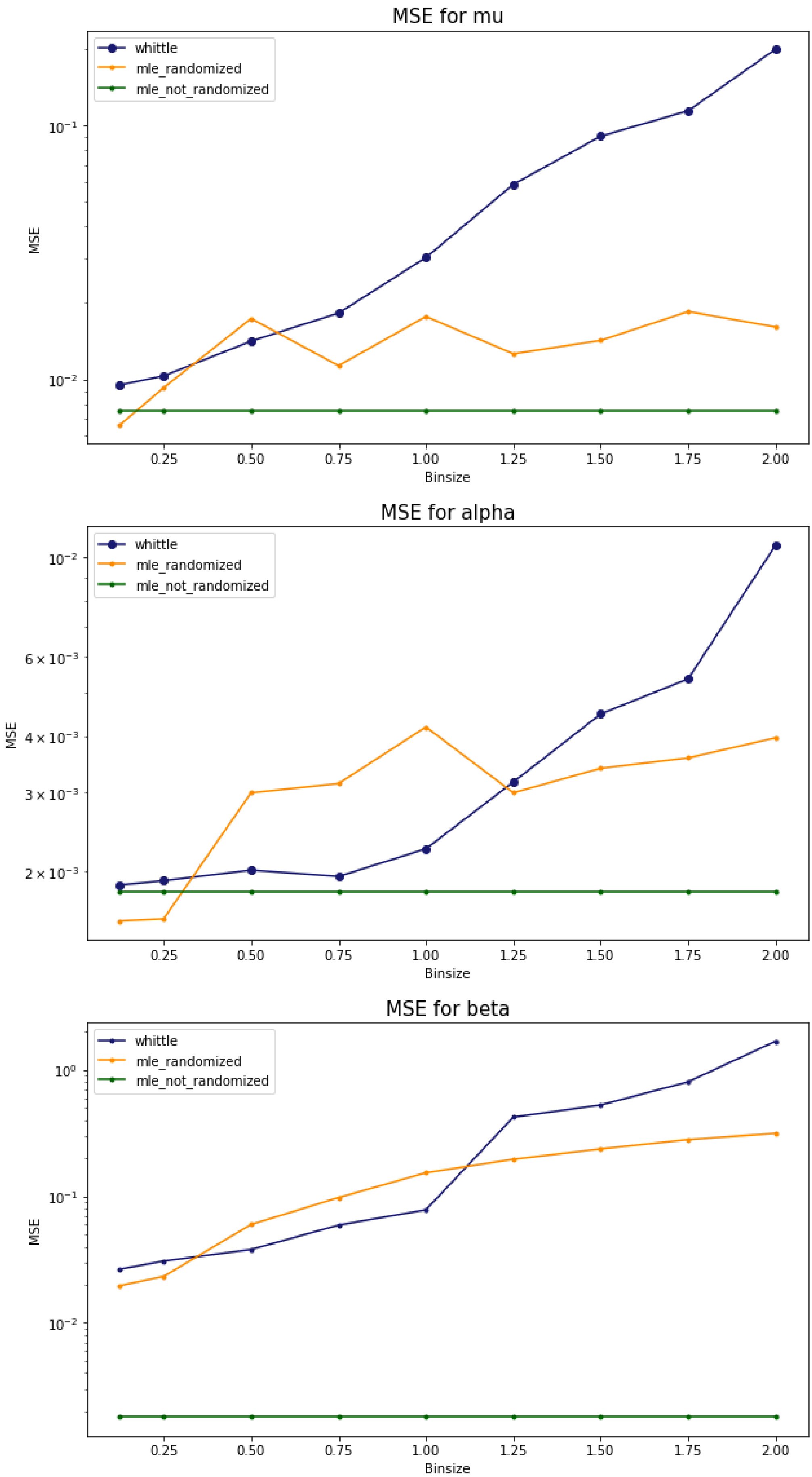

FIGURE 3.11 – Y-Log-scale Mean Square Error for 200 simulations of the stationary Hawkes process with kernel $h(t) = \alpha\beta e^{-\beta t}$ on the interval [0, 1000]. For comparison, the original MLE estimation is also displayed. True values are : $\mu = 1, \alpha = 0.5, \beta = 1$

As we notice in the figures, MLE remains more precise compared with the Whittle estimation for μ and when the binsize is relatively small ($\Delta \leq 0.5$). Computing power limits did not enable us to make larger simulations and to compute the mean squared error on more than 200 processes with different values of T. Nevertheless, we do see that as $\Delta \to 0$, both MLE and Whittle behave similarly.

For the estimation of α and β parameters, things get more interesting, though. For small values of binsize, as the degradation of the timestamps remains negligible in the bin, MLE is still more efficient. Yet, as the binsize increases, the mean squared error becomes higher for MLE than for Whittle, due to the randomization that significantly alters the original data : MLE estimation becomes biased. This is not an issue for Whittle, however, because the count-series is resilient to data randomization.

This shows clear evidence that the Whittle estimation can yield better estimation results when dealing with high frequency ticks, and thus confirms our hypothesis.

As expected, it should be noted that the relative error distance between the randomized MLE and Whittle behaves overall as a "smile-curve" ; indeed, as binsize increases, the edge of Whittle on MLE vanishes (as seen in 3.2.2). From this, we can make the assumption that $\Delta = 0.5$ is the best compromise between computing efficiency (3.1), precision (3.2) and data degradation for $T = 1000$ and $(\mu, \alpha, \beta) = (1, 0.5, 1)$.

Remark

In the future, other values, especially critical ones ($\frac{\alpha}{\beta} \to 1$) should also be tested.

In these simulations, only the exponential kernel was used. Our intuition is that MLE is highly dependent on the selection of self-exciting kernels, so the expected efficiency of the Whittle estimation after data degradation could be more accurate with other kernels (like Powerlaw for instance).

Again, increasing the number of simulations should yield better results and compensate for rare, degenerate realizations of Hawkes processes.

Chapitre 4

Conclusion

Our work consisted in studying the Whittle estimation for Hawkes processes and trying to determine to what extent it could be interesting compared to the powerful Maximum Likelihood Estimation method (MLE).

It is important to remind that the MLE, which needs the precise timestamps of events to work, can turn out to be unusable when precision on measurements is not enough to differentiate two events that are very close. This is the reason why, if time precision comes to lack, the Whittle estimation can come in handy because it is based on count series only. Also, in terms of complexity, the Whittle method is the most interesting when we do not have an exponential kernel, since the complexity is reduced to $O(nlog(n))$ (where n is the number of bins) instead of a quadratic complexity $O(n^2)$ for MLE. For more details, see the "Powerlaw" section of our Jupyter notebook.

During our study we performed several simulations using exponential kernels only, in both Python and R code, to highlight the asymptotic properties of the Whittle method and validate several results found by Felix Cheysson and Gabriel Lang [4]. We showed that the time of execution of the Whittle estimation increases linearly when we increased the time horizon T of our measurement and exponentially when the size of the bin decreased (see Figure 3.1). We also focused on the Mean Squared Error (MSE) of our estimations for both Whittle and MLE. The results obtained in Figure 3.6 confirm the convergence rate of the MSE is in $\mathcal{O}(n^{-1})$. Finally, we measured the influence of the binsize on the accuracy of the estimate. In accordance with our intuition, we found that the accuracy of the Whittle estimation increases when the binsize decreases and is comparable to that of the MLE estimation as the binsize tends to zero. However, the execution time in this case is much greater for the Whittle estimate. But, by increasing the binsize, we deteriorate the quality of the bin. Thus, choosing a good binsize boils down to finding a good compromise between precision and execution time. We saw that for the set of parameters chosen all along this paper (i.e $\mu = 1, \alpha = 0.5, \beta = 1$) a good binsize could be $\Delta = 0.5$. Intuitively, this bin choice strongly depends on the set of parameters and should come close to zero in the critical case ($\frac{\alpha}{\beta} \to 1$).

We also tried to plan for the future and thought about several aspects to improve our study and exhibit more elements to validate our findings on the Whittle estimation. Firstly, we could improve the **randomization method** : instead of using a uniform distribution to relocate the timestamps in the bin, we could optimize by relocating timestamps according to a Hawkes process whose parameters would be estimated progressively as we advance in the list of timestamps : the more we advance in time, the more we converge to the true parameters, and the better aligned with our initial Hawkes process our bin-randomization becomes, thus maximizing the information contained in the process. Another area of improvement is our **C++ implementation** of the Whittle method. We decided to implement this code for the following reason. At first, we thought that our Python code could be significantly improved by pre-compiling it with Cython. However, we had several import issues with Cython and we couldn't have it work. So we decided to bring all our code to C++. This was a hard task since C++ was totally unknown to us. The implementation of the code took us three long weeks of hard work but eventually we managed to come up with a functional code. The execution time results were not as promising as expected. This shows that faster optimization rely on C++ / pre-compiling methods. It is undeniably something we want to continue working on next year. Indeed, we believe that if R uses *Rcpp* to boost its time execution, then there is no reason for a 100% C++ code to perform worse.

Finally, we really look into applying these estimation methods to real financial data to determine, among other things, on which time scale they are truly relevant.

Bibliographie

[1] Yannick Bessy-Rol and Alice Launay. Modélisation du risque terroriste par les processus de hawkes. 2017.

[2] Felix Cheysson. "hawkesbow". https ://github.com/fcheysson/hawkesbow.

[3] Stephane Gaiffas Maryan Morel Søren Vinther Poulsen. Emmanuel Bacry, Martin Bompaire. "tick.hawkes". https ://x-datainitiative.github.io/tick/modules/hawkes.html.

[4] Gabriel Lang Felix Cheysson. Strong-mixing rates for hawkes processes and application to whittle estimation from count data. 2021.

[5] Riadh Zaatour José Da Fonseca. Hawkes process : Fast calibration, application to trade clustering and diffusive limit. 2013.

[6] Mehdi Lallouache and Damien Challet. The limits of statistical significance of hawkes processes fitted to financial data. 2015.

[7] Patrick Laub. Hawkes processes : Simulation, estimation, and validation. 2014.

[8] Ioane Muni Toke and Fabrizio Pomponio. Modelling trades-through in a limit order book using hawkes processes. 2012.

[9] D. Sornette V. Filimonov. Quantifying reflexivity in financial markets : Towards a prediction of flash crashes. 2012.

[10] Samuel N. Cohen Álvaro Cartea and Saad Labyad. Gradient-based estimation of linear hawkes processes with general kernels. 2021.