NNT : 2017SACLX068

Thèse de doctorat de l'Université Paris-Saclay préparée à l'École polytechnique

École doctorale n°574
École doctorale de mathématiques Hadamard
Spécialité de doctorat : Mathématiques appliquées

par

M. Massil Achab

Apprentissage statistique pour séquences d'évènements à l'aide de processus ponctuels

Thèse présentée et soutenue à Palaiseau, le 9 octobre 2017.

Composition du Jury :

M.	Manuel Gomez Rodriguez	Professeur MPI for Software Systems	Rapporteur
M.	Niels Richard Hansen	Professeur University of Copenhagen	Rapporteur
M.	Nicolas Vayatis	Professeur ENS Paris-Saclay	Examinateur
M.	Vincent Rivoirard	Professeur Université Paris-Dauphine	Examinateur
M.	Emmanuel Bacry	Directeur de recherche Uni. Paris-Dauph., École polytechnique	Directeur
M.	Stéphane Gaïffas	Professeur Uni. Paris-Diderot, École polytechnique	Directeur

À mes parents Ramdane et Samia, et à mon frère Mastane

Remerciements

Je tiens en premier lieu à exprimer ma plus profonde gratitude envers mes directeurs de thèse Stéphane Gaïffas et Emmanuel Bacry. Le premier pour son enthousiasme et son large spectre de connaissances en statistique, le second pour son flair et son recul dans l'étude des signaux temporels. Leur soutien et leur confiance durant ces trois ans m'ont permis de mener à bien ce travail. Merci pour tout ce que vous m'avez fait découvrir et pour tout ce que j'ai appris sous votre direction !

Je remercie Manuel Gomez Rodriguez et Niels Richard Hansen pour l'intérêt qu'ils ont porté à mon travail en acceptant de rapporter ma thèse, et m'excuse à nouveau de leur avoir volé une partie de leur été. Je suis également très honoré de la présence de Nicolas Vayatis et de Vincent Rivoirard dans mon jury de thèse.

Je suis très reconnaissant envers mes co-auteurs Agathe Guilloux, Iacopo Mastromatteo, Jean-François Muzy et Marcello Rambaldi pour leur temps et leur disponibilité, ainsi que pour toutes les discussions enrichissantes que j'ai pu avoir avec eux au cours de cette thèse. Jean-François, je n'oublierai pas ta capacité à mener les calculs les plus fous malgré l'avènement de logiciels de calcul formel performants. Marcello, j'ai beaucoup apprécié les différentes discussions, scientifiques ou non, qu'on a eues et j'espère que tu ne me tiendras pas rigueur de mon manque d'assiduité en escalade.

Je remercie également l'équipe administrative du CMAP pour leur sympathie et leur travail efficace : Nasséra, évidemment, Alexandra, Manoëlla et la doublette Vincent-Wilfried, avec qui j'ai écrit l'une des plus belles pages du football de ce département.

Mes pensées vont encore aux doctorants du CMAP avec qui j'ai pu échanger sur des sujets scientifiques pointus, je pense à Hadrien, Romain, Aline, Aymeric, Hélène, Antoine et Gustaw, sans oublier ceux qui ont préféré se concentrer sur le sport en espérant réitérer les exploits de leurs aînés, je pense à Othmane, Belhal, Geneviève et Cédric. Mention spéciale aux amoureux de la data, le nouvel or noir, que sont Alain, Maryan, Martin, Daniel, Youcef, Prosper, Sathiya, Fanny et Firas, ainsi qu'aux doctorants du CMLA que j'aurai vus, pour la plupart, plus souvent à l'étranger qu'à l'ENS Cachan.

Je salue enfin tous mes amis des années difficiles, d'où qu'ils viennent et où qu'ils aillent, et les remercie de ne jamais m'avoir pris au sérieux. J'espère qu'ils resteront les mêmes et que le nouveau grade que je vais atteindre ne changera rien à nos relations.

Mes derniers remerciements vont à mes parents, mon frère, et à l'ensemble des membres de ma famille pour leur confiance, leur soutien et leur organisation coordonnée du pot de thèse. Leur présence et leurs encouragements sont pour moi les piliers fondateurs de ce que je suis et de ce que je fais.

Résumé

Le but de cette thèse est de montrer comment certaines méthodes d'optimisation récentes permettent de résoudre des problèmes d'estimation difficiles posés par l'étude d'évènements aléatoires dans le temps. Alors que le cadre classique de l'apprentissage supervisé traite les observations comme une collection de couples indépendants de covariables et de labels, les modèles d'évènements s'intéressent aux temps d'arrivée, à valeurs continues, de ces évènements et cherchent à extraire de l'information sur la source de donnée. Ces évènements datés sont liés par la chronologie, et ne peuvent dès lors être considérés comme indépendants. Ce simple constat justifie l'usage d'un outil mathématique particulier, appelé processus ponctuel, pour apprendre une structure à partir de ces évènements.

Deux exemples de processus ponctuels sont étudiés dans cette thèse. Le premier est le processus ponctuel sous-jacent au modèle de Cox à risques proportionnels : son intensité conditionnelle permet de définir le ratio de risque, une quantité fondamentale dans la littérature de l'analyse de survie. Le modèle de régression de Cox relie la durée avant l'apparition d'un évènement, appelé défaillance, aux covariables d'un individu. Ce modèle peut être reformulé à l'aide du cadre des processus ponctuels. Le second est le processus de Hawkes qui modélise l'impact des évènements passés sur la probabilité d'apparition d'évènements futurs. Le cas multivarié permet d'encoder une notion de causalité entre les différentes dimensions considérées.

Cette thèse est divisée en trois parties. La première s'intéresse à un nouvel algorithme d'optimisation que nous avons développé. Il permet d'estimer le vecteur de paramètre de la régression de Cox lorsque le nombre d'observations est très important. Notre algorithme est basé sur l'algorithme SVRG et utilise une méthode MCMC pour approcher la direction de descente. Nous avons prouvé des vitesses de convergence pour notre algorithme et avons montré sa performance numérique sur des jeux de données simulées et issues du monde réel. La deuxième partie montre que la causalité au sens de Hawkes peut être estimée de manière non-paramétrique grâce aux cumulants intégrés du processus ponctuel multivarié. Nous avons développé deux méthodes d'estimation des intégrales des noyaux du processus de Hawkes, sans faire d'hypothèse sur la forme de ces noyaux. Nos méthodes sont plus rapides et plus robustes, vis-à-vis de la forme des noyaux, par rapport à l'état de l'art. Nous avons démontré la consistance statistique de la première méthode, et avons montré que la deuxième peut être réduite à un problème d'optimisation convexe. La dernière partie met en lumière les dynamiques de carnet d'ordre grâce à la première méthode d'estimation non-paramétrique introduite dans la partie précédente. Nous avons utilisé des données du marché à terme EUREX, défini de nouveaux modèles de carnet d'ordre (basés sur les précédents travaux de Bacry et al.) et appliqué la méthode d'estimation sur ces processus ponctuels. Les résultats obtenus sont très satisfaisants et cohérents avec une analyse économétrique. Un tel travail prouve que la méthode que nous avons développée permet d'extraire une structure à partir de données aussi complexes que celles issues de la finance haute-fréquence.

Abstract

The aim of this thesis is to show how recent optimization methods help solving tough estimation problems based on the event models. While the classical framework of supervised learning treats the observations as a collection of covariate and label independent pairs, event models only focus on the arrival dates of these events and then seek to extract information about the data source. These timestamped events are ordered chronologically and can not therefore be considered independent. This simple fact justifies the use of a particular mathematical tool called point process to learn some structure from these events.

Two examples of point processes are studied in this thesis. The first is the underlying point process in the Cox model with proportional hazards: its conditional intensity allows to define the risk ratio, a fundamental quantity in the literature of the survival analysis. The Cox regression model links the duration before the occurrence of an event, called failure, to an individual's covariates. This model can be reformulated using the framework of point processes. The second is the Hawkes process, which models the impact of past events on the probability of future events. The multivariate case makes it possible to encode a notion of causality between the different dimensions considered.

This thesis is divided into three parts. The first focuses on a new optimization algorithm we have developed. It allows to estimate the parameter vector of the Cox regression when the number of observations is very important. Our algorithm is based on the Stochastic Variance Reduced Gradient (SVRG) algorithm and uses a Monte Carlo Markov Chain (MCMC) method to approximate the descent direction. We have proved convergence rates for our algorithm and have shown its numerical performance on simulated and real world data sets. The second part shows that the Hawkes causality can be estimated in a non-parametric way by the integrated cumulants of the multivariate point process. We have developed two methods for estimating the integrals of the kernels of the Hawkes process, without making any hypothesis about the shape of these kernels. Our methods are faster and more robust, with respect to the shape of the kernel, compared to the state-of-the-art. We have demonstrated the statistical consistency of the first method, and have shown that the second method can be reduced to a convex optimization problem. The last part highlights the dynamics of the order book thanks to the first non-parametric estimation method introduced in the previous section. We used EUREX futures data, defined new order book models (based on previous work by Bacry et al.) and applied the estimation method on these point processes. The results obtained are very satisfactory and consistent with an econometric analysis. This work proves that the method that we have developed makes it possible to extract a structure from data as complex as those resulting from high-frequency finance.

List of papers being part of this thesis

- M. Achab, S. Gaïffas, A. Guilloux and E. Bacry, *SGD with Variance Reduction beyond Empirical Risk Minimization*, International Conference on Monte Carlo Methods and Applications, 2017.
- M. Achab, E. Bacry, S. Gaïffas, I. Mastromatteo and J.-F. Muzy, *Uncovering Causality from Multivariate Hawkes Integrated Cumulants*, International Conference of Machine Learning, 2017.
- M. Achab, E. Bacry, J.-F. Muzy and M. Rambaldi, *Analysis of order book flows using a nonparametric estimation of the branching ratio matrix*, to appear in Quantitative Finance, 2017.

Contents

Introduction

The guiding principle of this thesis is to show how the arsenal of recent optimization methods can help solving challenging new estimation problems on events models. While the classical framework of supervised learning [HTF09] treat the observations as a collection of independent couples of features and labels, events models focus on arrival timestamps to extract information from the source of data. These timestamped events are chronologically ordered and can't be regarded as independent. This mere statement motivates the use of a particular mathematical object called *point process* [DVJ07] to learn some patterns from events. Let us begin by presenting and motivating the questions on which we want to shed some light in this thesis.

Motivations

The amount of data being digitally collected and stored is vast and expanding rapidly. The use of predictive analytics that extract value of this data, often referred as the *data revolution*, has been successfully applied in astronomy [FB12], retail sales [MB+12] and search engines [CCS12], among others. Healthcare institutions are now also relying on data to build customized and personalized treatment models using tools from survival analysis [MD13]. Medical research often aims at uncovering relationships between the patient's covariates and the duration until a *failure* event (death or other adverse effects) happen. The information that some patients did not die during the study is obviously relevant, but can't be casted in a regression problem where one would need to observe the lifetime for all patients. This has been circumvented in [Dav72], one of the most cited scientific paper of all time [VNMN14], with its *proportional hazards model* that is regarded as a regression that can also extract information from *censored* data, *i.e.* patients whose failure time is not observed. An estimation procedure of the parameter vector of the *regression* without any assumption on the baseline hazard, regarded sometimes as a nuisance parameter, was introduced in [Cox75] and is done via the maximization of the *partial likelihood* of the model. Such procedure can efficiently handle high-dimensional covariates, which happens with biostatistics data, by adding penalization terms to the criterion to minimize [Goe10, Tib96]. However, algorithms to maximize Cox partial likelihood does not scale well when the number of patients is high, on the contrary to most algorithms that enabled the *data revolution*. We might thus ask ourselves the following question:

Question 1. *How to adapt Cox proportional hazards model regression parameter estimation algorithm to the large-scale setting ?*

Few years before the twentieth century, the French sociologist Durkheim already argued that human societies were like biological systems in that they were made up of interrelated components [Dur97]. Now that our technology enabled us to be remotely connected, plenty of fields involve networks, like social networks, information systems, marketing, epidemiology, national security, and others. A better understanding of those large real-world networks and processes that take place over them would have paramount applications in the mentioned domains [Rod13]. The observation of networks often reduces to noting when nodes of the network send a message, buy a product or get infected by a virus. We often observe *where and when* but not *how and why* messages are sent over a social network. Event data from multiple providers can however help uncovering the joint dynamics and revealing the underlying structure of a system. One way to recover the influence structure between different sources is to use a kind of point process named *Hawkes process* [Haw71b, Haw71a], whose arrival rate of events depend on the past events. Hawkes processes have been succesfully applied to model the mutual influence between earthquakes with different times and magnitudes [Oga88]. Namely, it encodes how an earthquake increases the occurence's probability of new earthquakes in the form of aftershocks, via the use of *Hawkes kernels*. Hawkes processes also enable measuring what we call *Hawkes causality* i.e. the average number of events of type i that are trigerred by events of type j. Hawkes process have been succesfully applied in a broad range of domains, the two main applications model interactions within social networks [BBH12, ZZS13, ISG13] and financial transactions [BMM15]. However, usual estimation of Hawkes causality is done by making strong assumptions on the shape of the Hawkes kernels to simplify the inference algorithm [ZZS13]. A common assumption is the monotonic decreasing shape of the kernels (exponential or power-law), meaning that an event impact is always instantly maximal, which is non-realistic since in practice there may exist a delay before the maximal impact. This leads to the following question:

Question 2. *Can we retrieve Hawkes causality without parametrizing the kernel functions ?*

To answer positively to the second question, we developed two new nonparametric estimation methods for Hawkes causality, faster and which scales better with a large number of nodes. In this part, we only focus on the first one, for which we have proved a consistency result. Since Bowsher's pioneering work [Bow07], who recognized the flexibility and the simplicity of using Hawkes processes in order to model the joint dynamics of trades and mid-price changes of the NYSE, Hawkes processes have steadily gained in popularity in the domain of high frequency finance, see [BMM15] for a review. Indeed, taking into account the irregular occurences of transaction data requires to consider it as a point process. Besides, in the financial area, plenty of features that summarize empirical findings are already known. For instance, the flow of trades is known to be autocorrelated and cross-correlated with price moves. Such features called *stylized facts*, from the economist Nicholas Kaldor [Kal57] who referred to statistical trends that need to be taken into account despite a possible lack of microscopic understanding. These stylized facts can advantageously be captured using the

notion of Hawkes causality. Understanding the order book dynamics is one of the core question in financial statistics, and previous nonparametric representations of order books with multivariate Hawkes processes were low-dimensional because of their estimation method's complexity. The nonparametric estimation of Hawkes causality introduced in the second part of this thesis is fast and robust to kernel functions' shape, and it is natural to wonder what kind of *stylized facts* it can uncover from order book timestamped data.

Question 3. *Can we draw a more precise picture of order book flows dynamics using Hawkes causality's nonparamatric estimation introduced in the second part ?*

Outline

Each question presented above corresponds to a part of the thesis.

In Part I, we answer Question 1 by introducing a new stochastic gradient descent algorithm applied to the maximization of regularized Cox partial-likelihood, see details below. Indeed, the regularized Cox partial-likelihood writes as a sum of subfunctions which depend on varying length sequences of observation, on the contrary to the usual empirical risk minimization framework where subfunctions depend on one observation. Classical stochastic gradient descent algorithms are less effective in our case. We adapt the algorithm SVRG [JZ13] [XZ14] by adding another sampling step: each subfunction's gradient is estimated using a Monte Carlo Markov Chain (MCMC). Our algorithm achieves linear convergence once the number of MCMC iterations is bigger than an explicit lower bound. We illustrate the outperformance of our algorithm on survival datasets.

Answers to Question 2 lie in Part II where we study two nonparametric estimation procedures for Hawkes causality. Both methods are based on the computation of the integrated cumulants of the Hawkes process and taking advantage of relations between the integrated cumulants and the Hawkes causality matrix. The first approach relies on matching the second and third order empirical integrated cumulants with their theoretical counterparts. This is done via the minimization of the squared norm of the difference between the two terms, which can be viewed as a Generalized Method of Moments [Hal05]. However, the optimization problem to solve is non-convex providing thus an approximate solution to the exact initial problem. This second approach is based on the completion of the Hawkes causality matrix using the first and second integrated cumulants. The relaxation of the exact problem writes as a convex optimization problem which enables us to provide the exact solution of this approximate problem.

Finally, in Part III, we apply the first method developed in Part II to high-frequency order book data from the EUREX exchange. We apply the procedure to fit a 12-dimensional Hawkes order book model for a single asset and estimate the influence of the different events on each other. Such order book model is a natural extension of the 8-dimensional model studied in [BJM16]. We then scale the dimension so as to account for events of two assets simultaneously and discuss the joint dynamics and the cross-asset effects. Usual nonparametric methods [BM14b]

[RBRGTM14] focus on the estimation of the kernel functions, and prevent order book model's dimension from being too large and/or the dataset from being too heavy. Our nonparametric method only estimates kernels' integral, involves a lighter computation and then scales better with a large number of nodes or large number of events. We also show that the Hawkes causality matrix provides a very rich summary of the system interactions. It can thus be a valuable tool in understanding the underlying structure of a system with many type of events.

Let us now rapidly review the main results of this thesis.

1 Part I: Large-scale Cox model

Many supervised machine learning problems can be cast into the minimization of an expected loss over a data distribution. Following the empirical risk minimization principle, the expected loss is approximated by an average of losses over training data, and a major success has been achieved by exploiting the sum-structure to design efficient stochastic algorithms [Bot10]. Such stochastic algorithms enable a very efficient extraction of value from massive data. Applying this to large-scale survival data, from biostatistics or economics, is of course of great importance.

In Chapter I, we review the recent advances in convex optimization with Stochastic Gradient Descent (SGD) algorithms, from the pioneering work of [RM51] to the recent variants with variance reduction [DBLJ14] [XZ14] [SSZ13] [RSB12]. We then introduce the notion of Point Process [DVJ07] which provides key tools for modeling events *i.e.* timestamps and/or locations data. We finally introduce the Cox proportional hazards model [Dav72] that relates the time that passes before some event occurs to one or more covariates via the notion of *hazard rate*. In Chapter II, we introduce our new optimization algorithm to help fitting large-scale Cox model.

1.1 Background on SGD algorithms, Point Processes and Cox proportional hazards model

In this chapter, we review the classic results behind Stochastic Gradient Descent algorithms and its variance reduced adaptations. We then introduce Cox proportional hazards model.

1.1.1 Stochastic Gradient Descent algorithms

SGD algorithms from a general distribution A variety of statistical and machine learning optimization problems writes

$$\min_{\theta \in \mathbb{R}^d} F(\theta) = f(\theta) + h(\theta) \qquad \text{with} \qquad f(\theta) = \mathbb{E}^{\xi}[\ell(\theta, \xi)],$$

where f is a goodness of fit measure depending implicitly on some observed data, h is a regularization term that imposes structure to the solution and ξ is a random variable. Typically, f is a differentiable function with a Lipschitz gradient, whereas h might be non-smooth -

typical examples include sparsity inducing penalty - such as the ℓ_1 penalization.

First-order optimization algorithms are all variants of *Gradient Descent* (GD), which can be traced back to Cauchy [Cau47]. Starting at some initial point θ^0, this algorithm minimizes a differentiable function f by iterating the following equation

$$\theta^{t+1} = \theta^t - \eta_t \nabla f(\theta^t). \tag{1}$$

where $\nabla f(\theta)$ stands for the gradient of f evaluated at θ and (η_t) is a sequence of step sizes. *Stochastic Gradient Descent* (SGD) algorithms focus on the case where ∇f is intractable or at least time-consuming to compute. Noticing that $\nabla f(\theta)$ writes as an expectation, one idea is to approximate the gradient in the update step (1) with a Monte Carlo Markov Chain [AFM17]. For instance, replacing the exact gradient $\nabla f(\theta)$ with its MCMC estimate has enabled a significant step forward in training Undirected Graphical Models [Hin02] and Restricted Boltzmann Machines [HS06]. This first form of Stochastic Gradient Descent is called *Contrastive Divergence* in the mentionned contexts.

SGD Algorithms from the uniform distribution Most machine learning optimization problems involve a data fitting loss function f averaged over sample points because of the empirical risk minimization principle [Vap13]. Namely, the objective function writes

$$\min_{\theta \in \mathbb{R}^d} F(\theta) = f(\theta) + h(\theta) \qquad \text{with} \qquad f(\theta) = \frac{1}{n} \sum_{i=1}^{n} f_i(\theta),$$

where n is the number of observations, and f_i is the loss associated to the i^{th} observation. In that case, instead of running MCMC to approximate ∇f, one uniformly samples a random integer i between 1 and n and replace $\nabla f(\theta)$ with $\nabla f_i(\theta)$ in the update step (1). In the large-scale setting, computing $\nabla f(\theta)$ at each update step represents the bottleneck of the minimization algorithm, and SGD helps decreasing the computation time. Assuming that the computation of each $\nabla f_i(\theta)$ costs 1, the computation of the full gradient $\nabla f(\theta)$ costs n, meaning that SGD's update step is n times faster than GD's one.

The comparison of the convergence rates is however different. Consider f twice differentiable on $\mathbb{R}^d$, μ-strongly-convex, meaning that eigenvalues of the Hessian matrix $\nabla^2 f(\theta)$ are greater than $\mu > 0$ for any $\theta \in \mathbb{R}^d$, and L-smooth, meaning that the same eigenvalues are smaller than $L > 0$. Convergence rates with other assumptions on the function f can be found in [B+15]. We denote θ^* its minimizer and define the *condition number* as $\kappa = L/\mu$. The convergence rate is defined for iterative methods as a tight upper bound of a pre-defined error, and is regarded as the speed at which the algorithm converges. Denoting θ^t the iterate after t steps of an iterative algorithm and considering the difference $\mathbb{E} f(\theta^t) - f(\theta^*)$ as error, Gradient Descent's convergence rate is $O(e^{-t/\kappa})$, while Stochastic Gradient Descent's one is $O(\kappa/t)$. A convergence rate of the form $O(e^{-\alpha t})$ with $\alpha > 0$ is called *linear convergence rate* since the error decrease after one iteration is at worst linear. Equivalently, convergence rates can be phrased as the total complexity to reach a fixed accuracy *i.e.* the number of iterations after

which the difference $\mathbb{E}f(\theta^t) - f(\theta^*)$ becomes smaller than $\epsilon > 0$ multiplied by the complexity per iteration. The algorithm Gradient Descent will reach the accuracy ϵ after $O\left(\kappa \log \frac{1}{\epsilon}\right)$ iterations resulting in a $O\left(nd\kappa \log \frac{1}{\epsilon}\right)$ complexity, while Stochastic Gradient Descent reachs such accuracy after $O\left(\frac{\kappa}{\epsilon}\right)$ iterations and then a $O\left(\frac{d\kappa}{\epsilon}\right)$ complexity.

Recently, different works improved Stochastic Gradient Descent using variance reduction techniques from Monte Carlo methods. The idea is to add a *control variate* term to the descent direction to improve the bias-variance tradeoff in the approximation of the real gradient $\nabla f(\theta)$. Those variants also enjoy linear convergence rates, and then smaller complexities (to reach accuracy ϵ) than Gradient Descent since the complexity per iteration of those algorithms is $O(d)$ versus $O(nd)$ for Gradient Descent. Those typically typically enjoy complexity of the form $O\left((n+\kappa)d \log \frac{1}{\epsilon}\right)$ in the strongly-convex case, see [SLRB17, JZ13, DBLJ14, SSZ13].

1.1.2 Point processes

Point process is a useful mathematical tool to describe phenomena occuring at random locations and/or times. A point process is a random element whose values are point patterns on a set S. We present here the useful results when the set S is the interval $[0, T)$, and points are timestamps of events; this special case is sometimes called *temporal point process.* The book [DVJ07] is regarded as the main reference on point processes' theory.

Every *realization* of a point process ξ can be written as $\xi = \sum_{i=1}^{n} \delta_{t_i}$ where δ is the Dirac measure, n is an integer-valued random variable and t_i's are random elements of $[0, T)$. It can be equivalently represented by a *counting process* $N_t = \int_0^t \xi(s)ds = \sum_{i=1}^{n} \mathbb{1}_{\{t_i \leq t\}}$. The usual characterization of temporal point process is done via the *conditional intensity* function, which is defined as the infinitesimal rate at which events are expected to occur after t, given the history of N_s prior to t:

$$\lambda(t|\mathscr{F}_t) = \lim_{h \to 0} \frac{\mathbb{P}(N_{t+h} - N_t = 1|\mathscr{F}_t)}{h},$$

where $\mathscr{F}_t$ is the filtration of the process that encodes information available up to (but not including) the time t. The most simple temporal point process is the *Poisson process* which assumes that the events arrive at a constant rate, which corresponds to a constant instensity function $\lambda_t = \lambda > 0$. Note that temporal point processes can also be characterized by the distribution of interevent times *i.e.* the duration between two consecutive events. We remind that the distribution of interevent times of a Poisson process with intensity λ is an exponential distribution of parameter λ. See the Page 41 of [DVJ07] for four equivalent ways of defining a temporal point process.

Two examples of temporal point process are treated in this thesis. The first is the point process behind Cox proportional hazards model: its conditional intensity function allows to define the *hazard ratio*, a fundamental quantity in survival analysis literature, see [ABGK12]. The Cox regression model relates the duration before an event called *failure* to some covariates. This model can be reformulated in the framework of point processes [ABGK12]. The second

is the Hawkes process which models how past events increase the probability of future events. Its multivariate version enables encoding a notion of causality between the different nodes. We introduce below the Cox proportional hazards model, and the Hawkes processes in Part II.

1.1.3 Cox proportional hazards model

Survival analysis focuses on time-to-event data, such as the death in biological organisms and failure in mechanical systems, and is now widespread in a variety of domains like biometrics, econometrics and insurance. The variable we study is the waiting time until a well-defined event occurs, and the main goal of survival analysis is to link the covariates, or features, of a patient to its survival time T.

Following the theory of point processes, we define the *intensity* as the conditioned probability that a patient dies immediately after t, given that he was alive before t:

$$\lambda(t) = \lim_{h \to 0} \frac{\mathbb{P}(t \leq T \leq t + h | t \leq T)}{h}.$$

The most popular approach, for some reasons explained below, is Cox proportional hazards model [Dav72]. The Cox model assumes a semi-parametric form for the hazard ratio at time t for the patient i, whose features are encoded in the vector $x_i \in \mathbb{R}^d$:

$$\lambda_i(t) = \lambda_0(t) \exp(x_i^\top \theta),$$

where $\lambda_0(t)$ is a baseline hazard ratio, which can be regarded as the hazard ratio of a patient whose covariates are $x = 0$. One estimation approach considers λ_0 as a *nuisance* and only estimates θ via maximizing a *partial likelihood* [Dav72]. This way of estimating suits clinical studies where physicians are only interested in the effects of the covariates encoded in x on the hazard ratio. This can be done with computing the ratio of hazard ratios from two different patients:

$$\frac{\lambda_i(t)}{\lambda_j(t)} = \exp((x_i - x_j)^\top \theta)$$

For that reason, Cox model is said to be a proportional hazards model.

However, maximizing this partial likelihood is a hard problem when we deal with large-scale (meaning a large number of observations n) and high-dimensional (meaning large d) data. To tackle the high-dimensionality, sparse penalized approaches have been considered in the literature [Tib96] [T$^+$97] [Goe10]. The problem is now to minimize the negative of the partial log-likelihood $f(\theta) = -\ell(\theta)$ with a penalization $h(\theta)$ that makes the predictor θ to become sparse and then select variables. We will discuss this approach and the different models in Chapter II. On the contrary, approaches to tackle the large-scale side of the problem do not yet exist.

1.2 SVRG beyond Empirical Risk Minimization

Survival data $(y_i, x_i, \delta_i)_{i=1}^{n_{\text{pat}}}$ contains, for each individual $i = 1, \dots, n_{\text{pat}}$, a features vector $x_i \in \mathbb{R}^d$, an observed time $y_i \in \mathbb{R}_+$, which is a failure time if $\delta_i = 1$ or a right-censoring time if $\delta_i = 0$. If $D = \{i : \delta_i = 1\}$ is the set of patients for which a failure time is observed, if $n = |D|$ is the total number of failure times, and if $R_i = \{j : y_j \geq y_i\}$ is the index of individuals still at risk at time y_i, the negative Cox partial log-likelihood writes

$$-\ell(\theta) = \frac{1}{n} \sum_{i \in D} \Big[-x_i^\top \theta + \log\Big(\sum_{j \in R_i} \exp(x_j^\top \theta) \Big) \Big] \tag{2}$$

for parameters $\theta \in \mathbb{R}^d$. Each gradient of the negative log-likelihood then writes as two nested expectations: one from an uniform distribution over D, the other over a Gibbs distribution, see Chapter II for details.

Our minimization algorithm is *doubly stochastic* in the sense that gradient steps are done using stochastic gradient descent (SGD) with variance reduction, and the inner expectations are approximated by a Monte Carlo Markov Chain (MCMC) algorithm. We derive conditions on the MCMC number of iterations guaranteeing convergence, and obtain a linear rate of convergence under strong convexity and a sublinear rate without this assumption.

2 Part II: Uncover Hawkes causality without parametrization

In Chapters III and IV, we study two methods to uncover causal relationships from a multivariate point process. We focus on one approach per chapter.

2.1 Hawkes processes

In order to model the joint dynamics of several point processes (for example timestamps of messages sent by different users of a social network), we will consider the multidimensional Hawkes model, introduced in 1971 in [Haw71a] and [Haw71b], with cross-influences between the different processes. By definition a family of d point processes is a multidimensional Hawkes process if the intensities of all of its components write as linear regressions over the past of the d processes:

$$\lambda_t^i = \mu^i + \sum_{k=1}^{D} \int_0^t \phi^{ij}(t-s) dN_s^j.$$

Another way to construct Hawkes processes is to consider the following population representation, see [HO74]: individuals of type i, $1 \leq i \leq d$, arrive as a Poisson process of intensity μ^i. Every individual can have children of all types and the law of the children of type i of an individual of type j who was born or migrated in t is an inhomogeneous Poisson process of intensity $\phi^{ij}(\cdot - t)$.

This construction is nice because it yields a natural way to define and measure the causality between events in the Hawkes model, where the *integrals*

$$g^{ij} = \int_0^{+\infty} \phi^{ij}(u)\, du \geq 0 \text{ for } 1 \leq i, j \leq d.$$

weight the directed relationships between individuals. Namely, introducing the counting function $N_t^{i \leftarrow j}$ that counts the number of events of i whose direct ancestor is an event of j, we know from [BMM15] that

$$\mathbb{E}[dN_t^{i \leftarrow j}] = g^{ij} \mathbb{E}[dN_t^j] = g^{ij} \Lambda^j dt, \tag{3}$$

where we introduced Λ^i as the intensity expectation, satisfying $\mathbb{E}[dN_t^i] = \Lambda^i dt$. However in practice, the Hawkes kernels are not directly measurable from the data and these measures of causality between the different kinds of events are thus inaccessible.

In the literature, there are main two classes of estimation procedures for Hawkes kernels: the parametric one and the nonparametric one. The first one assumes a parametrization of the Hawkes kernels, the most usual assumes the kernels are decaying exponential, and estimate the parameter via the maximization of the Hawkes log-likelihood, see for example [BGM15] or [ZZS13]. The second one is based either on the numerical resolution of Wiener-Hopf equations which links the Hawkes kernels to its correlation structure [BM14b] (or equivalently on the approximation of the Hawkes process as an Autoregressive model and the resolution of Yule-Walker equations [EDD17]), or on a method of moments via the minimization of the contrast function defined in [RBRGTM14].

In Chapters III and IV, we propose two new nonparametric estimation methods to infer the integrals of the kernels using only the integrated moments of the multivariate Hawkes process.

For all estimation procedures mentionned above, including ours, we need the following stability condition so that the process admits a version with a stationary intensity:

Assumption 1. *The spectral norm of* $\boldsymbol{G} = [g^{ij}]$ *satisfies* $||\boldsymbol{G}|| < 1$.

2.2 Generalized Method of Moments approach

A recent work [JHR15] proved that the integrated cumulants of Hawkes processes can be expressed as functions of $\boldsymbol{G} = [g^{ij}]$, and provided the constructive method to obtain these expressions. The first approach we developed in this part is a moment matching method that fits the second-order and the third-order integrated cumulants of the process. To that end, we have designed consistent estimators of the integrated first, second and third cumulants of the Hawkes process. Their theoretical counterparts are polynomials of $\boldsymbol{R} = (\boldsymbol{I} - \boldsymbol{G})^{-1}$, as shown in

[JHR15]:

$$\Lambda^i = \sum_{m=1}^{d} R^{im}\mu^m$$
$$C^{ij} = \sum_{m=1}^{d} \Lambda^m R^{im} R^{jm}$$
$$K^{ijk} = \sum_{m=1}^{d} (R^{im}R^{jm}C^{km} + R^{im}C^{jm}R^{km} + C^{im}R^{jm}R^{km} - 2\Lambda^m R^{im}R^{jm}R^{km}).$$

Once we observe the process $\boldsymbol{N}_t$ for $t \in [0, T]$, we compute the empirical integrated cumulants on windows $[-H_T, H_T]$, and minimize the squared difference $\mathscr{L}_T$ between the theoretical cumulants and the empirical ones. We have proven the consistency of our estimator in the limit $T \to \infty$, once the sequence (H_T) satisfies some conditions. Our problem can be seen as a Generalized Method of Moments [Hal05].

To prove the consistency of the empirical integrated cumulants, we need the following assumption:

Assumption 2. *The sequence of integration domain's half-length satisfies $H_T \to \infty$ and $H_T^2 / T \to 0$.*

We prove in Chaper III the following theorem of consistency.

Result 1. *Under Assumptions 1 and 2, the sequence of estimators defined by the minimization of $\mathscr{L}_T(\boldsymbol{R})$ converges in probability to the true value $\boldsymbol{G}$:*

$$\widehat{\boldsymbol{G}}_T = \boldsymbol{I} - \left(\underset{\boldsymbol{R} \in \Theta}{\operatorname{argmin}} \mathscr{L}_T(\boldsymbol{R}) \right)^{-1} \xrightarrow[T \to \infty]{\mathbb{P}} \boldsymbol{G}$$

The numerical part, on both simulated and real-world datasets, gives very satisfying results. We first simulated event data, using the thinning algorithm of [Oga81], with very different kernel shape - exponential, power law and rectangular - and recover the true value of $\boldsymbol{G}$ for each kind of kernel. Our method is, to the best of our knowledge, the most robust with respect to the shape of the kernels. We then ran our method on the 100 most cited websites of the MemeTracker database, and on financial order book data: we outperformed state-of-the-art methods on MemeTracker and extracted nice and interpretable features from the financial data. Let also mention that our method is significantly faster (roughly 50 times faster) since previous methods aim at estimating functions while we only focus on their integrals.

The simplicity of the method, that maps a list of list of timestamps to a causality map between the nodes, and its statistical consistency, incited us to design new point process models of order book and capture its dynamics. The features extracted using our method have very insightful economic interpretation. This is the main purpose of the Part III.

2.3 Constrained optimization approach

The previous approach based on the Generalized Method of Moments need the first three cumulants to obtain enough information from the data to recover the d^2 entries of $\boldsymbol{G}$. Assuming that the matrix $\boldsymbol{G}$ has a certain structure, we can get rid of the third order cumulant and design another estimation method using only the first two integrated cumulants. Plus, the resulting optimization problem is convex, on the contrary to the minimization of $\mathscr{L}_T$ above, which enables the convergence to the global minimum. The matrix we want to estimate minimize a simple criterion f convex, typically a norm, while being consistent with the first two empirical integrated cumulants.

We formulate our problem as the following constrained optimization problem:

$$\begin{aligned} \min_{\boldsymbol{G}} \quad & f(\boldsymbol{G}) \\ \text{s.t.} \quad & \boldsymbol{C} = (\boldsymbol{I} - \boldsymbol{G})^{-1}\boldsymbol{L}(\boldsymbol{I} - \boldsymbol{G}^\top)^{-1} \\ & ||\boldsymbol{G}|| < 1 \\ & g^{ij} \geq 0 \end{aligned}$$

where $f(\boldsymbol{G})$ is a norm that provides a particular structure to the solution. Every matrix $\boldsymbol{G}$ satisfying $\boldsymbol{C} = (\boldsymbol{I} - \boldsymbol{G})^{-1}\boldsymbol{L}(\boldsymbol{I} - \boldsymbol{G}^\top)^{-1}$ equals $\boldsymbol{I} - \boldsymbol{L}^{1/2}\boldsymbol{M}\boldsymbol{C}^{-1/2}$ with $\boldsymbol{M}$ an orthogonal matrix. Instead of the previous problem, we now focus on its convex relaxation, we split the variables $\boldsymbol{G}$ and $\boldsymbol{M}$, and solve the problem with the *Alternating Direction Method of Multipliers* algorithm, see [GM75] and [GM76]:

$$\begin{aligned} \min_{\boldsymbol{G},\boldsymbol{M}} \quad & f(\boldsymbol{G}) + \mathbb{1}_{\overline{\mathscr{B}}}(\boldsymbol{M}) + \mathbb{1}_{\mathscr{B}}(\boldsymbol{G}) + \mathbb{1}_{\mathbb{R}_+^{d\times d}}(\boldsymbol{G}) \\ \text{s.t.} \quad & \boldsymbol{G} = \boldsymbol{I} - \boldsymbol{L}^{1/2}\ \boldsymbol{M}\ \boldsymbol{C}^{-1/2}, \end{aligned}$$

where $\mathscr{B}$ (resp. $\overline{\mathscr{B}}$) is the open (resp. closed) unit ball w.r.t. the spectral norm. The closed unit ball w.r.t. the spectral norm is indeed the convex hull of the orthogonal group.

On the contrary to the optimization problem of the previous chapter, the problem just stated is convex. We test this procedure on numerical simulations of various Hawkes kernels and real order book data, and we show how the criterion f impact the matrices we retrieve.

3 Part III: Capture order book dynamics with Hawkes processes

Chapter V focus on the estimation of Hawkes kernels' integrals on financial data, using the estimation method introduced in Chapter III. This in turn allowed us to have a very precise picture of the high frequency order book dynamics. We used order book events associated with 4 very liquid assets from the EUREX exchange, namely DAX, EURO STOXX, Bund and Bobl future contracts.

3.1 A single asset 12-dimensional Hawkes order book model

As a first application of the procedure described in Chapter III, we consider the following 12-dimensional point process, a natural extension of the 8-dimensional point process introduced in [BJM16]:

$$\boldsymbol{N}_t = (T_t^+, T_t^-, L_t^+, L_t^-, C_t^+, C_t^-, T_t^a, T_t^b, L_t^a, L_t^b, C_t^a, C_t^b)$$

where each dimension counts the number of events before t:

- T^+ (T^-): upwards (downward) mid-price move triggered by a market order.
- L^+ (L^-): upwards (downward) mid-price move triggered by a limit order.
- C^+ (C^-): upwards (downward) mid-price move triggered by a cancel order.
- T^a (T^b): market order at the ask (bid) that does not move the price.
- L^a (L^b): limit order at the ask (bid) that does not move the price.
- C^a (C^b): cancel order at the ask (bid) that does not move the price.

We then use the causal interpretation of Hawkes processes to interpret our solution as a measure of the causality between events. This application of the method to this new model revealed the different interactions that lead to the high-frequency price mean reversion, and those between liquidity takers and liquidity makers.

For instance, one observes the effects of T^+ events on other events on Figure .1 (in the first columnn on the left). The most relevant interactions are the $T^+ \to L^+$ and $T^+ \to L^-$: the latter is more intense and related to the mean-reversion of the price. Indeedn when a market order consumes the liquidity available at the best ask, two main scenarios can occur for the mid-price to change again, either the consumed liquidity is replaced, reverting back the price (mean-reverting scenario, highly probable) or the price moves up again and a new best bid is created.

3.2 A multi-asset 16-dimensional Hawkes order book model

The nonparametric estimation method introduced in Chapter III allows a fast estimation for a nonparametric methodology. We then scale up the model so as to account for events on two assets simultaneously and unveil a precise structure of the high-frequency cross-asset dynamics. We consider a 16-dimensional model, made of two 8-dimensional models of the form

$$\boldsymbol{N}_t = (P_t^+, P_t^-, T_t^a, T_t^b, L_t^a, L_t^b, C_t^a, C_t^b)$$

where the dimension P^+ (P^-) counts upwards (downward) mid-price move triggered by any order.

We compared two couples of assets that share exposure to the same risk factors. The main empirical result of this study concerned the couple (DAX, EURO STOXX) for which price

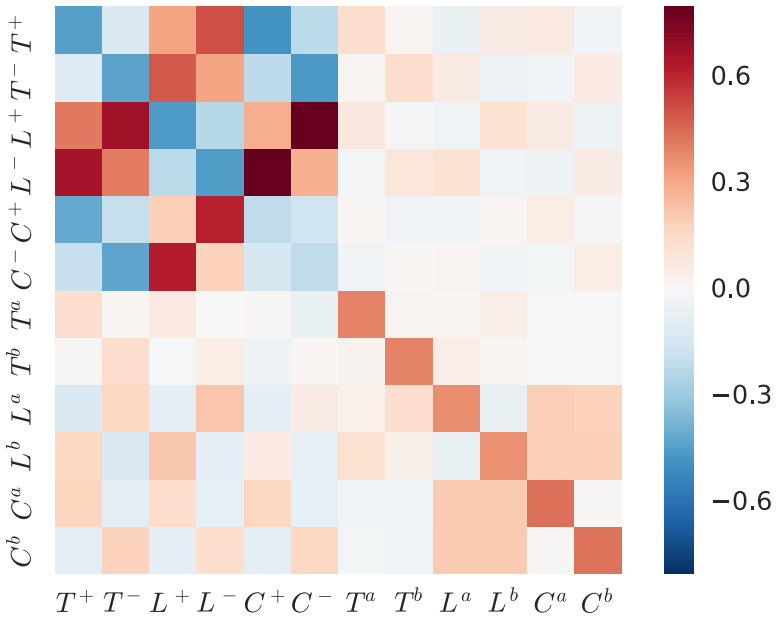


Figure .1: Kernel norm matrix $\boldsymbol{G}$ estimated for the DAX future with $H = 1s$.

changes and liquidity changes on the DAX (small tick) mainly influence liquidity on the EURO STOXX (large tick), while price changes and liquidity changes on the EURO STOXX tend to trigger price moves on the DAX. We ran the estimation procedure on the 16-dimensional model, we focus our discussion on the two non-diagonal 8×8 submatrices on Figure .2 that correspond to the interaction between the assets - the subscript D stands for DAX and X for EURO STOXX.

The most striking feature emerging from Figure .2 is the very intense relation between same-sign price movements on the two assets. Another notable aspect is the different effects of price moves and liquidity changes of one asset on events on the other asset. Price moves on the DAX have also an effect on the flow of limit orders on EURO STOXX ($P_D^+ \to L_X^b$ and $P_D^+ \to C_X^a$), whereas EURO STOXX price moves triggers mainly DAX price moves in the same direction ($P_X^+ \to P_D^+$). An important aspect for understanding this result is the different perceived tick sizes on the two assets. Note that the effects observed above can be explained with the notion of *latent price* [RR10], see Chapter V for further details.

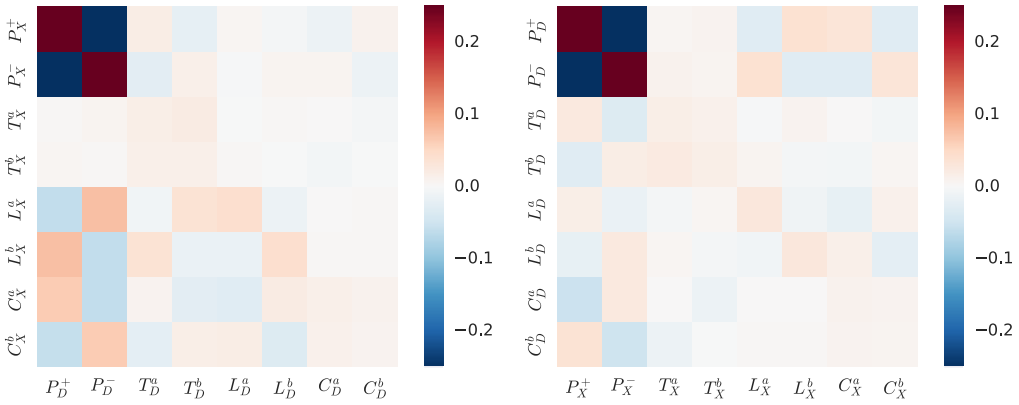


Figure .2: Submatrices of the Kernel norm matrix $\boldsymbol{G}$ corresponding to the effect of DAX events on EUROSTOXX STOXX events (left) and vice versa (right).

Part I

Large-scale Cox model

CHAPTER I

Background on SGD algorithms, Point Processes and Cox proportional hazards model

1 SGD algorithms

Objectives that are decomposable as a sum of a number of terms come up often in applied mathematics and scientific computing. They are particularly prevalent in machine learning applications, where one wants to minimize the average loss function over all observations. In the last two decades research on optimisation problems with a summation structure has focused more on the stochastic approximation setting, where the summation is assumed to be over an infinite set of terms [NJLS09, DS09, BCN16, Bot98]. The finite sum case has seen a resurgence in recent years after the discovery that there exist fast stochastic incremental gradient methods whose convergence rates are better deterministic first order methods. We provide a survey of fast stochastic gradient methods in the later parts of this section.

1.1 Definitions

In this work, we particularly focus on problems that have convex objectives. This is a major restriction, and one at the core of much of modern optimization theory. The primary reasons for targeting convex problems are their widespread use in applications and their relative ease of solving them. For convex problems, we can almost always establish theoretical results giving a practical bound on the amount of computation time required to solve a given convex problem [NN94]. Convex optimisation is still of interest when addressing non-convex problems though: many algorithms that were developed for convex problems, motivated by their provably fast convergence have later been applied to non-convex problems with good empirical results [GBC16].

We denote ∇f the gradient of f, $\nabla^2 f$ its Hessian matrix and $||\cdot||$ the Eucliean norm. Let now define some useful notions.

Definition 1. *A function f is L-smooth with $L > 0$ if f is diffentiable and its gradient is Lipschitz continuous, that is*

$$\forall \theta, \theta' \in \mathbb{R}^d, \qquad ||\nabla f(\theta) - \nabla f(\theta')|| \leq L||\theta - \theta'||.$$

If the function f is twice differentiable, the definition can be equivalently written:

$$\forall \theta \in \mathbb{R}^d, \qquad |eigenvalues[\nabla^2 f(\theta)]| \leq L.$$

The other assumption we will sometimes make is that of strong convexity.

Definition 2. *A function f is μ-strongly convex if:*

$$\forall \theta, \theta' \in \mathbb{R}^d, \forall t \in [0,1], \qquad f(t\theta + (1-t)\theta') \leq tf(\theta) + (1-t)f(\theta') - t(1-t)\frac{\mu}{2}||\theta - \theta'||^2.$$

If f is differentiable, the definition can be equivalently written:

$$\forall \theta, \theta' \in \mathbb{R}^d, \qquad f(\theta') \geq f(\theta) + \nabla f(\theta)^\top(\theta' - \theta) + \frac{\mu}{2}||\theta' - \theta||^2.$$

If the function f is twice differentiable, the definition can be equivalently written:

$$\forall \theta \in \mathbb{R}^d, \qquad |eigenvalues[\nabla^2 f(\theta)]| \geq \mu.$$

Gradient descent based algorithms can be easily extended to non-differentiable objectives F if they write $F(\theta) = f(\theta) + h(\theta)$ with f convex and differentiable, and h convex and non-differentiable whose *proximal operator* is easy to compute.

Definition 3. *Given a convex function h, we define its proximal operator as*

$$prox_h(x) = \underset{y}{arg\min}\left[h(y) + \frac{1}{2}||x - y||^2\right],$$

which is well-defined because of the strict convexity of the ℓ_2-norm.

The proximal operator can be seen as a generalization of the projection. Indeed, if $h = 0$ on $\mathscr{C}$ and $h = \infty$ on $\bar{\mathscr{C}}$, prox_h is exactly the projection over $\mathscr{C}$. The computation of the proximal operator is also an optimization problem, but when the function h is simple enough, the proximal operator has a closed form solution. Using these proximal operators, most algorithms enjoy the same theoretical convergence rates as if the objective was differentiable (i.e. $F(\theta) = f(\theta)$).

1.2 SGD algorithms from a general distribution

A variety of statistical and machine learning optimization problems writes

$$\min_{\theta \in \mathbb{R}^d} F(\theta) = f(\theta) + h(\theta) \qquad \text{with} \qquad f(\theta) = \mathbb{E}^\xi[\ell(\theta, \xi)],$$

where f is a goodness of fit measure depending implicitly on some observed data, h is a regularization term that imposes structure to the solution and ξ is a random variable. Typically, f is a differentiable function with a Lipschitz gradient, whereas h might be non-smooth (typical examples include sparsity inducing penalty).

First-order optimization algorithms are all variants of *Gradient Descent* (GD), which can be traced back to Cauchy [Cau47]. Starting at some initial point θ^0, this algorithm minimizes a differentiable function by iterating steps proportional to the negative of the gradient, as explained in Algorithm 1.

Algorithm 1 Gradient Descent (GD)

initialize θ
while not converged **do**
  $\theta \leftarrow \theta - \eta \nabla f(\theta)$
end while
return θ

Stochastic Gradient Descent (SGD) algorithms focus on the case where ∇f is intractable or at least time-consuming to compute. Noticing that $\nabla f(\theta)$ writes as an expectation like f, one idea is to approximate the gradient in the update step in Algorithm 1 with a Monte Carlo Markov Chain [AFM17]. Replacing the exact gradient $\nabla f(\theta)$ with its MCMC estimate is a general approach that enabled a significant step forward in training Undirected Graphical Models [Hin02] and Restricted Boltzmann Machines [HS06]. This form of Stochastic Gradient Descent is called *Contrastive Divergence* in the mentionned context.

Approximating the gradient of an expectation, sometimes named the *score function* [CH79], is a recurrent task for many other problems. Among them, we can cite posterior computation in variational inference [RMW14], value function and policy learning in reinforcement learning [PB11], derivative pricing [BG96], inventory control in operation research [Fu06] and optimal transport theory [GM98].

1.3 SGD algorithms from a uniform distribution

Most machine learning optimization problems involve a data fitting loss function f averaged over the uniform distribution, for instance when f is the average loss function over each observation of the data set. Namely, the optimization problem to solve writes

$$\min_{\theta \in \mathbb{R}^d} F(\theta) = f(\theta) + h(\theta) \qquad \text{with} \qquad f(\theta) = \frac{1}{n} \sum_{i=1}^{n} f_i(\theta),$$

where n is the number of observations, and f_i is the loss associated to the i^{th} observation. In that case, instead of running MCMC to approximate ∇f, one uniformly samples a random integer i between 1 and n and replace $\nabla f(\theta)$ with $\nabla f_i(\theta)$ in the update step, as shown in Algorithm 2. In the litterature, Stochastic Gradient Descent implicitly refers to the uniform

distribution case. In the large-scale setting, computing $\nabla f(\theta)$ at each update step represents the bottleneck of the minimization algorithm, and SGD helps decreasing the computation time.

Algorithm 2 Stochastic Gradient Descent (SGD)

initialize θ as the zero vector
while not converged **do**
  pick $i \sim \mathcal{U}[n]$
  $\theta \leftarrow \theta - \eta \nabla f_i(\theta)$
end while
return θ

Assuming the computation of each $\nabla f_i(\theta)$ costs 1, the computation of the full gradient $\nabla f(\theta)$ costs n, meaning SGD's update step is n times faster than GD's one.

The comparison of the convergence rates is however different. Consider f L-smooth and convex and denote θ^* its minimizer. We define the *condition number* $\kappa = L/\mu$. The convergence rate is measured via the difference $f(\theta^t) - f(\theta^*)$. Using the algorithm Gradient Descent with $\eta = 1/L$, the convergence rates are:

$$f(\theta^t) - f(\theta^*) \leq O\left(\frac{1}{t}\right),$$
$$f(\theta^t) - f(\theta^*) \leq O\left(e^{-t/\kappa}\right) \text{ if } f \text{ is } \mu\text{-strongly convex.}$$

The latter convergence rate which geometrically decrease the error is called *linear convergence rate* since the error decrease after one iteration is at worst linear. The convergence (in expectation) of the sequence (θ^t) produced by the algorithm Stochastic Gradient Descent need the step sizes to decrease to zero a specific way, see [RM51] for a general characterization. The convergence rate of stochastic algorithms is measured via the difference $\mathbb{E} f(\overline{\theta^t}) - f(\theta^*)$. Assuming each function f_i is L-Lipschitz (and not L-smooth) and f is convex, denoting $\overline{\theta}^t = \frac{1}{t}\sum_{u=1}^{t} \theta^u$, the convergence rates of Stochastic Gradient Descent are:

$$\mathbb{E} f(\overline{\theta^t}) - f(\theta^*) \leq O\left(\frac{1}{\sqrt{t}}\right) \quad \text{with} \quad \eta_t = \frac{1}{L\sqrt{t}},$$
$$\mathbb{E} f(\overline{\theta^t}) - f(\theta^*) \leq O\left(\frac{\kappa}{t}\right) \quad \text{with} \quad \eta_t = \frac{1}{\mu t} \quad \text{if } f \text{ is } \mu\text{-strongly convex.}$$

Convergence rates with other assumptions on the function f can be found in [B+15]. Recently, different works improved Stochastic Gradient Descent using variance reduction techniques from Monte Carlo methods. The idea is to add a *control variate* term to the descent direction to improve the bias-variance tradeoff in the approximation of the real gradient $\nabla f(\theta)$. Those variants also enjoy linear convergence rates with constant step-sizes.

1.4 SGD with Variance Reduction

The control variable is a variance reduction technique used in Monte Carlo methods [Gla13]. Its principle consists in estimating the population mean $\mathbb{E}(X)$ while reducing the variance of sample of X by using a sample from another variable Y with known expectation. We define a family of estimators

$$Z_\alpha = \alpha(X - Y) + \mathbb{E}(Y) \qquad \alpha \in [0, 1],$$

whose expectation and variance equal

$$\begin{aligned}\mathbb{E}(Z_a) &= \alpha\mathbb{E}(X) + (1-\alpha)\mathbb{E}(Y),\\ \mathbb{V}(Z_a) &= \alpha^2[\mathbb{V}(X) + \mathbb{V}(Y) - 2\mathrm{cov}(X, Y)].\end{aligned}$$

The case $\alpha = 1$ provides an unbiased estimator, while $0 < \alpha < 1$ implies Z_α to be biased with reduced variance. This control variates is particularly useful when Y is positively correlated with X.

The authors of [JZ13] observed that the variance induced by SGD's descent direction can only decrease to zero if decreasing step sizes are used, which prevents from linear convergence rate. In their work, they propose a variance reduction approach on the descent direction so as to use constant step sizes and obtain a linear convergence rate. The algorithms SAG [RSB12, SLRB17], SVRG [JZ13, XZ14], SAGA [DBLJ14] and SDCA [SSZ13] can be phrased with the variance reduction approach described above. Update steps of SAG, SAGA and SVRG with $i \sim \mathscr{U}[n]$ respectively write this way:

$$\text{(SAG)} \qquad \theta \leftarrow \theta - \eta\left(\frac{\nabla f_i(\theta) - y_i}{n} + \frac{1}{n}\sum_{j=1}^{n} y_j\right),$$

$$\text{(SAGA)} \qquad \theta \leftarrow \theta - \eta\left(\nabla f_i(\theta) - y_i + \frac{1}{n}\sum_{j=1}^{n} y_j\right),$$

$$\text{(SVRG)} \qquad \theta \leftarrow \theta - \eta\left(\nabla f_i(\theta) - \nabla f_i(\tilde{\theta}) + \frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\theta})\right).$$

From the control variate interpreation, we observe that SAG's descent direction is a biased estimate ($\alpha = 1/n$) of the gradient $\nabla f(\theta)$, while SAGA's and SVRG's ones are unbiased ($\alpha = 1$).

Stochastic Average Gradient (SAG) At each iteration, the algorithm SAG [RSB12] computes one gradient ∇f_i with the up-to-date value of θ, like SGD, and then descend in the direction of the average of the most recently computed gradients ∇f_j with equals weights, see Algorithm 3. Even though some gradients in the summation haven't been updated recently, the algorithm enjoys a linear convergence rate in the strongly-convex case. SAG can be regarded as a stochastic version of Incremental Average Gradient [BHG07], which has the same update with a different constant factor, and with cyclic computation of the gradient instead of

randomised. The convergence rates in the convex and strongly-convex cases with $\eta = 1/(16L)$ respectively involves the average iterate $\overline{\theta}^t$ and the iterate θ^t:

$$\mathbb{E} f(\overline{\theta}^t) - f(\theta^*) \leq O\left(\frac{1}{t}\right)$$
$$\mathbb{E} f(\theta^t) - f(\theta^*) \leq O\left(e^{-t\left(\frac{1}{8n} \wedge \frac{1}{16\kappa}\right)}\right) \qquad \text{if } f \text{ is } \mu\text{-strongly convex.}$$

The algorithm SAG is adaptative to the level of convexity of the problem, as it may be used with the same step size on both convex and strongly convex problems.

Algorithm 3 Stochastic Average Gradient (SAG)

initialize θ as the zero vector, $y_i = \nabla f_i(\theta)$ for each i
while not converged **do**
  $\theta \leftarrow \theta - \frac{\eta}{n} \sum_{j=1}^n y_j$
  pick $i \sim \mathcal{U}[n]$
  $y_i \leftarrow \nabla f_i(\theta)$
end while
return θ

Stochastic Variance Reduced Gradient (SVRG) The SVRG algorithm [XZ14, JZ13] is a recent stochastic gradient algorithm with variance reduction with linear convergence rate, given in Algorithm 4. Unlike SAG and SAGA, there is another parameter m to tune, which controls the update frequency of the control variate $\tilde{\theta}$. The algorithm S2GD [KR13] was developed at the same time, and has the same update as SVRG. The difference lies in the update of the control variate $\tilde{\theta}$:

- **Option I**: $\tilde{\theta}$ is the average of the θ values from the last m iterations, used in [JZ13].
- **Option II**: $\tilde{\theta}$ is a randomly sampled θ from the last m iterations, used for S2GD [KR13].

Consider f μ-strongly convex, a step size $\eta < 1/(2L)$, and assume m is sufficently large so that

$$\rho = \frac{1}{\mu\eta(1-2L\eta)m} + \frac{2L\eta}{1-2L\eta} < 1,$$

then the SVRG algorithm has a linear convergence rate if t is a multiple of m:

$$\mathbb{E} f(\tilde{\theta}^t) - f(\theta^*) \leq O\left(\rho^{t/m}\right).$$

Let us mention that SVRG does not require the storage of full gradients, on the contrary to SDCA, SAG and SAGA. The algorithm just stores the gradient $\nabla f(\tilde{\theta})$ and re-evaluates the gradient $\nabla f_i(\tilde{\theta})$ at each iteration.

Algorithm 4 Stochastic Variance Reduced Gradient (SVRG)

initialize θ and $\tilde{\theta}$ as zero vectors, t as zero
while not converged **do**
  pick $i \sim \mathcal{U}[n]$
  $\theta \leftarrow \theta - \eta(\nabla f_i(\theta) - \nabla f_i(\tilde{\theta}) + \nabla f(\tilde{\theta}))$
  $t \leftarrow t + 1$
  if t is a multiple of m **then**
    update $\tilde{\theta}$ with option **I** or **II**
  end if
end while
return θ

SAGA The algorithm SAGA [DBLJ14], described in Algorithm 5, enjoys a linear convergence rate in the strongly convex case, like SAG and SVRG, but it has the advantage with respect to SAG that it allows non-smooth penalty terms such as ℓ_1 regularization. The proof of the convergence rate is easier as well, especially because SAG's descent direction is a biased estimate of the gradient, while SAGA's one is unbiased. As SAG, the algorithm SAGA maintains the current iterate θ and a table of historical gradients.

The convergence rate of the algorithm SAGA writes:

$$\mathbb{E} f(\overline{\theta^t}) - f(\theta^*) \leq O\left(\frac{n}{t}\right) \qquad \text{with} \qquad \eta = \frac{1}{3L},$$
$$\mathbb{E}||\theta^t - \theta^*||^2 \leq O\left(e^{-\frac{t}{2(n+\kappa)}}\right) \qquad \text{with} \qquad \eta = \frac{1}{2(\mu n + L)} \qquad \text{if } f \text{ is } \mu\text{-strongly convex.}$$

Algorithm 5 SAGA

initialize θ as the zero vector, $y_i = \nabla f_i(\theta)$ for each i
while not converged **do**
  pick $i \sim \mathcal{U}[n]$
  $\theta \leftarrow \theta - \eta\left(\nabla f_i(\theta) - y_i + \frac{1}{n}\sum_{j=1}^n y_j\right)$
  $y_i \leftarrow \nabla f_i(\theta)$
end while
return θ

Composite case In the paragraphs above, we gave the convergence rates of the algorithm in the smooth case *i.e.* when the objective function to minimize is a smooth function. When the objective function is not smooth, one writes it as the sum of its smooth part $f(\theta)$ and its non-smooth part $h(\theta)$. One can easily adapt the previous algorithms by computing the gradient of the smooth part f and then project the iterate using the *proximal operator* of the non-smooth part h. This adds a projection step $\theta \leftarrow \text{prox}_h(\theta)$ at the end of each iteration.

Fortunately, the convergence rates stay the same, except for the algorithm [SLRB17], for which the authors haven't proved the convergence rate.

2 Point Processes

Point processes are useful to describe phenomena occurring at random locations and/or times. A point process is a random element whose values are point patterns on a set S. We present here the definitions and the useful results from point processes' theory. For further details, the book [DVJ07] is regarded as the main reference in the area of point processes.

2.1 Definitions

Let S be a locally compact metric space equipped with its Borel σ-algebra $\mathscr{B}$. Let X_S be the set of locally finite counting measures on S, and $\mathscr{N}_S$ the smallest σ-algebra on X_S such that all point counts $f_B : X_S \to \mathbb{N},\ \omega \mapsto \#(\omega \cap B)$ are measurable for B relatively compact in $\mathscr{B}$. A point process on S is a measurable map ξ from a probability space $(\Omega, \mathscr{F}, \mathbb{P})$ to the measurable space $(X_S, \mathscr{N}_S)$.

Every *realization* of a point process ξ can be written as $\xi = \sum_{i=1}^{n} \delta_{X_i}$ where δ is the Dirac measure, n is an integer-valued random variable and X_i's are random elements of S. A point process can be equivalenty represented by a *counting process*: $N(B) := \int_B \xi(x)dx$, which basically is the number of events in each Borel subset $B \in \mathscr{B}$. The *mean measure* M of a point process ξ is a measure on S that assigns to every $B \in \mathscr{B}$ the expected number of events of ξ in B, *i.e.*, $M(B) := \mathbb{E}[N(B)]$ for all $B \in \mathscr{B}$.

For *inhomogeneous Poisson process*, $M(B) = \int_B \lambda(x)dx$, where the intensity function $\lambda(x)$ yields a positive measurable function on S. Intuitively speaking, $\lambda(x)dx$ is the expected number of events in the infinitesimal dx. For the most common type of point process, a *homogeneous Poisson process*, $\lambda(x) = \lambda$ and $M(B) = \lambda|B|$, where $|\cdot|$ is the Lebesgue measure on $(S, \mathscr{B})$. More generally, we define *Cox point processes* - also known as *doubly stochastic Poisson processes* - as a generalization of Poisson processes where the intensity $\lambda(x)$ is itself a stationary stochastic process. Then, conditional on λ, the doubly stochastic Poisson process is simply an inhomogenous Poisson process with intensity $\lambda(x)$.

2.2 Temporal Point Processes

A particular interesting case of point processes is given when S is the time interval $[0, T)$, which we will call a *temporal point process*. Here, a realization is simply a set of time points: $\xi = \sum_{i=1}^{n} \delta_{t_i}$. With a slight notation abuse we will write $\xi = \{t_1, \dots, t_n\}$ where each t_i is a random time before T, and we define $N_t = \sum_{\tau \in \xi} \mathbb{1}_{\tau \leq t}$ the associated counting process. The *conditional intensity* function is the usual way to characterize temporal point processes where the present depends on the past. It is defined as the expected infinitesimal rate at which events are expected to occur after t, given the history of the counting process N_t prior to t.

Namely,

$$\lambda(t|\mathscr{F}_t) = \lim_{dt\to 0} \frac{\mathbb{P}(N_{t+dt} - N_t = 1|\mathscr{F}_t)}{dt},$$

where $\mathscr{F}_t$ is the natural filtration of the process, it represents the information available up to (but not including) the time t. The conditional intensity function is sometimes denoted $\lambda^*(t)$. The most simple temporal point process is the *homogeneous Poisson process* which assumes that the events arrive at a constant rate, which corresponds to a constant intensity function $\lambda(t|\mathscr{F}_t) = \lambda^*(t) = \lambda > 0$. More generally, we define the *inhomogeneous Poisson process* for which the conditional intensity function depends on t but not on the history *i.e.* $\lambda(t|\mathscr{F}_t) = \lambda^*(t) = \lambda(t)$.

The conditional intensity turns out to be interesting for multiple reasons. First, it is a convenient characterization of a temporal point process since it describes what is locally happening at t and is easy to interpret as an instantaneous probability. Secondly, the conditional intensity can be used for simulating a temporal point process: the basic idea is to simulate a Poisson process and use the cumulative conditional intensity to time scale the interevent times [Oga81]. Thirdly, the likelihood function can be expressed on closed form using the conditional intensity: if the point process is defined on $[0, T)$, then the likelihood and the log-likelihood functions are given by

$$L(\xi) = \left(\prod_{i=1}^{n} \lambda^*(t_i)\right) \exp\left(-\int_0^T \lambda^*(s)ds\right), \qquad \log L(\xi) = \sum_{i=1}^{n} \log \lambda^*(t_i) - \int_0^T \lambda^*(s)ds.$$

Finally, the conditional intensity function is useful for many other purposes, like a goodness-of-fit test known as residual analysis for point processes [Oga88], or the conditional distribution of interevent times between events [DVJ07]. We can also define the *compensator* $\Lambda(t)$ of the point process, with respect to $\mathscr{F}_t$, as the integral of the conditional intensity function: $\Lambda(t) = \int_0^t \lambda^*(s)ds$. We remind that $N_t - \Lambda(t)$ is then a $\mathscr{F}_t$-martingale.

We remind that the distribution of interevent times of a Poisson process with intensity λ is an exponential distribution of parameter λ. More generally, we denote $f^*(t)$ the conditional probability density function of the interevent time, t_n the last event that occured and T the random next one, $F^*(t) = \mathbb{P}(t_n \le T \le t|\mathscr{F}_t)$ the conditional cumulative density function, and $S^*(t) = 1 - F^*(t) = \mathbb{P}(T \ge t|\mathscr{F}_t)$ the survival function. Now,

$$\begin{aligned}
\lambda^*(t) &= \lim_{h\to 0} \frac{\mathbb{P}(t \le T \le t+h|T \ge t)}{h} \\
&= \lim_{h\to 0} \frac{1}{h} \frac{\mathbb{P}(t \le T \le t+h)}{\mathbb{P}(T \ge t)} \\
&= \lim_{h\to 0} \left(\frac{1}{h} \frac{f^*(t)h}{S^*(t)} + o(1)\right) \\
&= \frac{f^*(t)}{1 - F^*(t)}.
\end{aligned}$$

Conversely, we can write the likelihood function of the next event using the conditional intensity function:

$$f^*(t) = \lambda^*(t) \exp\left(-\int_{t_n}^{t} \lambda^*(s) ds\right).$$

This last formula enables writing a point process's realization's likelihood, already introduced above.

3 Cox proportional hazards model

3.1 Survival analysis

Survival analysis focuses on time-to-event data, such as the death in biological organisms and failure in mechanical systems, and is now widespread in a variety of domains like biometrics, econometrics and insurance [ABGK12]. The variable we study is the waiting time until a well-defined event occurs, and the main goal of survival analysis is to link the covariates, or features, of a patient to its survival time. We denote T the random variable of the time of death, we define the *survival function* as:

$$S(t) = \mathbb{P}(t \leq T).$$

However, and fortunately, not all affected patients die during a medical study and some patients can also leave the study before its end: we say that these observations are *right-censored*, in the sense that for some units the event of interest has not occured at the time the data are analyzed. The information about censored individual is incomplete, but it is still an information because one knows that an individual survived at least until the date he left the study. We will only study this kind of censoring in this part [1]. Let us now consider the probabilistic formulation for our framework: let T be a non-negative random variable representing the waiting time until the occurrence of an event (we will refer to this event as *failure* and to this waiting time as *failure time*). However, we don't always observe the random variable T since the patient can leave the study - before its death - at time C called the *censoring time*. Actually, we do observe $T \wedge C$ and we know if the patient died or left the study i.e. we know $\delta = \mathbb{1}_{\{T \leq C\}}$. We also assume that T and C are independent. We can now describe the model using counting processes.

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and $(\mathcal{F}_t)_{t \geq 0}$ a filtration satisfying the usual conditions. Let N be a point process with compensator Λ with respect to $(\mathcal{F}_t)_{t \geq 0}$ so that $N - \Lambda$ is a $(\mathcal{F}_t)_{t \geq 0}$-martingale. We denote $(T_1, ..., T_n)$ $i.i.d.$ copies of the random variable of interest T, corresponding to n different patients of a medical study for instance, $(C_1, ..., C_n)$ $i.i.d.$ copies of the censoring variable C and we define for each patient i: $\delta_i = \mathbb{1}_{\{T_i \leq C_i\}}$, the counting process $N_i(t) = \delta_i \mathbb{1}_{\{T_i \wedge C_i \leq t\}}$, and $Y_i(t) = \mathbb{1}_{\{T_i \wedge C_i \geq t\}}$, which is a predictable process. To understand the behavior of the counting process $N_i(t)$, we introduce its *intensity* $\alpha_i(t)$ defined as

[1] There are other types of censoring. For instance, left-censoring means the patient died or left the study before being observed : neglecting left-censoring will lead to overestimation of the survival time.

the conditional probability that the patient i dies immediately after t, given that he was alive before t:

$$\alpha(t) = \lim_{h \to 0} \frac{\mathbb{P}(t \leq T \leq t + h | t \leq T)}{h} = -\frac{S^{'}(t)}{S(t)}$$

Since the process can jump only once, the intensity of $N_i(t)$ takes the form $\alpha_i(t) = \lambda_i(t) Y_i(t)$, where $\lambda_i(t)$ is called the *hazard ratio*. We also introduce the *cumulative hazard* $\Lambda_i(t) = \int_0^t \lambda_i(s) ds$, which can be seen as the sum of the risks faced from 0 to t. Survival analysis generally aims at estimating either $S(t)$ or $\lambda(t)$ (or $\Lambda(t)$) given the observations of n individuals. Many approachs exist: the parametric one, which assumes that the functions can be described with a finite and small number of parameters, the nonparametric one, which assumes that the function of interest belongs to a certain class of smooth functions and the semi-parametric one, that has parametric and non-parametric components. The most popular approach, for some reasons explained below, is Cox proportional hazards model. The Cox model [Dav72] assumes a semi-parametric form for the hazard ratio at time t for the patient i, whose features are encoded in the vector $x_i \in \mathbb{R}^d$:

$$\lambda_i(t) = \lambda_0(t) \exp(x_i^\top \theta)$$

where $\lambda_0(t)$ is a baseline hazard ratio, which can be regarded as the hazard ratio of a patient whose covariates $x = 0$. Two estimation approachs exist: either estimating λ_0 and θ which can be done via maximizing the *full likelihood* of the model [RZ11] [She15], or considering λ_0 a *nuisance* and only estimating θ via maximizing a *partial likelihood* $L(\theta)$ [Dav72]. This way of estimating suits clinical studies where physicians are only interested in the effects of the covariates encoded in x on the hazard ratio. This can be done with computing the ratio of hazard ratios from two different patients:

$$\frac{\lambda_i(t)}{\lambda_j(t)} = \exp((x_i - x_j)^\top \theta)$$

For that reason, Cox model is said to be a proportional hazards model.

However, maximizing such functions is a hard problem when we deal with large-scale (meaning large n) and high-dimensional (meaning large d) data. To tackle to high-dimensionality, sparse penalized approaches have been considered in the literature [Tib96] [T$^+$97] [Goe10]. The problem is now to minimize the negative of the partial log-likelihood $-\ell(\theta)$ with a penalization that make the predictor θ become sparse and then select variables. We will discuss further this approach and the different models. On the contrary, approaches to tackle the large-scale side of the problem do not yet exist. We give an answer to this question in the following chapter.

3.2 Existing methods

The maximization of the partial likelihood $L_P(\theta)$ introduced in [Dav72] enables the estimation of θ - without the estimation of λ_0. The partial likelihood writes:

$$L_P(\theta) = \prod_{i=1}^{n} \left(\frac{\exp(x_i^\top \theta)}{\sum_{j \in R_i} \exp(x_j^\top \theta)} \right)^{\delta_i} \tag{1}$$

We prove in appendix that the negative of the partial log-likehood is convex, then the issue of *finding the θ that match our data* can be expressed as a classical convex optimization problem. We will consider the problem of maximizing the partial likelihood of the Cox model in the rest of this chapter.

However, in case of large-scale (meaning large n) and high-dimensional (meaning large p) data, this function becomes hard to maximize. To tackle to high-dimensionality, sparse penalized approaches have been considered in the literature. The problem is now to minimize the negative of the partial log-likelihood $-\ell(\theta) + \text{pen}(\theta)$ i.e.

$$\frac{1}{n}\sum_{i=1}^{n}\delta_i\left[-x_i^\top\theta + \log\left(\sum_{j\in R_i}\exp(x_j^\top\theta)\right)\right] + \text{pen}(\theta)$$

where $\text{pen}(\theta)$ is a penalization term that make the predictor θ become sparse and then select variables. For instance, the sparse penalties Lasso [Tib96] [T$^+$97], Elastic-Net [SFHT11] [YZ12], SCAD [FL01], Adaptative Lasso [Zou06], Graphical Lasso [FHT08], SLOPE [BvdBS$^+$15] and others.

Indeed, the *Lasso penalty* [Tib96]

$$\text{pen}^{\text{lasso}}(\theta) = \lambda||\theta||_1$$

can be used to obtain a penalized partial likelihood estimator $\widehat{\theta}$ [Goe10]. The lasso penalty tends to select only a few nonzero coefficients and does not handle well very correlated predictors: it will pick one and ignore the other.

Another well-known penalty called *Ridge penalty* $\text{pen}^{\text{ridge}}(\theta) = \frac{\lambda}{2}||\theta||_2^2$ tends to shrink all coefficients to zero and give equal weights to very correlated predictors. Zhou and Hastie [SFHT11] combined the strengths of the two approaches with the *Elastic-Net penalty*, where $\alpha \in [0,1]$ controls the behavior of the penalty:

$$\text{pen}^{\text{e-net}}(\theta) = \lambda\left(\alpha||\theta||_1 + \frac{1}{2}(1-\alpha)||\theta||_2^2\right)$$

The authors of [GTPV14] studied electronic medical records and used a sparse penalty which encodes the *a priori* relationship between predictors i and j: $A_{ij} = 1$ if predictors i and j share a temporal or well-known relation, $A_{ij} = 0$ otherwise.

$$\text{pen}(\theta) = \lambda_1||\theta||_1 + \frac{1}{2}\lambda_2\sum_{i,j}A_{ij}(\theta_i - \theta_j)^2$$

These methods handle the high-dimensional side of the dataset, but don't look relevant when the number of patients n is large. Indeed, the higher the number of examples n, the higher the time to compute the sum of loss functions (here, the negative of the penalized log-likelihood) and time can be the limiting factor when one envisions very large datasets. In the next chapter, we introduce a new stochastic algorithm with variance reduction that enables a faster minimization of the negative partial likelihood of the Cox model.

CHAPTER II

Large-scale Cox model

Abstract

We introduce a doubly stochastic proximal gradient algorithm for optimizing a finite average of smooth convex functions, whose gradients depend on numerically expensive expectations. Indeed, the effectiveness of SGD-like algorithms relies on the assumption that the computation of a subfunction's gradient is cheap compared to the computation of the total function's gradient. This is true in the Empirical Risk Minimization (ERM) setting, but can be false when each subfunction depends on a sequence of examples. Our main motivation is the acceleration of the optimization of the regularized Cox partial-likelihood (the core model in survival analysis), but other settings can be considered as well.

The proposed algorithm is doubly stochastic in the sense that gradient steps are done using stochastic gradient descent (SGD) with variance reduction, and the inner expectations are approximated by a Monte-Carlo Markov-Chain (MCMC) algorithm. We derive conditions on the MCMC number of iterations guaranteeing convergence, and obtain a linear rate of convergence under strong convexity and a sublinear rate without this assumption.

We illustrate the fact that our algorithm improves the state-of-the-art solver for regularized Cox partial-likelihood on several datasets from survival analysis.

Keywords. Convex Optimization, Stochastic Gradient Descent, Monte Carlo Markov Chain, Survival Analysis, Conditional Random Fields

1 Introduction

During the past decade, advances in biomedical technology have brought high dimensional data to biostatistics and survival analysis in particular. Today's challenge for survival analysis lays in the analysis of massively high dimensional (numerous covariates) and large-scale (large number of observations) data, see in particular [MD13]. Areas of application outside of biostatistics, such as economics (see [EL14]), or actuarial sciences (see [Ric12]) are also concerned.

One of the core models of survival analysis is the Cox model (see [Dav72]) for which we propose, in the present paper, a novel scalable optimization algorithm tuned to handle

massively high dimensional and large-scale data. Survival data $(y_i, x_i, \delta_i)_{i=1}^{n_{\text{pat}}}$ contains, for each individual $i = 1, \dots, n_{\text{pat}}$, a features vector $x_i \in \mathbb{R}^d$, an observed time $y_i \in \mathbb{R}_+$, which is a failure time if $\delta_i = 1$ or a right-censoring time if $\delta_i = 0$. If $D = \{i : \delta_i = 1\}$ is the set of patients for which a failure time is observed, if $n = |D|$ is the total number of failure times, and if $R_i = \{j : y_j \geq y_i\}$ is the index of individuals still at risk at time y_i, the negative Cox partial log-likelihood writes

$$-\ell(\theta) = \frac{1}{n} \sum_{i \in D} \Big[-x_i^\top \theta + \log\Big(\sum_{j \in R_i} \exp(x_j^\top \theta) \Big) \Big] \tag{1}$$

for parameters $\theta \in \mathbb{R}^d$. This model can be regarded as a regression of the n failure times, using information from the n_{pat} patients that took part to the study. With high-dimensional data, a regularization term is added to the partial likelihood to automatically favor sparsity in the estimates, see [T$^+$97] and [SFHT11] for a presentation of Lasso and elastic-net penalizations, see also the review paper by [WT09] for an exhaustive presentation. Several algorithms for the Cox model have been proposed to solve the regularized optimization problem at hand, see [PH07, SKJP09, Goe10] among others. These implementations use Newton-Raphson iterations, i.e. large matrices inversions, and can therefore not handle large-scale data. Cyclical coordinate descent algorithms have since been proposed and successfully implemented in `R` packages `coxnet` and `fastcox`, see [SFHT11, YZ12]. More recently [MMCB13] adapted the column relaxation with logistic loss algorithm of [ZO00] to the Cox model. The fact that all these algorithms are of cyclic coordinate descent type solve the problem, supported by Newton-Raphson type algorithms, of large matrices inversions.

Yet another computationnally costly problem, specific to the Cox model, has not been fully addressed: the presence of cumulative sums (over indices $j \in R_i$) in the Cox partial likelihood. This problem was noticed in [MMCB13], where a numerical workaround exploiting sparsity is proposed to reduce the computational cost. The cumulative sum prevents from successfully applying stochastic gradient algorithms, which are however known for their efficiency to handle large scale generalized linear models: see for instance SAG by [SLRB17], SAGA by [DBLJ14], Prox-SVRG by [XZ14] and SDCA by [SSZ12] that propose very efficient stochastic gradient algorithms with constant step-size (hence achieving linear rates), see also Catalyst by [LMH15] that introduces a generic scheme to accelerate and analyze the convergence of those algorithms.

Such recent stochastic gradient algorithms have shown that it is possible to improve upon proximal full gradient algorithms for the minimization of convex problems of the form

$$\min_{\theta \in \mathbb{R}^d} F(\theta) = f(\theta) + h(\theta) \text{ with } f(\theta) = \frac{1}{n} \sum_{i=1}^{n} f_i(\theta), \tag{2}$$

where the functions f_i are gradient-Lipschitz and h is prox-capable. These algorithms take advantage of the finite sum structure of f, by using some form of variance-reduced stochastic gradient descent. It leads to algorithms with a much smaller iteration complexity, as compared to proximal full gradient approach (FG), while preserving (or even improving) the linear convergence rate of FG in the strongly convex case. However, such algorithms are relevant

when gradients ∇f_i have a numerical complexity much smaller than ∇f, such as for linear classification or regression problems, where ∇f_i depends on a single inner product $x_i^\top \theta$ between features x_i and parameters θ.

In this paper, motivated by the important example of the Cox partial likelihood (1), we consider the case where gradients ∇f_i can have a complexity comparable to the one of ∇f. More precisely, we assume that they can be expressed as expectations, under a probability measure π_θ^i, of random variables $G_i(\theta)$, i.e.,

$$\nabla f_i(\theta) = \mathbb{E}^{G_i(\theta) \sim \pi_\theta^i}[G_i(\theta)]. \tag{3}$$

This paper proposes a new doubly stochastic proximal gradient descent algorithm (2SVRG), that leads to a low iteration complexity, while preserving linear convergence under suitable conditions for problems of the form (2) + (3).

Our main motivation for considering this problem is to accelerate the training-time of the the penalized Cox partial-likelihood. The function $-\ell(\theta)$ is convex (as a sum of linear and log-sum-exp functions, see Chapter 3 of [BV04], and fits in the setting (2) + (3). Indeed, fix $i \in D$ and introduce

$$f_i(\theta) = -x_i^\top \theta + \log\Big(\sum_{j \in R_i} \exp(x_j^\top \theta) \Big),$$

so that

$$\nabla f_i(\theta) = -x_i + \sum_{j \in R_i} x_j \pi_\theta^i(j)$$

where

$$\pi_\theta^i(j) = \frac{\exp(x_j^\top \theta)}{\sum_{j' \in R_i} \exp(x_{j'}^\top \theta)}, \quad \forall j \in R_i.$$

This entails that $\nabla f_i(\theta)$ satisfies (3) with $G_i(\theta)$ a random variable valued in $\{-x_i + x_j : j \in R_i\}$ and such that

$$\mathbb{P}(G_i(\theta) = -x_i + x_j) = \pi_\theta^i(j)$$

for $j \in R_i$. Note that the numerical complexity of ∇f_i can be comparable to the one of ∇f, when y_i is close to $\min_i y_i$ (recalling that $R_i = \{j : y_j \geq y_i\}$). Note also that a computational trick allows to compute $\nabla f(\theta)$ with a complexity $O(nd)$. Indeed, once all data points are sorted, the sum can be computed recursively. This makes this setting quite different from the usual case of empirical risk minimization (linear regression, logistic regression, etc.), where all the gradients ∇f_i share the same low numerical cost.

2 Comparison with previous work

SGD techniques. Recent proximal stochastic gradient descent algorithms by [DBLJ14], [XZ14], [SSZ12] and [SLRB17] build on the idea of [RM51] and [KW+52]. Such algorithms are designed to tackle large-scale optimization problems (n is large), where it is assumed implicitly that the ∇f_i (smooth gradients) have a low computational cost compared to ∇f, and where h is

eventually non-differentiable and is dealt with using a backward or projection step using its proximal operator.

The principle of SGD is, at each iteration t, to sample uniformly at random an index $i \sim \mathcal{U}[n]$, and to apply an update step of the form

$$\theta^{t+1} \leftarrow \theta^t - \gamma_t \nabla f_i(\theta^t).$$

This step is based on an unbiased but very noisy estimate of the full gradient ∇f, so the choice of the step size γ_t is crucial since it has to be decaying to curb the variance introduced by random sampling (excepted for averaged SGD in some particular cases, see [BM13]). This tends to slow down convergence to a minimum $\theta^\star \in \operatorname{argmin}_{\theta \in \mathbb{R}^d} f(\theta)$. Gradually reducing the variance of ∇f_i for $i \sim \mathcal{U}[n]$ as an approximation of ∇f allows to use larger – even constant – step sizes and to obtain faster convergence rates. This is the underlying idea of two recent methods - SAGA and SVRG respectively introduced in [DBLJ14], [XZ14] - that use updates of the form

$$w^{t+1} \leftarrow \theta^t - \gamma\Big(\nabla f_i(\theta^t) - \nabla f_i(\tilde{\theta}) + \frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\theta})\Big),$$

and $\theta^{t+1} \leftarrow \mathbf{prox}_{\gamma h}(w^{t+1})$. In [XZ14], $\tilde{\theta}$ is fully updated after a certain number of iterations, called *phases*, whereas in [DBLJ14], $\tilde{\theta}$ is partially updated after each iteration. Both methods use stochastic gradient descent steps, with variance reduction obtained via the centered control variable $-\nabla f_i(\tilde{\theta}) + \frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\theta})$, and achieve linear convergence when F is strongly-convex, namely $\mathbb{E}F(\theta^k) - \min_{\in \mathbb{R}^d} F(x) = \ O(\rho^k)$ with $\rho < 1$, which make these algorithms state-of-the-art for many convex optimization problems. Some variants of SVRG [XZ14] also approximate the full gradient $\frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\theta})$ using mini-batchs to decrease the computing time of each phase, see [LJ17, HAV+15].

Numerically hard gradients. A very different, nevertheless classical, "trick" to reduce the complexity of the gradient computation, is to express it, whenever the statistical problem allows it, as the expectation, with respect to a non-uniform distribution π_θ, of a random variable $G(\theta)$, i.e., $\nabla f(\theta) = \mathbb{E}^{G(\theta)\sim\pi_\theta}[G(\theta)]$. Optimization problems with such a gradient have generated an extensive literature from the first works by [RM51], and [KW+52]. Some algorithms are designed to construct stochastic approximations of the sub-gradient of $f + h$, see [NJLS09, JN+11, Lan12, DHS11]. Others are based on proximal operators to better exploit the smoothness of f and the properties of h, see [HPK09, Xia10, AFM17]. In this paper, we shall focus on the second kind of algorithms. Indeed, our approach is closer to the one developed in [AFM17], though, as opposed to ours, the algorithm developed in this latter work is based on proximal full gradient algorithms (not doubly stochastic as ours) and does not guarantee a linear convergence.

Contrastive divergence. The idea to approximate the gradient using MCMC already appeared in the litterature of Undirected Graphical Models under the name of Contrastive Divergence, see [Mur12, Hin02, CPH05]. Indeeed, for this class of model, the gradient of the log-likelihood $\nabla f(\theta)$ can be written as the difference of two expectations: one - tractable

- with respect to the data discrete distribution $\mathbf{X}$, the other - intractable - with respect to the model-dependent distribution $p(\cdot,\theta)$. The idea of Contrastive Divergence relies in the approximation of the intractable expectation using MCMC, with few iterations of the chain. However, in the framework of Cox model, and also Conditional Random Fields (see Section 6 below), this is the gradient $\nabla f_i(\theta)$ that writes as an time-consuming expectation, see Equation 3.

Our setting. The setting of our paper is original in the sense that it combines both previous settings, namely stochastic gradient descent and MCMC. As in the stochastic gradient setting, the gradient can be expressed as the sum of n components, where n can be very large. However, since these components are time-consuming to compute directly, following the expectation based gradient computation setting, they are expressed as averaged values of some random variables. More precisely, the gradient $\nabla f_i(\theta)$ is replaced by an approximation $\widehat{\nabla} f_i(\theta)$ obtained by an MCMC algorithm. Our algorithm is, to the best of our knowledge, the first one to propose a combination of two stochastic approximations in this way, hence the name *doubly stochastic*, which allow to deal with both, eventual large values for n and the inner complexity of each gradient ∇f_i computation.

The idea to mix SGD and MCMC has also been raised recently in the very different setting of *implicit* stochastic gradient descent, see [TA14]. Note also that in our approach we make two stochastic approximations to the gradient using random training points, while the doubly stochastic approach from [DXH+14] performs two stochastic approximations to the gradient using random training points and random features for kernel methods.

3 A doubly stochastic proximal gradient descent algorithm

Our algorithm 2SVRG is built upon the algorithm SVRG via an approximation function ApproxMCMC. We first present the meta-algorithm without specifying the approximation function, and then provide two examples for ApproxMCMC.

3.1 2SVRG: a meta-algorithm

Following the ideas presented in the previous section, we design a *doubly stochastic proximal gradient descent algorithm* (2SVRG), by combining a variance reduction technique for SGD given by Prox-SVRG [XZ14], and a Monte-Carlo Markov-Chain algorithm to obtain an approximation of the gradient $\nabla f_j(\theta)$ at each step. Thus, in the considered setting the full gradient writes

$$\nabla f(\theta) = \mathbb{E}^{i\sim\mathscr{U}}[\nabla f_i(\theta)] = \mathbb{E}^{i\sim\mathscr{U}}\,\mathbb{E}^{G_i(\theta)\sim\pi_\theta^i}[G_i(\theta)],$$

where $\mathscr{U}$ is the uniform distribution on $\{1,\dots,n\}$, so our algorithm contains two levels of stochastic approximation: uniform sampling of i (the variance-reduced SGD part) for the first expectation, and an approximation of the second expectation w.r.t π_θ^i by means of Monte-Carlo simulation. The 2SVRG algorithm is described in Algorithm 6.

Following Prox-SVRG by [XZ14], this algorithm decomposes in *phases*: iterations within a phase apply variance reduced stochastic gradient steps (with a backward proximal step, see

Algorithm 6 Doubly stochastic proximal gradient descent (2SVRG)

1: **Require**: Number of phases $K \geq 1$, phase-length $m \geq 1$, step-size $\gamma > 0$, MCMC number of iterations per phase $(N_k)_{k=1}^K$, starting point $\theta^0 \in \mathbb{R}^d$
2: **Initialize**: $\tilde{\theta} \leftarrow \theta^0$ and compute $\nabla f_i(\tilde{\theta})$ for $i = 1, \dots, n$
3: **for** $k = 1$ **to** K **do**
4: **for** $t = 0$ **to** $m - 1$ **do**
5: Pick $i \sim \mathscr{U}[n]$
6: $\widehat{\nabla} f_i(\theta^t) \leftarrow \text{ApproxMCMC}(i, \theta^t, N_k)$
7: $d^t = \widehat{\nabla} f_i(\theta^t) - \nabla f_i(\tilde{\theta}) + \frac{1}{n} \sum_{j=1}^n \nabla f_j(\tilde{\theta})$
8: $\omega^{t+1} \leftarrow \theta^t - \gamma d^t$
9: $\theta^{t+1} \leftarrow \mathbf{prox}_{\gamma h}(\omega^{t+1})$
10: **end for**
11: Update $\tilde{\theta} \leftarrow \frac{1}{m} \sum_{t=1}^m \theta^t$, $\theta^0 \leftarrow \tilde{\theta}$, $\tilde{\theta}^k \leftarrow \tilde{\theta}$
12: Compute $\nabla f_i(\tilde{\theta})$ for $i = 1, \dots, n$
13: **end for**
14: **Return**: $\tilde{\theta}^K$

lines 7 and 8 in Algorithm 6). At the end of a phase, a full-gradient is computed (lines 10, 11) and used in the next phase for variance reduction. Within a phase, each inner iteration samples uniformly at random an index i (line 4) and obtains an approximation of the gradient ∇f_i at the previous iterate θ^t by applying N_k iterations of a Monte-Carlo Markov-Chain (MCMC) algorithm.

Intuitively, the sequence N_k should be increasing with the phase number k, as we need more and more precision as the iterations goes on (this is confirmed in Section 4). The important point of our algorithm resides precisely in this aspect: very noisy estimates can be used in the early phases of the algorithm, hence allowing for an overall low complexity as compared to a full gradient approach.

3.2 Choice of ApproxMCMC

We focus now on two implementations of the function ApproxMCMC based on two famous MCMC algorithms: Metropolis-Hastings and Importance Sampling.

3.2.1 Independent Metropolis-Hastings

When the π_θ^i are Gibbs probability measures, as for the previously described Cox partial log-likelihood (but for other models as well, such as Conditional Random Fields, see [LMP+01]), one can apply Independent Metropolis-Hastings (IMH), see Algorithm 7 below, to obtain approximations $\widehat{\nabla} f_i$ of the gradients. In this case the produced chain is geometrically uniformly ergodic, see [Rob04], and therefore meets the general assumptions required in our results (see Proposition 1 below). The IMH algorithm uses a proposal distribution Q which is independent of the current state j_l of the Markov chain.

In the case of the Cox partial log-likelihood, at iteration t of phase k of Algorithm 6, we set $\pi = \pi_{\theta^t}^i$, and Q to be the uniform distribution over the set R_i. We implemented two versions

Algorithm 7 Independent Metropolis-Hastings (IMH) estimator (for the Cox model)

Require: Proposal distribution $Q = \mathcal{U}\{R_i\}$, starting point $j_0 \in R_i$, stationary distribution $\pi = \pi^i_{\theta^t}$

for $l = 0, \dots, N_k - 1$ **do**

1. **Generate**: $j' \sim Q$.
2. **Update**: $\alpha = \min\left(\frac{\pi(j')Q(j_l)}{\pi(j_l)Q(j')}, 1\right) = \min\left(\exp((x_{j'} - x_{j_l})^\top \theta^t), 1\right)$.
3. **Take**: $j_{l+1} = \begin{cases} j' & \text{with probability } \alpha \\ j_l & \text{otherwise.} \end{cases}$

end for

Return: $-x_i + \frac{1}{N_k}\sum_{l=1}^{N_k} x_{j_l}$

of Algorithm 6 with IMH: one with a uniform proposal Q, the other one with an adaptative proposal $\widetilde{Q}$. When we want to approximate $\nabla f_i(\theta)$, we can consider the adaptative proposal $\widetilde{Q} = \pi^i_{\tilde{\theta}}$, where $\tilde{\theta}$ is the iterate we have computed at the end of the previous phase, see Line 10 of Algorithm 6. Since we compute the full gradient only once every phase, the probabilities $\pi^i_{\tilde{\theta}}(j)$ are computed at the same time, which means that the use of an adaptative proposal adds no computational effort. Morever, the theoretical guarantees given in Section 4 make no difference between the two versions aformentionned, but a strong difference is observed in practice.

3.2.2 Importance Sampling

To choice of the adaptative proposal above reduces the variance of the estimator given by ApproxMCMC. The idea of sampling with $\widetilde{Q} = \pi^i_{\tilde{\theta}}$ can also be used in an Importance Sampling estimator as well.

$$\nabla f_i(\theta) = \mathbb{E}^{G_i(\theta)\sim\pi^i_\theta}[G_i(\theta)] = \mathbb{E}^{G_i(\theta)\sim\widetilde{Q}}\left[G_i(\theta)\frac{\pi^i_\theta(G_i(\theta))}{\widetilde{Q}(G_i(\theta))}\right]$$

Since the ratio $\pi^i_\theta(G_i(\theta))/\widetilde{Q}(G_i(\theta))$ still contains an expensive term to compute, we can divide the term above with $\mathbb{E}_{\widetilde{Q}}\left[\pi^i_\theta(G_i(\theta))/\widetilde{Q}(G_i(\theta))\right] = 1$ and approximate the resulting term. This trick provides an estimator called Normalized Importance Sampling estimator, which writes like this in the case of Cox partial likelihood:

$$\begin{aligned} \widehat{J}_N &= \sum_{k=1}^N (x_{j_k} - x_i)\frac{\pi^i_\theta(j_k)}{\widetilde{Q}(j_k)} \Bigg/ \sum_{k=1}^N \frac{\pi^i_\theta(j_k)}{\widetilde{Q}(j_k)}, && \text{with } j_k \sim \widetilde{Q} \\ &= -x_i + \sum_{k=1}^N \frac{\exp((\theta-\tilde{\theta})^\top x_{j_k})}{\sum_{l=1}^N \exp((\theta-\tilde{\theta})^\top x_{j_l})} x_{j_k}, && \text{with } j_k \sim \widetilde{Q} \end{aligned}$$

Section 4 below gives theoretical guarantees for Algorithm 6: linear convergence under strong-convexity of F is given in Theorem 1, and a convergence without strong convexity is given in Theorem 2. This improves the proximal stochastic gradient method of [AFM17],

Algorithm 8 Normalized Importance Sampling (NIS) estimator of $\nabla f_i(\theta)$ (for the Cox model)

Require: Proposal distribution $\widetilde{Q} = \pi^i_{\tilde{\theta}}$, stationary distribution π^i_{θ}, $V = 0 \in \mathbb{R}^d, S = 0 \in \mathbb{R}$
for $l = 1, \dots, N_k$ **do**
1. **Generate**: $j_l \sim \widetilde{Q}(\cdot)$.
2. **Update**: $V \leftarrow V + \exp((\theta - \tilde{\theta})^\top x_{j_l}) x_{j_l}$.
3. **Update**: $S \leftarrow S + \exp((\theta - \tilde{\theta})^\top x_{j_l})$.

end for
Return: $-x_i + V/S$

where the best case rate is $O(1/k^2)$ using Fista (see [BT09]) acceleration scheme. Numerical illustrations are given in Section 5, where a fair comparison between several state-of-the-art algorithms is proposed.

4 Theoretical guarantees

Definitions. All the functions f_i and h are proper convex lower-semicontinuous on $\mathbb{R}^d$. The norm $\|\cdot\|$ stands for the Euclidean norm on $\mathbb{R}^d$. A function $f : \mathbb{R}^d \to \mathbb{R}$ is L-smooth if it is differentiable and if its gradient is L-Lipschitz, namely if $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ for all $x, y \in \mathbb{R}^d$. A function $f : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex if $f(x + y) \geq f(x) + \nabla f(x)^\top y + \frac{\mu}{2}\|y\|^2$ for all $x, y \in \mathbb{R}^d$ i.e. if $f - \frac{\mu}{2}\|\cdot\|^2$ is convex. The proximal operator of $h : \mathbb{R}^d \to \mathbb{R}$ is uniquely defined by $\text{prox}_h(x) = \text{argmin}_{y \in \mathbb{R}^d}\{h(y) + \frac{1}{2}\|x - y\|^2\}$.

Notations. We denote by i_t the index randomly picked at the t^{th} iteration, see line 4 in Algorithm 6. We introduce the error of the MCMC approximation $\eta^t = \widehat{\nabla} f_{i_t}(\theta^{t-1}) - \nabla f_{i_t}(\theta^{t-1})$ and the filtration $\mathscr{F}_t = \sigma(\theta^0, i_1, \theta^1, \dots, i_t, \theta^t)$. In order to analyze the descent steps, we need different expectations: $\mathbb{E}_t$ the expectation w.r.t the distribution of the pair $(i_t, \widehat{\nabla} f_{i_t}(\theta^{t-1}))$ conditioned on $\mathscr{F}_{t-1}$, and $\mathbb{E}$ the expectation w.r.t all the random iterates (i_t, θ^t) of the algorithm. We also denote $\theta^* = \text{argmin}_{\theta \in \mathbb{R}^d} F(\theta)$.

Assumptions.

Assumption 1. *We consider $F = f + h$ where $f = \frac{1}{n}\sum_{i=1}^n f_i$, with each f_i being convex and L_i-smooth, $L_i > 0$, and h a lower semi-continuous and closed convex function. We denote $L = \max_{1 \leq i \leq n} L_i$. We assume that there exists $B > 0$ such that the iterates θ^t satisfy $\sup_{t \geq 0} \|\theta^t - \theta^*\| \leq B$.*

Assumption 2. *We assume that the bias and the expected squared error of the Monte Carlo estimation can be bounded in the following way:*

$$\|\mathbb{E}_t \eta^t\| \leq \frac{C_1}{N_k} \text{ and } \mathbb{E}_t\|\eta^t\|^2 \leq \frac{C_2}{N_k} \tag{4}$$

for the iterations t belonging to the k-th phase, where N_k is the number of iterations of the Markov chain used for the computation of $\widehat{\nabla} f_{i_t}(\theta^t)$ during phase k (see line 5 of Algorithm 6), and where C_1 and C_2 are positive constants.

Let us point out that Proposition 1 below gives a sufficient condition for Assumption 2 to hold.

Theorems. The theorems below provide upper bounds on the distance to the minimum in the strongly convex case, see Theorem 1 and in the convex case, see Theorem 2.

Theorem 1. *Suppose that $F = f + h$ is μ-strongly convex. Consider Algorithm 6, with a phase length m and a step-size $\gamma \in (0, \frac{1}{16L})$ satisfying*

$$\rho = \frac{1}{m\gamma\mu(1-8L\gamma)} + \frac{8L\gamma(1+1/m)}{1-8L\gamma} < 1. \tag{5}$$

Then, under Assumption 1 and Assumption 2, we have:

$$\mathbb{E}[F(\tilde{\theta}^K)] - F(\theta^*) \leq \rho^K \Big(F(\theta^0) - F(\theta^*) + \sum_{l=1}^{K} \frac{D}{\rho^l N_l} \Big), \tag{6}$$

where $D = \frac{3\gamma C_2 + BC_1}{1-8L\gamma}$.

In Theorem 1, the choice $N_k = k^\alpha \rho^{-k}$ with $\alpha > 1$ gives

$$\mathbb{E}[F(\tilde{\theta}^K)] - F(\theta^*) \leq D' \rho^K$$

where $D' = F(\theta^0) - F(\theta^*) + D\sum_{k\geq 1} k^{-\alpha}$ and $D > 0$ is a numerical constant. This entails that 2SVRG achieves a *linear rate* under strong convexity.

Remark 1 (An important remark). *The number N_k of MCMC iterations is growing quickly with the phase number k. So, we use in practice an hybrid version of 2SVRG called* HSVRG*: 2SVRG is used for the first phases (usally 4 or 5 phases in our experiments), and as soon as N_k exceeds n, we switch to a mini-batch version of Prox-SVRG (SVRG-MB), see [Nit14]. A precise description of HSVRG is given in Algorithm 9 from Section 5 below. Note that overall linear convergence of HSVRG is still guaranteed, since both 2SVRG and SVRG-MB decrease linearly the objective from one phase to the other.*

Theorem 2. *Consider Algorithm 6, with a phase length m and a step-size $\gamma \in (0, \frac{1}{8L(2m+1)})$. Then, under Assumption 1 and Assumption 2, we have*:

$$\mathbb{E}[F(\bar{\theta}^K)] - F(\theta^*) \leq \frac{D_1}{K} + \frac{D_2}{K} \sum_{k=1}^{K+1} \frac{1}{N_k}, \tag{7}$$

where D_1 and D_2 depend on the constants of the problem, and where $\bar{\theta}^K$ is the average of iterates $\tilde{\theta}^k$ until phase K.

In Theorem 2, the choice $N_k = k^\alpha$ with $\alpha > 1$ gives

$$\mathbb{E}[F(\tilde{\theta}^K)] - F(\theta^*) \leq \frac{D_3}{K}$$

for a constant $D_3 > 0$. This result is an improvement of the Stochastic Proximal Gradient algorithm from [AFM17] since it is not necessary to design a weighted averaged but just a simple average to reach the same convergence rate. Also, it provides a convergence guarantee for the non-strongly convex case, which is not proposed in [XZ14].

Theorems 1 and 2 show a trade-off between the linear convergence of the variance-reduced stochastic gradient algorithm and the MCMC approximation error. The next proposition proves that Algorithm 7 satisfies Assumption 2 under a general assumption on the proposal and the stationary distribution.

Proposition 1. *Suppose that there exists $M > 0$ such that the proposal Q and the stationary distribution π satisfy $\pi(x) \leq MQ(x)$, for all x in the support of π. Then, the error η^t obtained by Algorithm 7 satisfies Assumption 2.*

Remark 2 (Specifics for the Cox partial likelihood). *Note that the assumptions required in Proposition 1 are met for the Cox partial likelihood: in this case, a simple choice is $M = n \max_{x \in supp(\pi)} \pi(x)$, and the Monte Carlo error η^t induced by computing the gradient of f_i at phase k using Algorithm 7 satisfies* (4) *with*

$$C_1 = \frac{2}{|R_i|} \max_{j \in R_i} \pi^i_{\theta^{t-1}}(j)$$
$$C_2 = 36 \mathscr{C}_2 C_1^2 (1 + C_1) \max_{j \in R_i} \|x_j\|_2^2,$$

where $\mathscr{C}_2$ is the Rosenthal constant of order 2, see Proposition 12 in [FM+ 03].

5 Numerical experiments

We compare several solvers for the minimization of the objective given by an elastic-net penalization of the Cox partial likelihood

$$F(\theta) = -\ell(\theta) + \lambda\Big(\alpha\|\theta\|_1 + \frac{1-\alpha}{2}\|\theta\|_2^2\Big),$$

where we recall that the partial likelihood ℓ is defined in Equation (1) and where $\lambda > 0$ and $\alpha \in [0, 1]$ are tuning parameters.

A fair comparison of algorithms. The doubly stochastic nature of the considered algorithms makes it hard to compare them to batch algorithms in terms of iteration number or epoch number (number of full passes over the data), as this is usually done for SGD-based algorithm. Hence, we proceed by plotting the evolution of $F(\tilde{\theta}) - F(\theta^*)$ (where $\theta^* \in \operatorname{argmin}_{u \in \mathbb{R}^d} F(u)$ and $\tilde{\theta}$ is the current iterate of a solver) as a function of the number of inner products between a feature vector x_i and θ, effectively computed by each algorithm, to obtain the current iterate $\tilde{\theta}$. This gives a fair way of comparing the effective complexity of all algorithms.

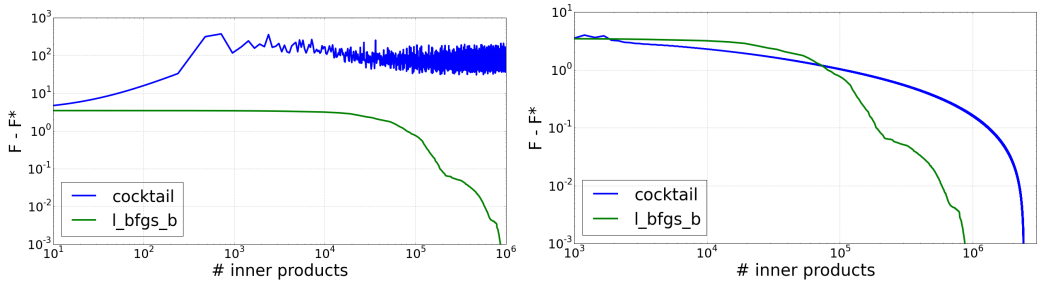


Figure II.1: Convergence of Cocktail and L-BFGS-B on Lymphoma dataset. *Top:* the starting point is $\theta^0 = \mathbf{0} \in \mathbb{R}^d$. *Bottom:* the starting point is $\theta^0 = \widehat{\theta}^{(l)}$ (solution to the same objective with a slightly larger λ). This illustrates the fact that Cocktail cannot minimize directly a single objective (with a fixed λ) and requires to compute the full path of solution to converge.

About the baselines specific to the Cox model. State-of-the-art algorithms to fit the elastic-net penalized Cox partial likelihood are `cocktail` by [YZ12] and `coxnet`, by [SFHT11]. Both algorithms are combining the ideas of coordinate descent and majoration-minimization. Full convergence results for these algorithms have not yet been established, although `Cocktail` has a coordinate-wise descent property.

These algorithms however need a good starting point (near the actual minimizer) to achieve convergence (this fact is due to a diagonal approximation of the Hessian matrix, see [HT90], Chapter 8.). They are therefore tuned to provide good path of solutions while varying by small steps the penalization parameter λ. Indeed in this case, this starting point is naturally set at the minimizer at the previous value of λ, when minimizing along a path but cannot be guessed outside of a path. We illustrate this fact on Figure II.1, where the convergence of `Cocktail` and L-BFGS-B algorithms are compared for two starting points θ_0.

Even when the starting point is set to the previous minimizer (second case in Figure II.1, `cocktail`'s convergence is slower than the one of L-BFGS-B. As a consequence, we decided that no fair comparison could be conducted with `cocktail` and `coxnet` algorithms.

Hybrid SVRG algorithm Since N_k exponentially increases, the 2SVRG's complexity is higher than SVRG's original complexity. However, the algorithm 2SVRG is very efficient during the first phases: we introduce an hybrid solver that begins with 2SVRG and switches to SVRG with mini-batchs (denoted SVRG-MB). Mini-batching simply consists in replacing single stochastic gradients ∇f_i by an average over a subset $\mathscr{B}$ of size n_{mb} uniformly selected at random. This is useful in our case, since we can use a computational trick (recurrence formula) to compute mini-batched gradients. In our experiments, we used $n_{\text{mb}} = 0.1n$ or $n_{\text{mb}} = 0.01n$, a constant step-size γ designed for each dataset, and switched from 2SVRG to SVRG-MB after $K_S = 5$ phases. We set $N_k = n^{k/(K_S+2)}$ so that N_k never exceeds n.

Baselines. We describe in this paragraph the algorithm that we put in competition in our experiments.

Algorithm 9 Hybrid SVRG (HSVRG)

Require: Number of phases before switching $K_S \geq 1$, total number of phases $K \geq K_S$, phase-length $m \geq 1$, step-size $\gamma > 0$, MCMC number of iterations per phase $(N_k)_{k=1}^K$, starting point $\theta^0 \in \mathbb{R}^d$
Initialize: $\tilde{\theta} \leftarrow \theta^0$ and compute $\nabla f_i(\tilde{\theta})$ for $i = 1, \dots, n$
for $k = 1$ **to** K_S **do**
  for $t = 0$ **to** $m - 1$ **do**
    Pick $i \sim \mathcal{U}[n]$
    $\widehat{\nabla} f_i(\theta^t) \leftarrow \text{ApproxMCMC}(i, \theta^t, N_k)$
    $d^t = \widehat{\nabla} f_i(\theta^t) - \nabla f_i(\tilde{\theta}) + \frac{1}{n} \sum_{j=1}^n \nabla f_j(\tilde{\theta})$
    $\omega^{t+1} \leftarrow \theta^t - \gamma d^t$
    $\theta^{t+1} \leftarrow \mathbf{prox}_{\gamma h}(\omega^{t+1})$
  end for
  Update $\tilde{\theta} \leftarrow \frac{1}{m} \sum_{t=1}^m \theta^t$, $\theta^0 \leftarrow \tilde{\theta}$, $\tilde{\theta}^k \leftarrow \tilde{\theta}$
  Compute $\nabla f_i(\tilde{\theta})$ for $i = 1, \dots, n$
end for
for $k = K_S + 1$ **to** K **do**
  for $t = 0$ **to** $m_{\text{mb}} - 1 = \lfloor (m-1)/n_{\text{mb}} \rfloor$ **do**
    Pick a set of random indices $\mathscr{B} \sim (\mathcal{U}[n])^{n_{\text{mb}}}$
    $d^t = \nabla f_{\mathscr{B}}(\theta^t) - \nabla f_{\mathscr{B}}(\tilde{\theta}) + \frac{1}{n} \sum_{j=1}^n \nabla f_j(\tilde{\theta})$
    $\omega^{t+1} \leftarrow \theta^t - \gamma d^t$
    $\theta^{t+1} \leftarrow \mathbf{prox}_{\gamma h}(\omega^{t+1})$
  end for
  Update $\tilde{\theta} \leftarrow \frac{1}{m_{\text{mb}}} \sum_{t=1}^{m_{\text{mb}}} \theta^t$, $\theta^0 \leftarrow \tilde{\theta}$, $\tilde{\theta}^k \leftarrow \tilde{\theta}$
end for
Return: $\tilde{\theta}^K$

FISTA This is accelerated proximal gradient from [BT09] with backtracking linesearch. Inner products necessary inside the backtracking are counted as well.

L-BFGS-B A state-of-the-art quasi-Newton solver which provides a usually strong baseline for many batch optimization algorithms, see [LN89]. We use the original implementation of the algorithm proposed in `python`'s `scipy.optimize` module. Non-differentiability of the ℓ_1-norm in the elastic-net penalization is dealt with the standard trick of reformulating the problem, using the fact that $|a| = a_+ + a_-$ for $a \in \mathbb{R}$.

HSVRG-UNIF-IMH This is Algorithm 9 where ApproxMCMC is done via Algorithm 7 with uniform proposal Q.

HSVRG-ADAP-IMH This is Algorithm 9 where ApproxMCMC is done via Algorithm 7 with adaptative proposal $Q = \pi_{\tilde{\theta}}^{\cdot}$.

HSVRG-AIS This is Algorithm 9 where ApproxMCMC is done via Algorithm 8, that is Adaptative Importance Sampling.

SVRG-MB Mini-Batch Prox-SVRG described in [Nit14], which can be seen as Algorithm 9 (see below) with $K_S = 0$. This is a *simply* stochastic algorithm, since there is no MCMC

approximation of the gradients ∇f_i. The question of mini-batch sizing is critical and is adressed in Section 10. We used $n_{\text{mb}} = 0.1n$ or $n_{\text{mb}} = 0.01n$ in our experiments.

The "simply stochastic" counterpart SVRG-MB is way slower than the corresponding doubly stochatic versions, since they rely on many computations of stochastic gradients ∇f_i, which are numerically costly, as explained above. The same settings are used throughout all experiments, some of them being tuned by hand: steps size for the variants of HSVRG are taken as $\gamma_t = \gamma_0 \in \{10^{-2}, 10^{-3}, 10^{-4}\}$ where γ_0 depends on the dataset, the phase length m is equal to the number n of failures of each datasets as suggested in [KLRT16]. As mentionned above, the doubly stochastic algorithms use different verions of ApproxMCMC.

Datasets We compare algorithms on the following datasets. The first three are standard benchmarks in survival analysis, the fourth one is a large simulated dataset where the number of observations n exceeds the number of features d. This differs from supervised gene expression data: such a large-scale setting happens for longitudinal clinical trials, medical adverse event monitoring and business data minings tasks.

- NKI70 contains survival data for 144 breast cancer patients, 5 clinical covariates and the expressions from 70 gene signatures, see [VDVHVV$^+$02].
- Luminal contains survival data for 277 patients with breast cancer who received the adjuvant tamoxifen, with 44,928 expressions measurements, see [LHKD$^+$07].
- Lymphoma contains 7399 gene expressions data for 240 lymphoma patients. The data was originally published in [AED$^+$00].
- We generated a Gaussian features matrix X with $n = 10,000$ observations and $d = 500$ predictors, with a Toeplitz covariance and correlation equal to 0.5. The failure times follow a Weibull distribution. See Section 9 for details on simulation in this model.

We compare in Figures II.2 and II.3 all algorithms for ridge penalization, namely $\alpha = 0$ and $\lambda = 1/\sqrt{n}$. Experiences with other values of α and λ are given in Section (including the Lasso penalization for instance).

Conclusions. The experiments first show that the solvers HSVRG-ADAP-IMH and HSVRG-AIS give better results than HSVRG-UNIF-IMF. However, the HSVRG solvers behave particularly well during the first phases where the gradients can be noisy - due to a small number of iterations of the MCMC - and still point a decent descent direction.

6 Conclusion

We have proposed a *doubly* stochastic gradient algorithm to extend SGD-like algorithms beyond the empirical risk minimization setting. The algorithm we proposed is the result of two different ideas: sampling from uniform distribution to avoid the computation of a large sum,

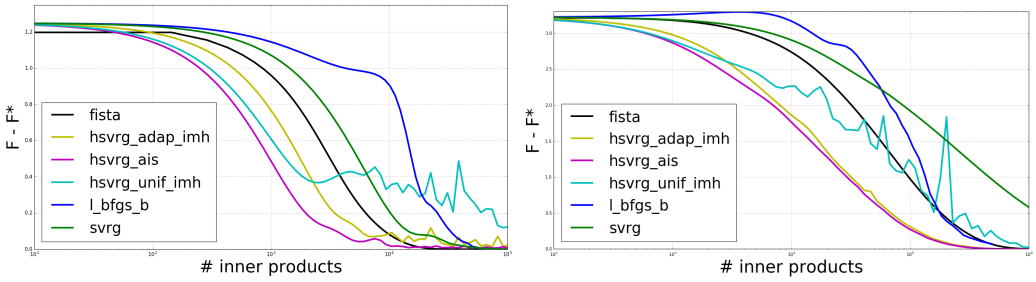


Figure II.2: Distance to optimum of all algorithms on NKI70 (left) and Lymphoma (right) with ridge penalization ($\alpha = 0$ and $\lambda = 1/\sqrt{n}$)

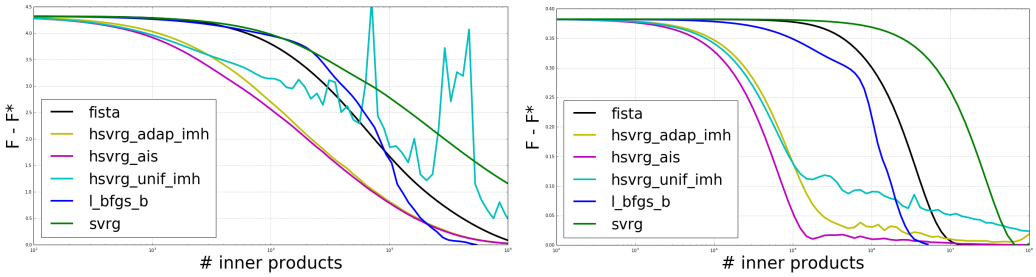


Figure II.3: Distance to optimum of all algorithms on Luminal (left) and on the simulated dataset (right) with ridge penalization ($\alpha = 0$ and $\lambda = 1/\sqrt{n}$)

and sampling using MCMC methods to avoid the computation of a more complicated expectation. We have also provided theoretical guarantees of convergence for both the convex and the strongly-convex setting.

This ***doubly*** stochastic gradient algorithm is very efficient during the early phases. The hybrid version of our algorithm, at the crossing of ***simply*** and ***doubly*** stochastic gradient algorithms, significantly outperforms state-of-the-art methods.

In a future work, we intend to extend our algorithm to Conditional Random Fields (CRF), where each subfunction's gradient takes the form

$$\nabla f_i(\theta) = \nabla(-\log(p(y_i|x_i,\theta)) = \sum_{Y \in \mathscr{Y}_i} \frac{e^{H(X_i,Y)^\top \theta}}{\sum_{Y' \in \mathscr{Y}_i} e^{H(X_i,Y')^\top \theta}} (H(X_i,Y) - H(X_i,Y_i)),$$

for a certain function H (see Page 2 in [SBA+15]). Notice that the Cox negative partial likelihood can be seen as a particular case of CRF by setting $X_i = [x_j]_{j \in R_i} \in \mathbb{R}^{d \times |R_i|}$, $Y_i = [\mathbb{1}_{j \in R_i}]_{j \in R_i} \in \{0,1\}^{|R_i|}$, $H(X,Y) = XY$ and $\mathscr{Y}_i = \{[\mathbb{1}_{j=k}]_{j \in R_i} : k \in R_i\}$.

7 Proofs

7.1 Proof of Proposition 1

We first prove Proposition 1 that ensures that Algorithm 7 provides the bounds of Assumption 2.

Proof. Since there exists $M > 0$ such that the proposal Q and the stationary distribution π satisfy $\pi(x) \leq MQ(x)$, for all x in the support of π, the Theorem 7.8 in [Rob04] states that the Algorithm 7 produces a geometrically ergodic Markov kernel P with ergodicity constants uniformly controlled:

$$\|P^k(x,\cdot) - \pi\|_{TV} \leq 2\left(1 - \frac{1}{M}\right)^k, \tag{8}$$

where P^k is the kernel of the k^{th} iteration of the algorithm and $\|\cdot\|_{TV}$ is the total variation norm. Since $\widehat{\nabla} f_{i_t}(\theta^{t-1})$ is computed as the mean of the iterates of the Markov chain, a simple computation enables us to bound the bias of the error and Proposition 12 from [FM$^+$03] gives the upper bound for the expected squared error:

$$\|\mathbb{E}_t \eta^t\| \leq \frac{C_1}{N_k} \text{ and } \mathbb{E}_t \|\eta^t\|^2 \leq \frac{C_2}{N_k} \tag{9}$$

where C_1 and C_2 are some finite constants, and N_k the number of iterations of the Markov chain. It can be shown that $C_1 = 2M$ and that C_2 is related to a constant from the Rosenthal's inequality. ■

7.2 Preliminaries to the proofs of Theorems 1 and 2

In what follows, the key lemmas for the proofs of Theorems 1 and 2 are stated and proved when not directly borrowed from previous articles.

Lemma 1. *For* $\Delta^t := \widehat{\nabla} f_{i_t}(\theta^{t-1}) - \nabla f_{i_t}(\tilde{\theta}) + \nabla f(\tilde{\theta}) - \nabla f(\theta^{t-1})$, *we have:*

$$\begin{aligned}\mathbb{E}_t \|\Delta^t\|^2 \leq 8L[&F(\theta^{t-1}) - F(\theta^*) \\ &+ F(\tilde{\theta}) - F(\theta^*)] + 3\mathbb{E}_t \|\eta^t\|^2.\end{aligned}$$

The proof of Lemma 1 uses Lemma 1 in [XZ14].

Lemma 2. *[JZ13, XZ14] Consider F satisfying Assumption 1. Then,*

$$\frac{1}{n}\sum_{i=1}^{n} \|\nabla f_i(\theta) - \nabla f_i(\theta^*)\|^2 \leq 2L[F(\theta) - F(\theta^*)]$$

Proof of Lemma 1. For the sake of simplicity, we now denote $d_i^t = \nabla f_i(\theta^{t-1}) - \nabla f_i(\tilde{\theta})$ and $d^t = \nabla f(\theta^{t-1}) - \nabla f(\tilde{\theta})$, so that one gets $\Delta^t = d_{i_t}^t - d^t + \eta^t$. Then, using the expectation introduced

in Section 4, we repeatedly use the identity $\mathbb{E}_t\|\xi\|^2 = \mathbb{E}_t\|\xi - \mathbb{E}_t\xi\|^2 + \|\mathbb{E}_t\xi\|^2$. First with $\xi = \Delta^t$ (since $\mathbb{E}_t d^t_{i_t} = d^t$, one gets $\mathbb{E}_t\xi = \mathbb{E}_t\eta^t$) :

$$\mathbb{E}_t\|\Delta^t\|^2 = \mathbb{E}_t\|d^t_{i_t} + \eta^t - \left(d^t + \mathbb{E}_t\eta^t\right)\|^2 + \|\mathbb{E}_t\eta^t\|^2$$

then, successively with $\xi = d^t_{i_t} + \eta^t$, $\xi = d^t + \eta^t$ and finally $\xi = \eta^t$:

$$\begin{aligned}
\mathbb{E}_t\|\Delta^t\|^2 &= \mathbb{E}_t\|d^t_{i_t} + \eta^t\|^2 + \|\mathbb{E}_t\eta^t\|^2 - \|d^t + \mathbb{E}_t\eta^t\|^2 \\
&= \mathbb{E}_t\|d^t_{i_t} + \eta^t\|^2 + \|\mathbb{E}_t\eta^t\|^2 \\
&\quad - \left(\mathbb{E}_t\|d^t + \eta^t\|^2 - \mathbb{E}_t\|\eta^t - \mathbb{E}_t\eta^t\|^2\right) \\
&= \mathbb{E}_t\|d^t_{i_t} + \eta^t\|^2 + \mathbb{E}_t\|\eta^t\|^2 - \mathbb{E}_t\|d^t + \eta^t\|^2.
\end{aligned}$$

Now we remark that $\mathbb{E}_t\|d^t + \eta^t\|^2 \geq 0$, and the identity $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$ gives the majoration

$$\mathbb{E}_t\|\Delta^t\|^2 \leq 2\mathbb{E}_t\|d^t_{i_t}\|^2 + 3\mathbb{E}_t\|\eta^t\|^2.$$

Now rewriting $d^t_{i_t} = \nabla f_{i_t}(\theta^{t-1}) - \nabla f_{i_t}(\theta^*) + \nabla f_{i_t}(\theta^*) - \nabla f_{i_t}(\tilde{\theta})$, the same identity leads to

$$\begin{aligned}
\mathbb{E}_t\|\Delta^t\|^2 \leq\ & 4\mathbb{E}_t\|\nabla f_{i_t}(\theta^{t-1}) - \nabla f_{i_t}(\theta^*)\|^2 \\
& + 4\mathbb{E}_t\|\nabla f_{i_t}(\tilde{\theta}) - \nabla f_{i_t}(\theta^*)\|^2 + 3\mathbb{E}_t\|\eta^t\|^2.
\end{aligned}$$

The desired result follows applying twice Lemma 2. ■

When F is μ-strongly convex, the next Lemma (Lemma 3 in [XZ14]) provides a key lower bound.

Lemma 3. *[XZ14] Consider $F = f + h$ satifying Assumption 1, where f is L_f-smooth, $L_f > 0$, f is μ_f-strongly convex, $\mu_f \geq 0$, h is μ_h-strongly convex, $\mu_h \geq 0$. For any $x, v \in \mathbb{R}^d$, we define $x^+ = \boldsymbol{prox}_{\gamma h}(x - \gamma v)$, $g = \frac{1}{\gamma}(x - x^+)$, where $\gamma \in (0, \frac{1}{L_f}]$. Then, for any $y \in \mathbb{R}^d$:*

$$\begin{aligned}
F(y) \geq\ & F(x^+) + g^\top(y - x) + \frac{\gamma}{2}\|g\|^2 + \frac{\mu_f}{2}\|y - x\|^2 \\
& + \frac{\mu_h}{2}\|y - x^+\|^2 + (v - \nabla f(x))^\top(x^+ - y).
\end{aligned} \tag{10}$$

Remark 3. *Note that in Lemma 3, one can freely choose μ_f and μ_h (in particular one can take $\mu_f = 0$ or $\mu_h = 0$), as long as $\mu_f + \mu_h = \mu$.*

The following Lemma comes from [AFM17] (Lemma 14):

Lemma 4. *[AFM17] Consider $F = f + h$ satifying Assumption 1, where f is L_f-smooth, and $T_\gamma : x \mapsto \boldsymbol{prox}_{\gamma h}[x - \gamma\nabla f(x)]$ with $\gamma \in (0, 2/L_f]$. Let $x, y \in \mathbb{R}^d$, we have:*

$$\|T_\gamma(x) - T_\gamma(y)\| \leq \|x - y\|$$

7.3 Proof of Theorem 1

Proof. The proof begins with the study of the distance $\|\theta^t - \theta^*\|^2$ *between* the phases $k-1$ and k. To ease the reading, when staying between these two phases, we write $\tilde{\theta}$ instead of $\tilde{\theta}^{k-1}$. Introducing $g^t = \frac{1}{\gamma}(\theta^{t-1} - \theta^t)$, we may write:

$$\begin{aligned}\|\theta^t - \theta^*\|^2 &= \|\theta^{t-1} - \gamma g^t - \theta^*\|^2 \\ &= \|\theta^{t-1} - \theta^*\|^2 - 2\gamma (g^t)^\top (\theta^{t-1} - \theta^*) \\ &\quad + \gamma^2 \|g^t\|^2.\end{aligned}$$

To upper bound the term $-2\gamma (g^t)^\top (\theta^{t-1} - \theta^*) + \gamma^2 \|g^t\|^2$, we apply the Lemma 3 with $x = \theta^{t-1}$, $x^+ = \theta^t$ and $y = \theta^*$. With again $\Delta^t = \widehat{\nabla} f_{i_t}(\theta^{t-1}) - \nabla f_{i_t}(\tilde{\theta}) + \nabla f(\tilde{\theta}) - \nabla f(\theta^{t-1})$, we obtain

$$\begin{aligned}&-(g^t)^\top (\theta^{t-1} - \theta^*) + \frac{\gamma}{2}\|g^t\|^2 \\ &\qquad \le F(\theta^*) - F(\theta^t) - \frac{\mu_f}{2}\|\theta^{t-1} - \theta^*\|^2 \\ &\qquad - \frac{\mu_h}{2}\|x^t - \theta^*\|^2 - (\Delta^t)^\top (\theta^t - \theta^*),\end{aligned}$$

and

$$\begin{aligned}\|\theta^t - \theta^*\|^2 \le \|\theta^{t-1} - \theta^*\|^2 &+ 2\gamma [F(\theta^*) - F(\theta^t)] \\ &- 2\gamma (\Delta^t)^\top (\theta^t - \theta^*).\end{aligned} \tag{11}$$

We now concentrate on the quantity $-2\gamma (\Delta^t)^\top (\theta^t - \theta^*)$. Introducing $\nu^t = \mathrm{prox}_{\gamma h}[\theta^{t-t1} - \gamma \nabla f(\theta^{t-1})] \in \mathscr{F}_{t-1}$ i.e. the vector obtained from θ^{t-1} after an exact proximal gradient descent step, we get

$$\begin{aligned}&-2\gamma (\Delta^t)^\top (\theta^t - \theta^*) \\ &\qquad = -2\gamma (\Delta^t)^\top (\theta^t - \nu^t) - 2\gamma (\Delta^t)^\top (\nu^t - \theta^*) \\ &\qquad \le 2\gamma \|\Delta^t\| \cdot \|\theta^t - \nu^t\| - 2\gamma (\Delta^t)^\top (\nu^t - \theta^*)\end{aligned}$$

where the inequality follows from the Cauchy-Schwartz inequality. Now the non-expansiveness property of proximal operators $\|\mathrm{prox}_{\gamma h}(x) - \mathrm{prox}_{\gamma h}(y)\| \le \|x - y\|$ leads to

$$\begin{aligned}&-2\gamma (\Delta^t)^\top (\theta^t - \theta^*) \\ &\quad \le 2\gamma \|\Delta^t\| \cdot \|\{\theta^{t-1} - \gamma(\Delta^t + \nabla f(\theta^{t-1}))\} \\ &\quad - \{\theta^{t-1} - \gamma \nabla f(\theta^{t-1})\}\| - 2\gamma (\Delta^t)^\top (\nu^t - \theta^*) \\ &\quad \le 2\gamma^2 \|\Delta^t\|^2 - 2\gamma (\Delta^t)^\top (\nu^t - \theta^*).\end{aligned}$$

Reminding that $\nu^t \in \mathscr{F}_{t-1}$, we derive:

$$\begin{aligned}&-2\gamma \mathbb{E}_t (\Delta^t)^\top (\theta^t - \theta^*) \\ &\quad \le 2\gamma^2 \mathbb{E}_t \|\Delta^t\|^2 - 2\gamma (\mathbb{E}_t \Delta^t)^\top (\nu^t - \theta^*) \\ &\quad \le 2\gamma^2 \mathbb{E}_t \|\Delta^t\|^2 + 2\gamma \|\mathbb{E}_t \Delta^t\| \cdot \|\nu^t - \theta^*\|,\end{aligned}$$

the last inequality comes from the Cauchy-Schwartz inequality. Since θ^* is the minimum of $F = f + h$, it satisfies $\theta^* = \text{prox}_{\gamma h}[\theta^* - \gamma \nabla f(\theta^*)]$. Thus, the Lemma 4 and the Assumption 1 on the sequence (θ^t) give us $\|\nu^t - \theta^*\| \leq \|\theta^{t-1} - \theta^*\| \leq B$. We also remark that $\mathbb{E}_t \Delta^t = \mathbb{E}_t \eta^t$. For all t between phases $k-1$ and k, we finally apply Lemma 1 to obtain:

$$\begin{aligned} -2\gamma \mathbb{E}_t (\Delta^t)^\top (\theta^t - \theta^*) & \\ \leq 16\gamma^2 L[F(\theta^{t-1}) - F(\theta^*) + F(\tilde{\theta}) - F(\theta^*)] & \\ + 6\gamma^2 \mathbb{E}_t \|\eta^t\|^2 + 2\gamma B \|\mathbb{E}_t \eta^t\|. & \end{aligned} \tag{12}$$

Taking the expectation $\mathbb{E}_t$ on inequation (11) and combining with previous inequality leads to

$$\begin{aligned} \mathbb{E}_t \|\theta^t - \theta^*\|^2 \leq \|\theta^{t-1} - \theta^*\|^2 + 2\gamma[F(\theta^*) - F(\theta^t)] & \\ + 16\gamma^2 L[F(\theta^{t-1}) - F(\theta^*) + F(\tilde{\theta}) - F(\theta^*)] & \\ + 6\gamma^2 \mathbb{E}_t \|\eta^t\|^2 + 2\gamma B \|\mathbb{E}_t \eta^t\|. & \end{aligned}$$

With the notation of Algorithm 6, $\tilde{\theta} = \tilde{\theta}^{k-1} = \theta^0$. Now, applying iteratively the previous inequality over $t = 1, 2, \ldots, m$ and taking the expectation $\mathbb{E}$ over $i_1, \theta^1, i_2, \theta^2, \ldots, i_m, \theta^m$, we obtain:

$$\begin{aligned} \mathbb{E}\|\theta^m - \theta^*\|^2 + 2\gamma[\mathbb{E}F(\theta^m) - F(\theta^*)] & \\ + 2\gamma(1 - 8L\gamma) \sum_{t=1}^{m-1} [\mathbb{E}F(\theta^t) - F(\theta^*)] & \\ \leq \|\theta^0 - \theta^*\|^2 + 16L\gamma^2[F(\theta^0) - F(\theta^*) + m(F(\tilde{\theta}) & \\ - F(\theta^*))] + 6\gamma^2 \sum_{t=1}^{m} \mathbb{E}\|\eta^t\|^2 + 2\gamma B \sum_{t=1}^{m} \mathbb{E}\|\mathbb{E}_t \eta^t\|. & \end{aligned}$$

Now, by convexity of F and the definition $\tilde{\theta}^k = \frac{1}{m} \sum_{t=1}^m \theta^t$, we may write $F(\tilde{\theta}^k) \leq \frac{1}{m} \sum_{t=1}^m F(\theta^t)$. Noticing that $2\gamma(1 - 8L\gamma) < 2\gamma$ leads to

$$\begin{aligned} 2\gamma(1 - 8L\gamma) m[\mathbb{E}F(\tilde{\theta}^k) - F(\theta^*)] & \\ \leq \|\tilde{\theta} - \theta^*\|^2 + 16L\gamma^2(m+1)[F(\tilde{\theta}) - F(\theta^*)] & \\ + 6\gamma^2 \sum_{t=1}^{m} \mathbb{E}\|\eta^t\|^2 + 2\gamma B \sum_{t=1}^{m} \|\mathbb{E}\eta^t\|. & \end{aligned}$$

Under the Assumption 2, we have

$$\begin{aligned} 6\gamma^2 \sum_{t=1}^{m} \mathbb{E}\|\eta^t\|^2 + 2\gamma B \sum_{t=1}^{m} \|\mathbb{E}\eta^t\| & \\ \leq (6\gamma^2 C_2 + 2\gamma B C_1) \frac{m}{N_k} & \end{aligned}$$

whereas the μ-strong convexity of F implies $\|\tilde{\theta}^{k-1} - \theta^*\|^2 \leq \frac{2}{\mu}[F(\tilde{\theta}^{k-1}) - F(\theta^*)]$. This leads to

$$\mathbb{E}F(\tilde{\theta}^k) - F(\theta^*) \leq \rho \left(\mathbb{E}F(\tilde{\theta}^{k-1}) - F(\theta^*) \right) + \frac{D}{N_k}$$

for D and ρ as defined in the theorem. Applying the last inequality recursively leads to the result. ■

7.4 Proof of Theorem 2

Proof. As at the begining of the proof of Theorem 1, we consider that we stand between phase $k-1$ and phase k of Algorithm 6 and consequently $\theta^0 = \tilde{\theta}^{k-1}$. We use the same arguments until (11), with the difference that, in this non-strongly convex case, we have $\mu_f = \mu_h = 0$. We obtain for all t between phases 1 and m

$$F(\theta^t) - F(\theta^*) \leq \frac{1}{2\gamma}(\|\theta^{t-1} - \theta^*\|^2 - \|\theta^t - \theta^*\|^2) - (\theta^t - \theta^*)^\top \Delta^t.$$

Summing over $t = 1, \dots, \tau$ (for $\tau \leq m$) leads to

$$\sum_{t=1}^{\tau} [F(\theta^t) - F(\theta^*)] \leq \frac{1}{2\gamma}(\sum_{t=0}^{\tau-1} \|\theta^t - \theta^*\|^2 - \sum_{t=1}^{\tau} \|\theta^t - \theta^*\|^2) - \sum_{t=1}^{\tau} (\theta^t - \theta^*)^\top \Delta^t. \tag{13}$$

We now use Equation (13) (with $\tau = m$) and the convexity of $\|\cdot\|^2$ with $\tilde{\theta}^k = \frac{1}{m}\sum_{t=1}^{m} \theta^t$ to write

$$\sum_{t=1}^{m} [F(\theta^t) - F(\theta^*)] \leq \frac{1}{2\gamma}\left(\sum_{t=0}^{m-1} \|\theta^t - \theta^*\|^2 - m\|\tilde{\theta}^k - \theta^*\|^2\right) - \sum_{t=1}^{m} (\theta^t - \theta^*)^\top \Delta^t. \tag{14}$$

Starting from Equation (13) again but now summing over $l = 1, \dots, t$, we get

$$\begin{aligned} \frac{1}{2\gamma}(\|\theta^0 - \theta^*\|^2 - \|\theta^t - \theta^*\|^2) - \sum_{l=1}^{t} (\theta^l - \theta^*)^\top \Delta^l &\geq \sum_{l=1}^{t} [F(\theta^l) - F(\theta^*)] \\ &\geq 0, \end{aligned} \tag{15}$$

where the last inequality follows from the definition of θ^*. In (14), we now substitute $\|\theta^t - \theta^*\|^2$ by the upper bound derived from (15) to write (noticing that $\theta^0 = \tilde{\theta}^{k-1}$):

$$\begin{aligned}
&\sum_{t=1}^{m} [F(\theta^t) - F(\theta^*)] \\
&\quad \leq \frac{m}{2\gamma}(\|\tilde{\theta}^{k-1} - \theta^*\|^2 - \|\tilde{\theta}^k - \theta^*\|^2) \\
&\qquad\qquad - \sum_{t=1}^{m-1} \sum_{l=1}^{t} (\theta^l - \theta^*)^\top \Delta^l - \sum_{t=1}^{m} (\theta^t - \theta^*)^\top \Delta^t \\
&\quad \leq \frac{m}{2\gamma}(\|\tilde{\theta}^{k-1} - \theta^*\|^2 - \|\tilde{\theta}^k - \theta^*\|^2) \\
&\qquad\qquad - \sum_{t=1}^{m} (m+1-t)(\theta^t - \theta^*)^\top \Delta^t.
\end{aligned}$$

As in the proof of Theorem 1 (see Equation (12)), each term $-\mathbb{E}_t(\theta^t - \theta^*)^\top \Delta^t$ is upper bounded by $8\gamma L[F(\theta^{t-1}) - F(\theta^*) + F(\tilde{\theta}^{k-1}) - F(\theta^*)] + 3\gamma \mathbb{E}_t||\eta^t||^2 + B||\mathbb{E}_t \eta^t||$. Now with $m + 1 - t \leq m$ and Assumption 2, we obtain:

$$\begin{aligned}
&\frac{1}{m} \sum_{t=1}^{m} \mathbb{E}[F(\theta^t) - F(\theta^*)] \\
&\quad \leq \frac{1}{2\gamma}(\|\tilde{\theta}^{k-1} - \theta^*\|^2 - \mathbb{E}\|\tilde{\theta}^k - \theta^*\|^2) \\
&\quad + 8L\gamma \Big\{ \sum_{t=1}^{m} [\mathbb{E}F(\theta^{t-1}) - F(\theta^*)] + F(\theta^*) - \mathbb{E}[F(\theta^m)] \\
&\quad + (m+1)[\mathbb{E}[F(\tilde{\theta}^{k-1})] - F(\theta^*)] \Big\} + m \frac{3\gamma C_2 + BC_1}{N_k}.
\end{aligned}$$

By definition of γ, we have $8Lm\gamma < 1$, and we can use the convexity of F to lower bound the left hand side. With the inequality $\mathbb{E}[F(\theta^m)] - F(\theta^*) \geq 0$, one has:

$$\begin{aligned}
&(1 - 8L\gamma m) \left[\mathbb{E}[F(\tilde{\theta}^k)] - F(\theta^*) \right] \\
&\quad \leq \frac{1}{2\gamma} \left(\|\tilde{\theta}^{k-1} - \theta^*\|^2 - \mathbb{E}\|\tilde{\theta}^k - \theta^*\|^2 \right) \\
&\quad + 8L\gamma(m+1) \left[\mathbb{E}[F(\tilde{\theta}^{k-1})] - F(\theta^*) \right] \\
&\quad + m \frac{3\gamma C_2 + BC_1}{N_k}
\end{aligned}$$

We now take the expectation $\mathbb{E}$ on all iterates of the algorithm i.e. on the iterates $i_1, \theta^1, i_2, \theta^2, \ldots, i_m, \theta^m$ from the first phase. Introduce the notations $A^k = \mathbb{E}[F(\tilde{\theta}^k)] - F(\theta^*)$ and $a = (8L\gamma(m+1))/(1 -$

$8Lm\gamma) < 1$, last inequality leads to:

$$\begin{aligned} & A^k - aA^{k-1} \\ & \quad \leq \frac{1}{2\gamma(1-8Lm\gamma)} \left(\mathbb{E}\|\tilde{\theta}^{k-1} - \theta^*\|^2 - \mathbb{E}\|\tilde{\theta}^k - \theta^*\|^2 \right) \\ & \quad + \frac{D}{N_k}, \end{aligned}$$

where D is defined in the theorem. Summing over the phases $k = 1, 2, \ldots, K+1$ and lower bounding A^{K+1} with 0, we obtain:

$$\begin{aligned} & (1-a) \sum_{k=1}^{K} A^k \\ & \leq aA^0 + \frac{1}{2\gamma(1-8Lm\gamma)} \|\tilde{\theta}^0 - \theta^*\|^2 + \sum_{k=1}^{K+1} \frac{D}{N_k} \end{aligned}$$

The last argument is the use of the convexity of F. Remark the explicit forms of the constants in the theorem:

$$D_1 = \frac{a}{1-a} A^0 + \frac{1}{1-a} \frac{\|\tilde{\theta}^0 - \theta^*\|^2}{2\gamma(1-8Lm\gamma)}$$

and $D_2 = \frac{D}{1-a}$. ■

8 Supplementary experiments

We have tested all algorithms with other settings for the penalization. Namely, we considered:

High lasso. We take $\alpha = 1$ and $\lambda = 1/\sqrt{n}$ and illustrate our results in Figure II.5.

Low lasso. We take $\alpha = 1$ and $\lambda = 1/n$ and illustrate our results in Figure II.4.

High ridge. We take $\alpha = 0$ and $\lambda = 1/\sqrt{n}$ and illustrate our results in Figures II.2 and Figures II.3.

Low ridge. We take $\alpha = 0$ and $\lambda = 1/n$ and illustrate our results in Figure II.6.

9 Simulation of data

With Cox model, the hazard ratio for the failure time T_i of the i^{th} patient takes the form:

$$\lambda_i(t) = \lambda_0(t) \exp(x_i^\top \theta),$$

where $\lambda_0(t)$ is a baseline hazard ratio, and $x_i \in \mathbb{R}^d$ the covariates of the i^{th} patient.
We first simulate the feature matrix $X \in \mathbb{R}^{n \times d}$ as a Gaussian vector with a Toepliz covariance, where the correlation between features j and j' is equal to $\rho^{|j-j'|}$, for some $\rho \in (0, 1)$.

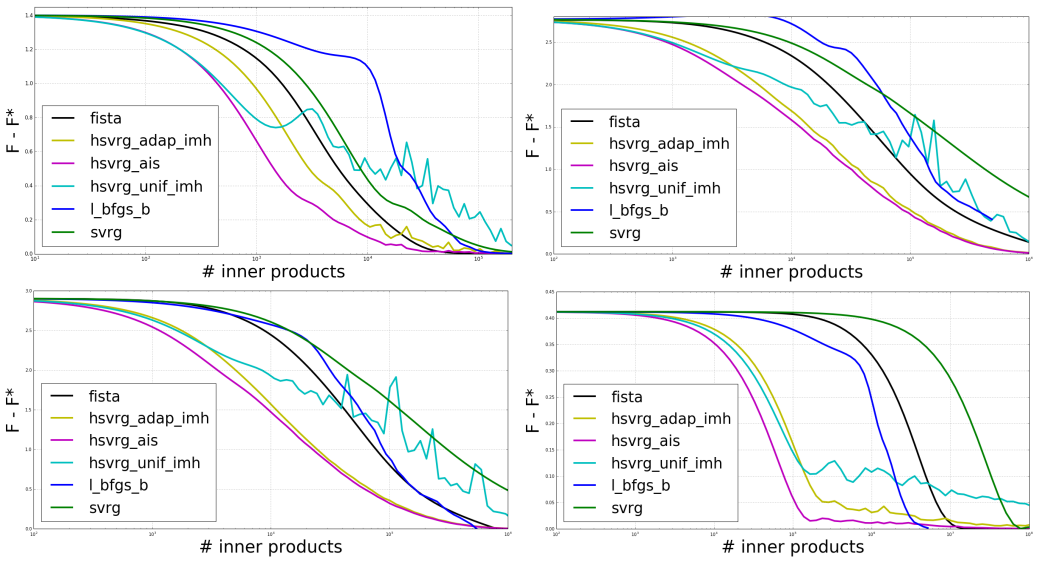


Figure II.4: Distance to optimum of all algorithms on NKI70, Lymphoma, Luminal and on the simulated dataset (respectively from top to bottom) for **Low-ridge** penalization

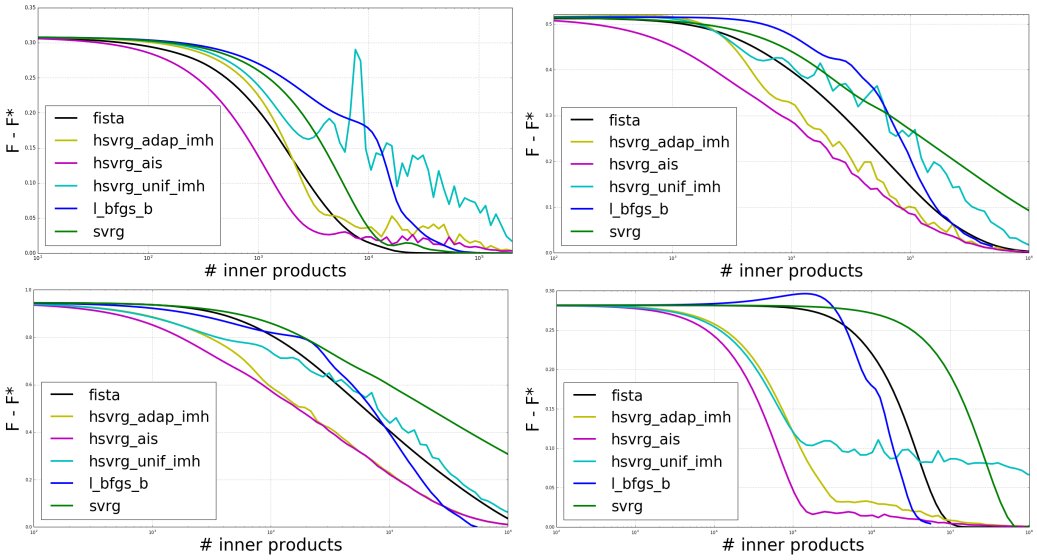


Figure II.5: Distance to optimum of all algorithms on NKI70, Lymphoma, Luminal and on the simulated dataset (respectively from top to bottom) for **High-lasso** penalization

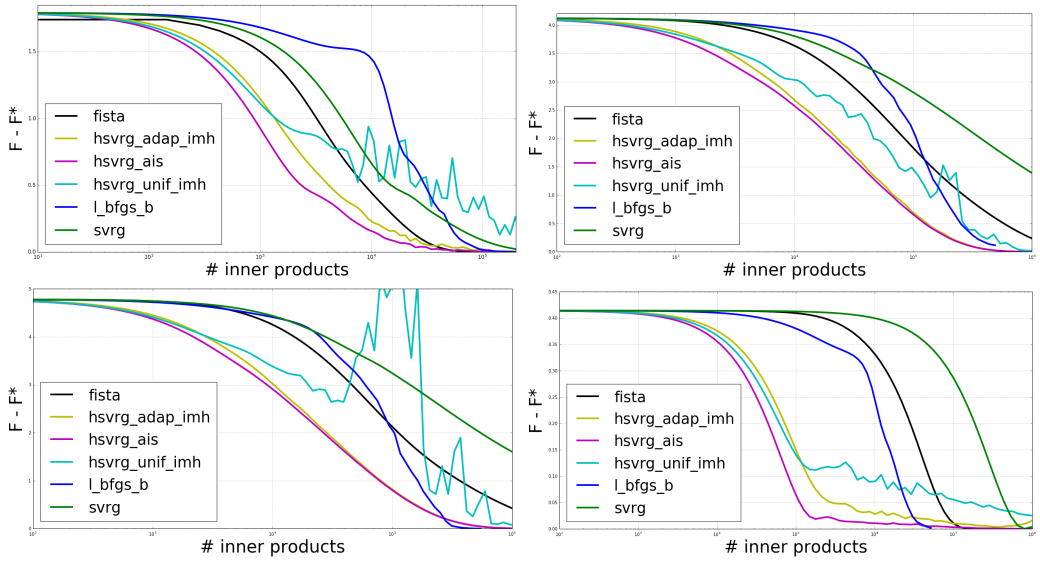


Figure II.6: Distance to optimum of all algorithms on NKI70, Lymphoma, Luminal and on the simulated dataset (respectively from top to bottom) for **Low-lasso** penalization

We want now to simulate the observed time y_i that corresponds to x_i. We denote the cumulative hazard function $\Lambda(t) = \int_0^t \lambda(s)ds$. Using the definition $\lambda(t) = \frac{f(t)}{1-F(t)}$, we know that $\Lambda(t) = -\log(1-F(t))$, where f is the p.d.f. and F is the c.d.f. of T.
It is easily seen that $\Lambda(T)$ has distribution Exp(1) (Exponential with intensity equal to 1): since Λ is an increasing function, we have

$$\begin{aligned}\mathbb{P}(\Lambda(T) \geq t) = \mathbb{P}(T \geq \Lambda^{-1}(t)) &= \int_{\Lambda^{-1}(t)}^{\infty} f(s)ds \\ &= 1 - F(\Lambda^{-1}(t)) \\ &= \exp(-\Lambda(\Lambda^{-1}(t))) \\ &= \exp(-t),\end{aligned}$$

so that simulating failure times is simply achieved by using $T_i = \Lambda^{-1}(E_i)$ where $E_i \sim \text{Exp}(1)$. To compute Λ, we should have a parametric form for λ_0. We assume that T follows the Weibull distribution $\mathcal{W}(1,\nu)$ (when $x_i = 0$). This choice is motivated by the following facts:

- Its cumulative hazard function is easy to invert. Indeed the hazard ratio is given by $\lambda_0(t) = \frac{\nu t^{\nu-1}e^{-t^\nu}}{1-(1-e^{-t^\nu})} = \nu t^{\nu-1}$, so that $\Lambda^{-1}(y) = \left(\frac{y}{\exp(x_i^\top \theta)}\right)^{1/\nu}$.
- It enables two different trends - increasing or decreasing – for the baseline hazard ratio that correspond to two typical behaviours in the medical field.
  - decreasing: after taking a treatment, time before a side-effect's appearence

– increasing: no memory process and patient's health is worsening

This method enables us to simulate n failures times $T_1, T_2, \dots, T_n$.
Then, we simulate $C_1, C_2, \dots, C_n$ with exponential distribution. This finally gives us a set of observed times $(y_i)_{i=1}^n = (T_i \wedge C_i)_{i=1}^n$ and a set of censoring indicators $(\delta_i)_{i=1}^n = (\mathbb{1}_{\{T_i \wedge C_i\}})_{i=1}^n$.

10 Mini-batch sizing

The mini-batch sizing question is essential since it is a natural trade-off between computing time and precision. We know that computing $\nabla f_i(\theta)$ needs the computation of $|R_i| \in \{1, \dots, n_{\text{pat}}\}$ inner products. One proves easily that computing a mini-batch $(1/n_{\text{mb}})\nabla f_{\mathscr{B}}(\theta)$ - where $\mathscr{B}$ is the set of n_{mb} index randomly picked - only needs $\max_{i \in \mathscr{B}} |R_i|$ inner products. A simple probability exercise gives us a key insight about the mini-batch size.
Let's assume that censoring is ***uniform*** over the set $\{1, 2, \dots, n_{\text{pat}}\}$ meaning that $|R_i| = ci$ with $c > 1$. Then, we denote $u_1, u_2, \dots, u_{n_{\text{mb}}} \sim \mathscr{U}[n]$ the indices independently sampled to compute the mini-batch i.e. $\mathscr{B} = \{u_i\}_{i=1}^{n_{\text{mb}}}$. Now we study the c.d.f. of $\max_{1 \le i \le n_{\text{mb}}} u_i$: for $k \in \{1, 2, \dots, n\}$,

$$\mathbb{P}\left(\max_{1 \le i \le n_{\text{mb}}} u_i \le k\right) = \prod_{i=1}^{n_{\text{mb}}} \mathbb{P}(u_i \le k) = \left(\frac{\lfloor k \rfloor}{n}\right)^{n_{\text{mb}}},$$
$$\mathbb{P}\left(\max_{i \in \mathscr{B}} |R_i| \le ck\right) = \left(\frac{\lfloor k \rfloor}{n}\right)^{n_{\text{mb}}},$$
$$\mathbb{P}\left(\max_{i \in \mathscr{B}} |R_i| \ge a\right) = 1 - \left(\frac{\lfloor a/c \rfloor}{n}\right)^{n_{\text{mb}}}, \text{ for } a < n_{\text{pat}}$$

The third equation leads us to consider $1 \ll n_{\text{mb}} \ll n$ to prevent both $\max_{i \in \mathscr{B}} |R_i|$ and $|\mathscr{B}|$ from being too large. This is why we used $n_{\text{mb}} = 0.1n$ or $n_{\text{mb}} = 0.01n$, depending of the size n of the dataset.

Part II

Uncover Hawkes causality without parametrization

CHAPTER III

Generalized Method of Moments approach

Abstract

We design a new nonparametric method that allows one to estimate the matrix of integrated kernels of a multivariate Hawkes process. This matrix not only encodes the mutual influences of each node of the process, but also disentangles the causality relationships between them. Our approach is the first that leads to an estimation of this matrix *without any parametric modeling and estimation of the kernels themselves.* As a consequence, it can give an estimation of causality relationships between nodes (or users), based on their activity timestamps (on a social network for instance), without knowing or estimating the shape of the activities lifetime. For that purpose, we introduce a moment matching method that fits the second-order and the third-order integrated cumulants of the process. A theoretical analysis allows us to prove that this new estimation technique is consistent. Moreover, we show, on numerical experiments, that our approach is indeed very robust with respect to the shape of the kernels and gives appealing results on the MemeTracker database and on financial order book data.

Keywords. Hawkes Process, Causality Inference, Cumulants, Generalized Method of Moments

1 Introduction

In many applications, one needs to deal with data containing a very large number of irregular timestamped events that are recorded in continuous time. These events can reflect, for instance, the activity of users on a social network, see [SAD+16], the high-frequency variations of signals in finance, see [BMM15], the earthquakes and aftershocks in geophysics, see [Oga98], the crime activity, see [MSB+11] or the position of genes in genomics, see [RBS10]. The succession of the precise timestamps carries a great deal of information about the dynamics of the underlying systems. In this context, multidimensional counting processes based models play a paramount role. Within this framework, an important task is to recover the mutual influence of the nodes (i.e., the different components of the counting process), by leveraging on their timestamp patterns, see, for instance, [BM16, LV14, LM11, ZZS13, GRLS13, FWR+15, XFZ16].

Consider a set of nodes $I = \{1, \dots, d\}$. For each $i \in I$, we observe a set Z^i of *events*, where each $\tau \in Z^i$ labels the occurrence time of an event related to the activity of i. The events of all nodes can be represented as a vector of counting processes $\boldsymbol{N}_t = [N_t^1 \cdots N_t^d]^\top$, where N_t^i counts the number of events of node i until time $t \in \mathbb{R}^+$, namely $N_t^i = \sum_{\tau \in Z^i} \mathbb{1}_{\{\tau \leq t\}}$. The vector of stochastic intensities $\boldsymbol{\lambda}_t = [\lambda_t^1 \cdots \lambda_t^d]^\top$ associated with the multivariate counting process $\boldsymbol{N}_t$ is defined as

$$\lambda_t^i = \lim_{dt \to 0} \frac{\mathbb{P}(N_{t+dt}^i - N_t^i = 1 | \mathscr{F}_t)}{dt}$$

for $i \in I$, where the filtration $\mathscr{F}_t$ encodes the information available up to time t. The coordinate λ_t^i gives the expected instantaneous rate of event occurrence at time t for node i. The vector $\boldsymbol{\lambda}_t$ characterizes the distribution of $\boldsymbol{N}_t$, see [DVJ07], and patterns in the events time-series can be captured by structuring these intensities.

The Hawkes process introduced in [Haw71a] corresponds to an autoregressive structure of the intensities in order to capture self-excitation and cross-excitation of nodes, which is a phenomenon typically observed, for instance, in social networks, see for instance [CS08]. Namely, $\boldsymbol{N}_t$ is called a *Hawkes point process* if the stochastic intensities can be written as

$$\lambda_t^i = \mu^i + \sum_{j=1}^{d} \int_0^t \phi^{ij}(t - t') dN_{t'}^j,$$

where $\mu^i \in \mathbb{R}^+$ is an exogenous intensity and ϕ^{ij} are positive, integrable and causal (with support in $\mathbb{R}_+$) functions called *kernels* encoding the impact of an action by node j on the activity of node i. Note that when all kernels are zero, the process is a simple homogeneous multivariate Poisson process.

Most of the litterature uses a parametric approach for estimating the kernels. With no doubt, the most popular parametrization form is the exponential kernel $\phi^{ij}(t) = \alpha_{ij} \beta_{ij} e^{-\beta_{ij} t}$ because it definitely simplifies the inference algorithm (e.g., the complexity needed for computing the likelihood is much smaller). When d is large, in order to reduce the number of parameters, some authors choose to arbitrarily share the kernel shapes across the different nodes. Thus, for instance, in [YZ13, ZZS13, FWR+15], they choose $\phi^{ij}(t) = \alpha_{ij} h(t)$ with $\alpha_{ij} \in \mathbb{R}^+$ quantifies the intensity of the influence of j on i and $h(t)$ a (normalized) function that characterizes the time-profile of this influence and that is *shared* by all couples of nodes (i, j) (most often, it is chosen to be either exponential $h(t) = \beta e^{-\beta t}$ or power law $h(t) = \beta t^{-(\beta+1)}$). Both approaches are, most of the time, highly non-realistic. On the one hand there is a priori no reason for assuming that the time-profile of the influence of a node j on a node i does not depend on the pair (i, j). On the other hand, assuming an exponential shape or a power law shape for a kernel arbitrarily imposes an event impact that is always instantly maximal and that can only decrease with time, while in practice, there may exist a latency between an event and its maximal impact.

In order to have more flexibility on the shape of the kernels, nonparametric estimation can be considered. Expectation-Maximization algorithms can be found in [LM11] (for $d = 1$) or in [ZZS13] ($d > 1$). An alternative method is proposed in [BM16] where the nonparametric

estimation is formulated as a numerical solving of a Wiener-Hopf equation. Another nonparametric strategy considers a decomposition of kernels on a dictionary of function $h_1, \dots, h_K$, namely $\phi^{ij}(t) = \sum_{k=1}^{K} a_k^{ij} h_k(t)$, where the coefficients a_k^{ij} are estimated, see [HRBR+15, LV14] and [XFZ16], where group-lasso is used to induce a sparsity pattern on the coefficients a_k^{ij} that is shared across $k = 1, \dots, K$.

Such methods are heavy when d is large, since they rely on likelihood maximization or least squares minimization within an over-parametrized space in order to gain flexibility on the shape of the kernels. This is problematic, since the original motivation for the use of Hawkes processes is to estimate the influence and causality of nodes, the knowledge of the full parametrization of the model being of little interest for causality purpose.

Our paper solves this problem with a different and more direct approach. Instead of trying to estimate the kernels ϕ^{ij}, we focus on the direct estimation of their *integrals*. Namely, we want to estimate the matrix $\boldsymbol{G} = [g^{ij}]$ where

$$g^{ij} = \int_0^{+\infty} \phi^{ij}(u)\, du \geq 0 \quad \text{for} \quad 1 \leq i, j \leq d. \tag{1}$$

As it can be seen from the cluster representation of Hawkes processes ([HO74]), this integral represents the mean total number of events of type i directly triggered by an event of type j, and then encodes a notion of *causality*. Actually, as detailed below (see Section 2.1), such integral can be related to the Granger causality ([Gra69]).

The main idea of the method we developed in this paper is to estimate the matrix $\boldsymbol{G}$ directly using a matching cumulants (or moments) method. Apart from the mean, we shall use second and third-order cumulants which correspond respectively to centered second and third-order moments. We first compute an estimation $\widehat{\boldsymbol{M}}$ of these centered moments $M(\boldsymbol{G})$ (they are uniquely defined by $\boldsymbol{G}$). Then, we look for a matrix $\widehat{\boldsymbol{G}}$ that minimizes the L^2 error $\|M(\widehat{\boldsymbol{G}}) - \widehat{\boldsymbol{M}}\|^2$. Thus the integral matrix $\widehat{\boldsymbol{G}}$ is directly estimated without making hardly any assumptions on the shape the involved kernels. As it will be shown, this approach turns out to be particularly robust to the kernel shapes, which is not the case of all previous Hawkes-based approaches that aim causality recovery. We call this method NPHC (Non Parametric Hawkes Cumulant), since our approach is of nonparametric nature. We provide a theoretical analysis that proves the consistency of the NPHC estimator. Our proof is based on ideas from the theory of Generalized Method of Moments (GMM) but requires an original technical trick since our setting strongly departs from the standard parametric statistics with i.i.d observations. Note that moment and cumulant matching techniques proved particularly powerful for latent topic models, in particular Latent Dirichlet Allocation, see [PBLJ15]. A small set of previous works, namely [DFZ14, ASCDL10], already used method of moments with Hawkes processes, but only in a parametric setting. Our work is the first to consider such an approach for a nonparametric counting processes framework.

The paper is organized as follows: in Section 2, we provide the background on the integrated kernels and the integrated cumulants of the Hawkes process. We then introduce

the method, investigate its complexity and explain the consistency result we prove. In Section 3, we estimate the matrix of Hawkes kernels' integrals for various simulated datasets and for real datasets, namely the MemeTracker database and financial order book data. We then provide in Section 4 the technical details skipped in the previous parts and the proof of our consistency result. Section 5 contains concluding remarks.

2 NPHC: The Non Parametric Hawkes Cumulant method

In this Section, we provide the background on integrals of Hawkes kernels and integrals of Hawkes cumulants. We then explain how the NPHC method enables estimating $\boldsymbol{G}$.

2.1 Branching structure and Granger causality

From the definition of Hawkes process as a Poisson cluster process, see [JHR15] or [HO74], g^{ij} can be simply interpreted as the average total number of events of node i whose *direct* ancestor is a given event of node j (by direct we mean that interactions mediated by any other intermediate event are not counted). In that respect, $\boldsymbol{G}$ not only describes the mutual influences between nodes, but it also quantifies their *direct causal* relationships. Namely, introducing the counting function $N_t^{i\leftarrow j}$ that counts the number of events of i whose direct ancestor is an event of j, we know from [BMM15] that

$$\mathbb{E}[dN_t^{i\leftarrow j}] = g^{ij}\mathbb{E}[dN_t^j] = g^{ij}\Lambda^j dt, \tag{2}$$

where we introduced Λ^i as the intensity expectation, namely satisfying $\mathbb{E}[dN_t^i] = \Lambda^i dt$. Note that Λ^i does not depend on time by stationarity of $\boldsymbol{N}_t$, which is known to hold under the *stability condition* $\|\boldsymbol{G}\| < 1$, where $\|\boldsymbol{G}\|$ stands for the spectral norm of $\boldsymbol{G}$. In particular, this condition implies the non-singularity of $\boldsymbol{I_d} - \boldsymbol{G}$.

Since the question of a *real causality* is too complex in general, most econometricians agreed on the simpler definition of Granger causality [Gra69]. Its mathematical formulation is a statistical hypothesis test: X causes Y *in the sense of Granger causality* if forecasting future values of Y is more successful while taking X past values into account. In [EDD17], it is shown that for $\boldsymbol{N}_t$ a multivariate Hawkes process, N_t^j does not Granger-cause N_t^i w.r.t $\boldsymbol{N}_t$ if and only if $\phi^{ij}(u) = 0$ for $u \in \mathbb{R}_+$. Since the kernels take positive values, the latter condition is equivalent to $\int_0^\infty \phi^{ij}(u)du = 0$. In the following, we'll refer to *learning the kernels' integrals* as *uncovering causality* since each integral encodes the notion of Granger causality, and is also linked to the number of events directly caused from a node to another node, as described above at Eq. (2).

2.2 Integrated cumulants of the Hawkes process

A general formula for the integral of the cumulants of a multivariate Hawkes process is provided in [JHR15]. As explained below, for the purpose of our method, we only need to consider cumulants up to the third order. Given $1 \le i, j, k \le d$, the first three integrated

cumulants of the Hawkes process can be defined as follows thanks to stationarity:

$$\Lambda^i dt = \mathbb{E}(dN_t^i) \tag{3}$$

$$C^{ij} dt = \int_{\tau \in \mathbb{R}} \Big(\mathbb{E}(dN_t^i dN_{t+\tau}^j) - \mathbb{E}(dN_t^i)\mathbb{E}(dN_{t+\tau}^j) \Big) \tag{4}$$

$$\begin{aligned} K^{ijk} dt = \iint_{\tau,\tau' \in \mathbb{R}^2} \Big(& \mathbb{E}(dN_t^i dN_{t+\tau}^j dN_{t+\tau'}^k) + 2\mathbb{E}(dN_t^i)\mathbb{E}(dN_{t+\tau}^j)\mathbb{E}(dN_{t+\tau'}^k) \\ & - \mathbb{E}(dN_t^i dN_{t+\tau}^j)\mathbb{E}(dN_{t+\tau'}^k) - \mathbb{E}(dN_t^i dN_{t+\tau'}^k)\mathbb{E}(dN_{t+\tau}^j) - \mathbb{E}(dN_{t+\tau}^j dN_{t+\tau'}^k)\mathbb{E}(dN_t^i) \Big), \end{aligned} \tag{5}$$

where Eq. (3) is the mean intensity of the Hawkes process, the second-order cumulant (4) refers to the integrated covariance density matrix and the third-order cumulant (5) measures the skewness of $\boldsymbol{N}_t$. Using the martingale representation from [BM16] or the Poisson cluster process representation from [JHR15], one can obtain an explicit relationship between these integrated cumulants and the matrix $\boldsymbol{G}$. If one sets

$$\boldsymbol{R} = (\boldsymbol{I_d} - \boldsymbol{G})^{-1}, \tag{6}$$

straightforward computations (see Section 4) lead to the following identities:

$$\Lambda^i = \sum_{m=1}^{d} R^{im} \mu^m \tag{7}$$

$$C^{ij} = \sum_{m=1}^{d} \Lambda^m R^{im} R^{jm} \tag{8}$$

$$K^{ijk} = \sum_{m=1}^{d} (R^{im} R^{jm} C^{km} + R^{im} C^{jm} R^{km} + C^{im} R^{jm} R^{km} - 2\Lambda^m R^{im} R^{jm} R^{km}). \tag{9}$$

Equations (8) and (9) are proved in Section 4. Our strategy is to use a convenient subset of Eqs. (3), (4) and (5) to define $\boldsymbol{M}$, while we use Eqs. (7), (8) and (9) in order to construct the operator that maps a candidate matrix $\boldsymbol{R}$ to the corresponding cumulants $M(\boldsymbol{R})$. By looking for $\widehat{\boldsymbol{R}}$ that minimizes $\boldsymbol{R} \mapsto \|M(\boldsymbol{R}) - \widehat{\boldsymbol{M}}\|^2$, we obtain, as illustrated below, good recovery of the ground truth matrix $\boldsymbol{G}$ using Equation (6).

The simplest case $d = 1$ has been considered in [HB14], where it is shown that one can choose $M = \{C^{11}\}$ in order to compute the kernel integral. Eq. (8) then reduces to a simple second-order equation that has a unique solution in $\boldsymbol{R}$ (and consequently a unique $\boldsymbol{G}$) that accounts for the stability condition ($\|\boldsymbol{G}\| < 1$).

Unfortunately, for $d > 1$, the choice $M = \{C^{ij}\}_{1 \le i \le j \le d}$ is not sufficient to uniquely determine the kernels integrals. In fact, the integrated covariance matrix provides $d(d+1)/2$ independent coefficients, while d^2 parameters are needed. It is straightforward to show that the remaining $d(d-1)/2$ conditions can be encoded in an orthogonal matrix $\boldsymbol{O}$, reflecting the fact that Eq. (8) is invariant under the change $\boldsymbol{R} \to \boldsymbol{OR}$, so that the system is under-determined.

Our approach relies on using the third order cumulant tensor $\boldsymbol{K} = [K^{ijk}]$ which contains $(d^3 + 3d^2 + 2d)/6 > d^2$ independent coefficients that are sufficient to uniquely fix the matrix

$\boldsymbol{G}$. This can be justified intuitively as follows: while the integrated covariance only contains symmetric information, and is thus unable to provide causal information, *the skewness given by the third order cumulant in the estimation procedure can break the symmetry between past and future so as to uniquely fix* $\boldsymbol{G}$. Thus, our algorithm consists of selecting d^2 third-order cumulant components, namely $M = \{K^{iij}\}_{1\le i,j\le d}$. In particular, we define the estimator of $\boldsymbol{R}$ as $\widehat{\boldsymbol{R}} \in \operatorname{argmin}_{\boldsymbol{R}} \mathscr{L}(\boldsymbol{R})$, where

$$\mathscr{L}(\boldsymbol{R}) = (1-\kappa)\|\boldsymbol{K}^{\boldsymbol{c}}(\boldsymbol{R}) - \widehat{\boldsymbol{K}^{\boldsymbol{c}}}\|_2^2 + \kappa\|\boldsymbol{C}(\boldsymbol{R}) - \widehat{\boldsymbol{C}}\|_2^2, \tag{10}$$

where $\|\cdot\|_2$ stands for the Frobenius norm, $\boldsymbol{K}^{\boldsymbol{c}} = \{K^{iij}\}_{1\le i,j\le d}$ is the matrix obtained by the contraction of the tensor $\boldsymbol{K}$ to d^2 indices, $\boldsymbol{C}$ is the covariance matrix, while $\widehat{\boldsymbol{K}^{\boldsymbol{c}}}$ and $\widehat{\boldsymbol{C}}$ are their respective estimators, see Equations (12), (13) below. It is noteworthy that the above mean square error approach can be seen as a particular Generalized Method of Moments (GMM), see [Hal05]. This framework allows us to determine the optimal weighting matrix involved in the loss function. However, this approach is unusable in practice, since the associated complexity is too high. Indeed, since we have d^2 parameters, this matrix has d^4 coefficients and GMM calls for computing its inverse leading to a $O(d^6)$ complexity. In this work, we use the coefficient κ to scale the two terms, as

$$\kappa = \frac{\|\widehat{\boldsymbol{K}^{\boldsymbol{c}}}\|_2^2}{\|\widehat{\boldsymbol{K}^{\boldsymbol{c}}}\|_2^2 + \|\widehat{\boldsymbol{C}}\|_2^2},$$

see Section 4.4 for an explanation about the link between κ and the weighting matrix. Finally, the estimator of $\boldsymbol{G}$ is straightforwardly obtained as

$$\widehat{\boldsymbol{G}} = \boldsymbol{I_d} - \widehat{\boldsymbol{R}}^{-1},$$

from the inversion of Eq. (6). Let us mention an important point: the matrix inversion in the previous formula is not the bottleneck of the algorithm. Indeed, its has a complexity $O(d^3)$ that is cheap compared to the computation of the cumulants when $n = \max_i |Z^i| \gg d$, which is the typical scaling satisfied in applications. Solving the considered problem on a larger scale, say $d \gg 10^3$, is an open question, even with state-of-the-art parametric and nonparametric approaches, see for instance [ZZS13, XFZ16, ZZS13, BM16], where the number of components d in experiments is always around 100 or smaller. Note that, actually, our approach leads to a *much faster* algorithm than the considered state-of-the-art baselines, see Tables 1–4 from Section 3 below.

2.3 Estimation of the integrated cumulants

In this section we present explicit formulas to estimate the three moment-based quantities listed in the previous section, namely, $\boldsymbol{\Lambda}$, $\boldsymbol{C}$ and $\boldsymbol{K}$. We first assume there exists $H > 0$ such that the truncation from $(-\infty, +\infty)$ to $[-H, H]$ of the domain of integration of the quantities appearing in Eqs. (4) and (5), introduces only a small error. In practice, this amounts to neglecting border effects in the covariance density and in the skewness density that is a good

approximation if the support of the kernel $\phi^{ij}(t)$ is smaller than H and the spectral norm $\|\boldsymbol{G}\|$ satisfies $\|\boldsymbol{G}\| < 1$.
In this case, given a realization of a stationary Hawkes process $\{\boldsymbol{N}_t : t \in [0, T]\}$, as shown in Section 4, we can write the estimators of the first three cumulants (3), (4) and (5) as

$$\widehat{\Lambda}^i = \frac{1}{T} \sum_{\tau \in Z^i} 1 = \frac{N_T^i}{T} \tag{11}$$

$$\widehat{C}^{ij} = \frac{1}{T} \sum_{\tau \in Z^i} \left(N_{\tau+H}^j - N_{\tau-H}^j - 2H\widehat{\Lambda}^j \right) \tag{12}$$

$$\begin{aligned} \widehat{K}^{ijk} = \frac{1}{T} \sum_{\tau \in Z^i} & \left(N_{\tau+H}^j - N_{\tau-H}^j - 2H\widehat{\Lambda}^j \right) \cdot \left(N_{\tau+H}^k - N_{\tau-H}^k - 2H\widehat{\Lambda}^k \right) \\ & - \frac{\widehat{\Lambda}^i}{T} \sum_{\tau \in Z^j} \sum_{\tau' \in Z^k} (2H - |\tau' - \tau|)^+ + 4H^2 \widehat{\Lambda}^i \widehat{\Lambda}^j \widehat{\Lambda}^k. \end{aligned} \tag{13}$$

Let us mention the following facts.

Bias. While the first cumulant $\hat{\Lambda}^i$ is an unbiased estimator of Λ^i, the other estimators $\widehat{C}^{ij}$ and $\widehat{K}^{ijk}$ introduce a bias. However, as we will show, in practice this bias is small and hardly affects numerical estimations (see Section 3). This is confirmed by our theoretical analysis, which proves that if H does not grow too fast compared to T, then these estimated cumulants are consistent estimators of the theoretical cumulants (see Section 2.6).

Complexity. The computations of all the estimators of the first, second and third-order cumulants have complexity respectively $O(nd)$, $O(nd^2)$ and $O(nd^3)$, where $n = \max_i |Z^i|$. However, our algorithm requires a lot less than that: it computes only d^2 third-order terms, of the form $\widehat{K}^{iij}$, leaving us with only $O(nd^2)$ operations to perform.

Symmetry. While the values of Λ^i, C^{ij} and K^{ijk} are symmetric under permutation of the indices, their estimators are generally not symmetric. We have thus chosen to symmetrize the estimators by averaging their values over permutations of the indices. Worst case is for the estimator of $\boldsymbol{K^c}$, which involves only an extra factor of 2 in the complexity.

2.4 The NPHC algorithm

The objective to minimize in Equation (10) is non-convex. More precisely, the loss function is a polynomial of $\boldsymbol{R}$ of degree 6. However, the expectations of cumulants $\boldsymbol{\Lambda}$ and $\boldsymbol{C}$ defined in Eq. (4) and (5) that appear in the definition of $\mathscr{L}(\boldsymbol{R})$ are unknown and should be replaced with $\widehat{\boldsymbol{\Lambda}}$ and $\widehat{\boldsymbol{C}}$. We denote $\widetilde{\mathscr{L}}(\boldsymbol{R})$ the objective function, where the expectations of cumulants Λ^i and C^{ij} have been replaced with their estimators in the right-hand side of Eqs. (8) and (9):

$$\widetilde{\mathscr{L}}(\boldsymbol{R}) = (1-\kappa)\|\boldsymbol{R}^{\odot 2}\widehat{\boldsymbol{C}}^\top + 2[\boldsymbol{R} \odot (\widehat{\boldsymbol{C}} - \boldsymbol{R}\widehat{\boldsymbol{L}})]\boldsymbol{R}^\top - \widehat{\boldsymbol{K}^c}\|_2^2 + \kappa\|\boldsymbol{R}\widehat{\boldsymbol{L}}\boldsymbol{R}^\top - \widehat{\boldsymbol{C}}\|_2^2 \tag{14}$$

As explained in [CHM+15], the loss function of a typical multilayer neural network with simple nonlinearities can be expressed as a polynomial function of the weights in the network, whose degree is the number of layers. Since the loss function of NPHC writes as a polynomial of degree 6, we expect good results using optimization methods designed to train deep multilayer neural networks. We used the AdaGrad from [DHS11], a variant of the Stochastic Gradient Descent with adaptive learning rates. AdaGrad scales the learning rates coordinate-wise using the online variance of the previous gradients, in order to incorporate second-order information during training. The NPHC method is summarized schematically in Algorithm 10.

Algorithm 10 Non Parametric Hawkes Cumulant method

1: **Input**: $\boldsymbol{N}_t$
2: **Output**: $\widehat{\boldsymbol{G}}$
3: Estimate $\widehat{\Lambda}^i$, $\widehat{C}^{ij}, \widehat{K}^{iij}$ from Eqs. (11, 12, 13)
4: Design $\widetilde{\mathscr{L}}(\boldsymbol{R})$ using the computed estimators.
5: Minimize numerically $\widetilde{\mathscr{L}}(\boldsymbol{R})$ so as to obtain $\widehat{\boldsymbol{R}}$
6: Return $\widehat{\boldsymbol{G}} = \boldsymbol{I_d} - \widehat{\boldsymbol{R}}^{-1}$.

Our problem being non-convex, the choice of the starting point has a major effect on the convergence. Here, the key is to notice that the matrices $\boldsymbol{R}$ that match Equation (8) writes $\boldsymbol{C}^{1/2}\boldsymbol{O}\boldsymbol{L}^{-1/2}$, with $\boldsymbol{L} = \mathrm{diag}(\boldsymbol{\Lambda})$ and $\boldsymbol{O}$ an orthogonal matrix. Our starting point is then simply chosen by setting $\boldsymbol{O} = \boldsymbol{I_d}$ in the previous formula, leading to nice convergence results. Even though our main concern is to retrieve the matrix $\boldsymbol{G}$, let us notice we can also obtain an estimation of the baseline intensities' from Eq. (3), which leads to $\widehat{\boldsymbol{\mu}} = \widehat{\boldsymbol{R}}^{-1}\widehat{\boldsymbol{\Lambda}}$. An efficient implementation of this algorithm with TensorFlow, see [AAB+16], is available on GitHub: https://github.com/achab/nphc.

2.5 Complexity of the algorithm

Compared with existing state-of-the-art methods to estimate the kernel functions, e.g., the ordinary differential equations-based (ODE) algorithm in [ZZS13], the Granger Causality-based algorithm in [XFZ16], the ADM4 algorithm in [ZZS13], and the Wiener-Hopf-based algorithm in [BM16], our method has a very competitive complexity. This can be understood by the fact that those methods estimate the kernel functions, while in NPHC we only estimate their integrals. The ODE-based algorithm is an EM algorithm that parametrizes the kernel function with M basis functions, each being discretized to L points. The basis functions are updated after solving M Euler-Lagrange equations. If n denotes the maximum number of events per component (i.e. $n = \max_{1\leq i\leq d} |Z^i|$) then the complexity of one iteration of the algorithm is $O(Mn^3d^2 + ML(nd + n^2))$. The Granger Causality-based algorithm is similar to the previous one, without the update of the basis functions, that are Gaussian kernels. The complexity per iteration is $O(Mn^3d^2)$. The algorithm ADM4 is similar to the two algorithms above, as EM algorithm as well, with only one exponential kernel as basis function. The complexity per iteration is then $O(n^3d^2)$. The Wiener-Hopf-based algorithm is not iterative,

on the contrary to the previous ones. It first computes the empirical conditional laws on many points, and then invert the Wiener-Hopf system, leading to a $O(nd^2L + d^4L^3)$ computation. Similarly, our method first computes the integrated cumulants, then minimize the objective function with N_{iter} iterations, and invert the resulting matrix $\widehat{\boldsymbol{R}}$ to obtain $\widehat{\boldsymbol{G}}$. In the end, the complexity of the NPHC method is $O(nd^2 + N_{\text{iter}}d^3)$. According to this analysis, summarized in Table III.1 below, one can see that in the regime $n \gg d$, the NPHC method outperforms all the other ones.

Table III.1: Complexity of state-of-the-art methods. NPHC's complexity is very low , especially in the regime $n \gg d$.

Method	Total complexity
ODE [ZZS13]	$O(N_{\text{iter}}M(n^3d^2 + L(nd + n^2)))$
GC [XFZ16]	$O(N_{\text{iter}}Mn^3d^2)$
ADM4 [ZZS13]	$O(N_{\text{iter}}n^3d^2)$
WH [BM16]	$O(nd^2L + d^4L^3)$
NPHC	$O(nd^2 + N_{\text{iter}}d^3)$

2.6 Theoretical guarantee: consistency

The NPHC method can be phrased using the framework of the Generalized Method of Moments (GMM). GMM is a generic method for estimating parameters in statistical models. In order to apply GMM, we have to find a vector-valued function $g(X, \theta)$ of the data, where X is distributed with respect to a distribution $\mathbb{P}_{\theta_0}$, which satisfies the *moment condition*: $\mathbb{E}[g(X, \theta)] = 0$ if and only if $\theta = \theta_0$, where θ_0 is the "ground truth" value of the parameter. Based on i.i.d. observed copies $x_1, \dots, x_n$ of X, the GMM method minimizes the norm of the empirical mean over n samples, $\| \frac{1}{n} \sum_{i=1}^{n} g(x_i, \theta) \|$, as a function of θ, to obtain an estimate of θ_0.

In the theoretical analysis of NPHC, we use ideas from the consistency proof of the GMM, but the proof actually relies on very different arguments. Indeed, the integrated cumulants estimators used in NPHC are not unbiased, as the theory of GMM requires, but asymptotically unbiased. Moreover, the setting considered here, where data consists of a single realization $\{\boldsymbol{N}_t\}$ of a Hawkes process strongly departs from the standard i.i.d setting. Our approach is therefore based on the GMM idea but the proof is actually not using the theory of GMM.

In the following, we use the subscript T to refer to quantities that only depend on the process $(\boldsymbol{N_t})$ in the interval $[0, T]$ (e.g., the truncation term H_T, the estimated integrated covariance $\widehat{\boldsymbol{C}}_T$ or the estimated kernel norm matrix $\widehat{\boldsymbol{G}}_T$). In the next equation, $\odot$ stands for the Hadamard product and $\odot 2$ stands for the entrywise square of a matrix. We denote

$\boldsymbol{G}_0 = \boldsymbol{I_d} - \boldsymbol{R}_0^{-1}$ the true value of $\boldsymbol{G}$, and the $\mathbb{R}^{2d\times d}$ valued vector functions

$$g_0(\boldsymbol{R}) = \begin{bmatrix} \boldsymbol{C} - \boldsymbol{R}\boldsymbol{L}\boldsymbol{R}^\top \\ \boldsymbol{K^c} - \boldsymbol{R}^{\odot 2}\boldsymbol{C}^\top - 2[\boldsymbol{R}\odot(\boldsymbol{C}-\boldsymbol{R}\boldsymbol{L})]\boldsymbol{R}^\top \end{bmatrix}$$
$$\widehat{g}_T(\boldsymbol{R}) = \begin{bmatrix} \widehat{\boldsymbol{C}}_T - \boldsymbol{R}\widehat{\boldsymbol{L}}_T\boldsymbol{R}^\top \\ \widehat{\boldsymbol{K}^{\boldsymbol{c}}_{\boldsymbol{T}}} - \boldsymbol{R}^{\odot 2}\widehat{\boldsymbol{C}}_T^\top - 2[\boldsymbol{R}\odot(\widehat{\boldsymbol{C}}_T - \boldsymbol{R}\widehat{\boldsymbol{L}}_T)]\boldsymbol{R}^\top \end{bmatrix}.$$

Using these notations, $\widetilde{\mathscr{L}}_T(\boldsymbol{R})$ can be seen as the weighted squared Frobenius norm of $\widehat{g}_T(\boldsymbol{R})$. Moreover, when $T \to +\infty$, one has $\widehat{g}_T(\boldsymbol{R}) \xrightarrow{\mathbb{P}} g_0(\boldsymbol{R})$ under the conditions of the following theorem, where $\xrightarrow{\mathbb{P}}$ stands for convergence in probability.

Theorem 1 (Consistency of NPHC). *Suppose that* $(\boldsymbol{N_t})$ *is observed on* $\mathbb{R}^+$ *and assume that*

1. $g_0(\boldsymbol{R}) = 0$ *if and only if* $\boldsymbol{R} = \boldsymbol{R}_0$;
2. $\boldsymbol{R} \in \Theta$, *where* Θ *is a compact set*;
3. *the spectral radius of the kernel norm matrix satisfies* $\|\boldsymbol{G}_0\| < 1$;
4. $H_T \to \infty$ *and* $H_T^2 / T \to 0$.

Then

$$\widehat{\boldsymbol{G}}_T = \boldsymbol{I_d} - \left(\arg\min_{\boldsymbol{R}\in\Theta} \widetilde{\mathscr{L}}_T(\boldsymbol{R})\right)^{-1} \xrightarrow{\mathbb{P}} \boldsymbol{G}_0.$$

The proof of the Theorem is given in Section 4.5 below. Assumption 3 is mandatory for stability of the Hawkes process, and Assumptions 3 and 4 are sufficient to prove that the estimators of the integrated cumulants defined in Equations (11), (12) and (13) are asymptotically consistent. Assumption 2 is a very mild standard technical assumption allowing to prove consistency for estimators based on moments. Assumption 1 is a standard asymptotic moment condition, that allows to identify parameters from the integrated cumulants.

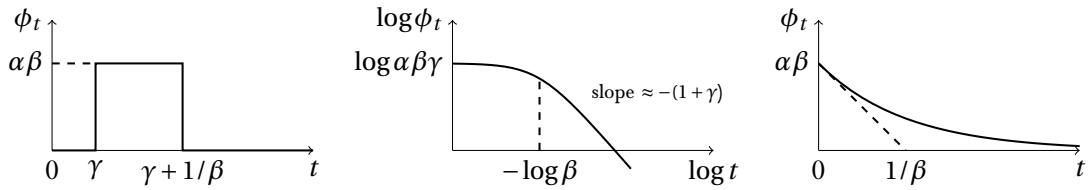
3 Numerical Experiments

In this Section, we provide a comparison of NPHC with the state-of-the art, on simulated datasets with different kernel shapes, the MemeTracker dataset (social networks) and the order book dynamics dataset (finance).

Simulated datasets. We simulated several datasets with Ogata's Thinning algorithm [Oga81] using the open-source library `tick`[1], each corresponding to a shape of kernel: rectangular, exponential or power law kernel, see Figure III.1 below.

The integral of each kernel on its support equals α, $1/\beta$ can be regarded as a characteristic time-scale and γ is the scaling exponent for the power law distribution and a delay parameter

[1] https://github.com/X-DataInitiative/tick



(a) Rectangular kernel
$\phi_t = \alpha\beta\mathbb{1}_{[0,1/\beta]}(t-\gamma)$

(b) Power law kernel on log-log scale
$\phi_t = \alpha\beta\gamma(1+\beta t)^{-(1+\gamma)}$

(c) Exponential kernel
$\phi_t = \alpha\beta\exp(-\beta t)$

Figure III.1: The three different kernels used to simulate the datasets.

for the rectangular one. We consider a non-symmetric block-matrix $\boldsymbol{G}$ to show that our method can effectively uncover causality between the nodes, see Figure III.2. The matrix $\boldsymbol{G}$ has constant entries α on the three blocks - $\alpha = g^{ij} = 1/6$ for dimension 10 and $\alpha = g^{ij} = 1/10$ for dimension 100 -, and zero outside. The two other parameters' values are the same for dimensions 10 and 100. The parameter γ is set to 1/2 on the three blocks as well, but we set three very different β_0, β_1 and β_2 from one block to the other, with ratio $\beta_{i+1}/\beta_i = 10$ and $\beta_0 = 0.1$. The number of events is roughly equal to 10^5 on average over the nodes. We ran the algorithm on three simulated datasets: a 10-dimensional process with rectangular kernels named Rect10, a 10-dimensional process with power law kernels named PLaw10 and a 100-dimensional process with exponential kernels named Exp100.

MemeTracker dataset. We use events of the most active sites from the MemeTracker dataset[2]. This dataset contains the publication times of articles in many websites/blogs from August 2008 to April 2009, and hyperlinks between posts. We extract the top 100 media sites with the largest number of documents, with about 7 million of events. We use the links to trace the flow of information and establish an estimated ground truth for the matrix $\boldsymbol{G}$. Indeed, when an hyperlink j appears in a post in website i, the link j can be regarded as a direct ancestor of the event. Then, Eq. (2) shows g^{ij} can be estimated by $N_T^{i\leftarrow j}/N_T^j = \#\{\text{links } j \to i\}/N_T^j$.

Order book dynamics. We apply our method to financial data, in order to understand the self and cross-influencing dynamics of all event types in an order book. An order book is a list of buy and sell orders for a specific financial instrument, the list being updated in real-time throughout the day. This model has first been introduced in [BJM16], and models the order book via the following 8-dimensional point process: $N_t = (P_t^{(a)}, P_t^{(b)}, T_t^{(a)}, T_t^{(b)}, L_t^{(a)}, L_t^{(b)}, C_t^{(a)}, C_t^{(b)})$, where $P^{(a)}$ (resp. $P^{(b)}$) counts the number of upward (resp. downward) price moves, $T^{(a)}$ (resp. $T^{(b)}$) counts the number of market orders at the ask[3] (resp. at the bid) that do not move the price, $L^{(a)}$ (resp. $L^{(b)}$) counts the number of limit orders at the ask[4] (resp. at the bid) that do

[2] https://www.memetracker.org/data.html

[3] i.e. buy orders that are executed and removed from the list

[4] i.e. buy orders added to the list

not move the price, and $C^{(a)}$ (resp. $C^{(b)}$) counts the number of cancel orders at the ask[5] (resp. at the bid) that do not move the price. The financial data has been provided by QuantHouse EUROPE/ASIA, and consists of DAX future contracts between 01/01/2014 and 03/01/2014.

Baselines. We compare NPHC to state-of-the art baselines: the ODE-based algorithm (ODE) by [ZZS13], the Granger Causality-based algorithm (GC) by [XFZ16], the ADM4 algorithm (ADM4) by [ZZS13], and the Wiener-Hopf-based algorithm (WH) by [BM16].

Metrics. We evaluate the performance of the proposed methods using the computing time, the Relative Error

$$\mathrm{RelErr}(\boldsymbol{A}, \boldsymbol{B}) = \frac{1}{d^2} \sum_{i,j} \frac{|a^{ij} - b^{ij}|}{|a^{ij}|} \mathbb{1}_{\{a^{ij} \neq 0\}} + |b^{ij}| \mathbb{1}_{\{a^{ij} = 0\}}$$

and the Mean Kendall Rank Correlation

$$\mathrm{MRankCorr}(\boldsymbol{A}, \boldsymbol{B}) = \frac{1}{d} \sum_{i=1}^{d} \mathrm{RankCorr}([a^{i\bullet}], [b^{i\bullet}]),$$

where $\mathrm{RankCorr}(x, y) = \frac{2}{d(d-1)}(N_{\mathrm{concordant}}(x, y) - N_{\mathrm{discordant}}(x, y))$ with $N_{\mathrm{concordant}}(x, y)$ the number of pairs (i, j) satisfying $x_i > x_j$ and $y_i > y_j$ or $x_i < x_j$ and $y_i < y_j$ and $N_{\mathrm{discordant}}(x, y)$ the number of pairs (i, j) for which the same condition is not satisfied.

Note that RankCorr score is a value between -1 and 1, representing rank matching, but can take smaller values (in absolute value) if the entries of the vectors are not distinct.

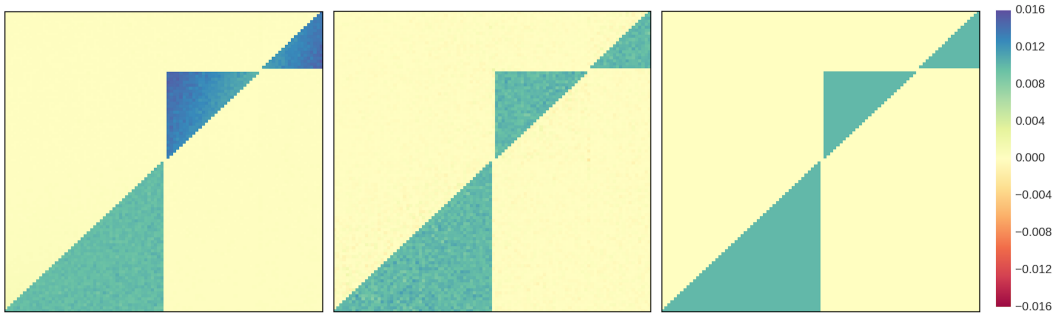


Figure III.2: On Exp100 dataset, estimated $\widehat{\boldsymbol{G}}$ with ADM4 (left), with NPHC (middle) and the ground-truth matrix $\boldsymbol{G}$ (right). Both ADM4 and NPHC estimates recover the three blocks. However, ADM4 overestimates the integrals on two of the three blocks, while NPHC gives the same value on each blocks.

[5] i.e. the number of times a limit order at the ask is canceled: in our dataset, almost 95% of limit orders are canceled before execution.

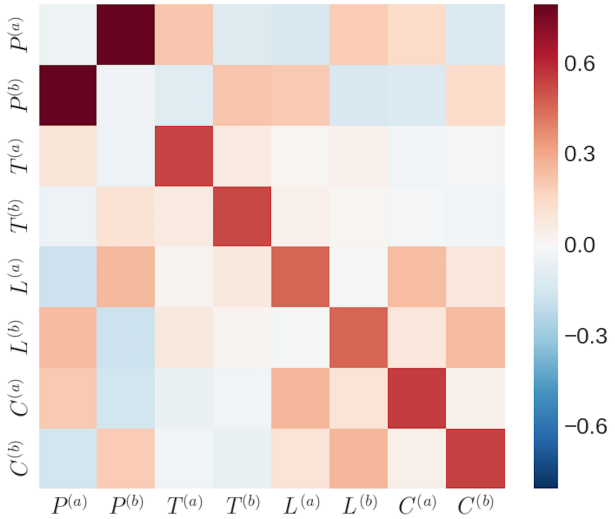


Figure III.3: Estimated $\widehat{\boldsymbol{G}}$ via NPHC on DAX order book data.

Table III.2: Metrics on Rect10: comparable rank correlation, strong improvement for relative error and computing time.

Method	ODE	GC	ADM4	WH	NPHC
RelErr	0.007	0.15	0.10	0.005	**0.001**
MRankCorr	0.33	0.02	0.21	**0.34**	**0.34**
Time (s)	846	768	709	933	**20**

Table III.3: Metrics on PLaw10: comparable rank correlation, strong improvement for relative error and computing time.

Method	ODE	GC	ADM4	WH	NPHC
RelErr	0.011	0.09	0.053	0.009	**0.0048**
MRankCorr	0.31	0.26	0.24	**0.34**	0.33
Time (s)	870	781	717	946	**18**

Discussion. We perform the ADM4 estimation, with exponential kernel, by giving the exact value $\beta = \beta_0$ of one block. Let us stress that this helps a lot this baseline, in comparison to NPHC where nothing is specified on the shape of the kernel functions. We used $M = 10$ basis functions for both ODE and GC algorithms, and $L = 50$ quadrature points for WH. We did not run WH on the 100-dimensional datasets, for computing time reasons, because its complexity scales with d^4. We ran multi-processed versions of the baseline methods on 56 cores, to decrease the computing time.

Our method consistently performs better than all baselines, on the three synthetic datasets,

Table III.4: Metrics on Exp100: comparable rank correlation, strong improvement for relative error and computing time.

Method	ODE	GC	ADM4	NPHC
RelErr	0.092	0.112	0.079	**0.008**
MRankCorr	0.032	0.009	**0.049**	0.041
Time (s)	3215	2950	2411	**47**

Table III.5: Metrics on MemeTracker: strong improvement in relative error, rank correlation and computing time.

Method	ODE	GC	ADM4	NPHC
RelErr	0.162	0.19	0.092	**0.071**
MRankCorr	0.07	0.053	0.081	**0.095**
Time (s)	2944	2780	2217	**38**

on MemeTracker and on the financial dataset, both in terms of Kendall rank correlation and estimation error. Moreover, we observe that our algorithm is roughly 50 times faster than all the considered baselines.

On Rect10, PLaw10 and Exp100 our method gives very impressive results, despite the fact that it does not uses any prior shape on the kernel functions, while for instance the ADM4 baseline do. On Figure III.2, we observe that the matrix $\widehat{\boldsymbol{G}}$ estimated with ADM4 recovers well the block for which $\beta = \beta_0$, i.e. the value we gave to the method, but does not perform well on the two other blocks, while the matrix $\widehat{\boldsymbol{G}}$ estimated with NPHC approximately reaches the true value for each of the three blocks. On these simulated datasets, NPHC obtains a comparable or slightly better Kendall rank correlation, but improves a lot the relative error.

On MemeTracker, the baseline methods obtain a high relative error between 9% and 19% while our method achieves a relative error of 7% which is a strong improvement. Moreover, NPHC reaches a much better Kendall rank correlation, which proves that it leads to a much better recovery of the relative order of estimated influences than all the baselines. Indeed, it has been shown in [ZZS13] that kernels of MemeTracker data are not exponential, nor power law. This partly explains why our approach behaves better.

On the financial data, the estimated kernel norm matrix obtained via NPHC, see Figure III.3, gave some interpretable results (see also [BJM16]):

1. Any 2×2 sub-matrix with same kind of inputs (i.e. Prices changes, Trades, Limits or Cancels) is symmetric. This shows empirically that ask and bid have symmetric roles.

2. The prices are mostly cross-excited, which means that a price increase is very likely to be followed by a price decrease, and conversely. This is consistent with the wavy prices we observe on financial markets.

3. The market, limit and cancel orders are strongly self-excited. This can be explained by the persistence of order flows, and by the splitting of meta-orders into sequences of

smaller orders. Moreover, we observe that orders impact the price without changing it. For example, the increase of cancel orders at the bid causes downward price moves.

4 Technical details

We show in this section how to obtain the equations stated above, the estimators of the integrated cumulants and the scaling coefficient κ that appears in the objective function. We then prove the theorem of the paper.

4.1 Proof of Equation (8)

We denote $\boldsymbol{\nu}(z)$ the matrix

$$\nu^{ij}(z) = \mathscr{L}_z\Big(t \to \frac{\mathbb{E}(dN_u^i dN_{u+t}^j)}{dudt} - \Lambda^i \Lambda^j\Big),$$

where $\mathscr{L}_z(f)$ is the Laplace transform of f, and $\psi_t = \sum_{n\geq 1} \phi_t^{(\star n)}$, where $\phi_t^{(\star n)}$ refers to the n^{th} auto-convolution of ϕ_t. Then we use the characterization of second-order statistics, first formulated in [Haw71a] and fully generalized in [BM16],

$$\boldsymbol{\nu}(z) = (\boldsymbol{I_d} + \mathscr{L}_{-z}(\boldsymbol{\Psi}))\boldsymbol{L}(\boldsymbol{I_d} + \mathscr{L}_z(\boldsymbol{\Psi}))^\top,$$

where $\boldsymbol{L}^{ij} = \Lambda^i \delta^{ij}$ with δ^{ij} the Kronecker symbol. Since $\boldsymbol{I_d} + \mathscr{L}_z(\boldsymbol{\Psi}) = (\boldsymbol{I_d} - \mathscr{L}_z(\boldsymbol{\Phi}))^{-1}$, taking $z = 0$ in the previous equation gives

$$\begin{aligned}\boldsymbol{\nu}(0) &= (\boldsymbol{I_d} - \boldsymbol{G})^{-1}\boldsymbol{L}(\boldsymbol{I_d} - \boldsymbol{G}^\top)^{-1},\\ \boldsymbol{C} &= \boldsymbol{R}\boldsymbol{L}\boldsymbol{R}^\top,\end{aligned}$$

which gives us the result since the entry (i, j) of the last equation gives $C^{ij} = \sum_m \Lambda^m R_{im} R_{jm}$.

4.2 Proof of Equation (9)

We start from [JHR15], cf. Eqs. (48) to (51), and group some terms:

$$\begin{aligned}K^{ijk} = &\sum_m \Lambda^m R_{im} R_{jm} R_{km}\\ &+ \sum_m R_{im} R_{jm} \sum_n \Lambda^n R_{kn} \mathscr{L}_0(\psi^{mn})\\ &+ \sum_m R_{im} R_{km} \sum_n \Lambda^n R_{jn} \mathscr{L}_0(\psi^{mn})\\ &+ \sum_m R_{jm} R_{km} \sum_n \Lambda^n R_{in} \mathscr{L}_0(\psi^{mn}).\end{aligned}$$

Using the relations $\mathscr{L}_0(\psi^{mn}) = R_{mn} - \delta^{mn}$ and $C^{ij} = \sum_m \Lambda^m R_{im} R_{jm}$, proves Equation (9).

4.3 Integrated cumulants estimators

For $H > 0$ let us denote $\Delta_H N_t^i = N_{t+H}^i - N_{t-H}^i$. Let us first remark that, if one restricts the integration domain to $(-H, H)$ in Eqs. (4) and (5), one gets by permuting integrals and expectations:

$$\begin{aligned}
\Lambda^i dt &= \mathbb{E}(dN_t^i) \\
C^{ij} dt &= \mathbb{E}\Big(dN_t^i(\Delta_H N_t^j - 2H\Lambda^j)\Big) \\
K^{ijk} dt &= \mathbb{E}\Big(dN_t^i(\Delta_H N_t^j - 2H\Lambda^j)(\Delta_H N_t^k - 2H\Lambda^k)\Big) \\
&\quad - dt\Lambda^i \mathbb{E}\Big((\Delta_H N_t^j - 2H\Lambda^j)(\Delta_H N_t^k - 2H\Lambda^k)\Big).
\end{aligned}$$

The estimators (11) and (12) are then naturally obtained by replacing the expectations by their empirical counterparts, notably

$$\frac{\mathbb{E}(dN_t^i f(t))}{dt} \to \frac{1}{T} \sum_{\tau \in Z^i} f(\tau).$$

For the estimator (13), we shall also notice that

$$\begin{aligned}
&\mathbb{E}((\Delta_H N_t^j - 2H\Lambda^j)(\Delta_H N_t^k - 2H\Lambda^k)) \\
&\quad = \int\int \mathbb{1}_{[-H,H]}(t)\mathbb{1}_{[-H,H]}(t') C_{t-t'}^{jk} dt dt' \\
&\quad = \int (2H - |t|)^+ C_t^{jk} dt.
\end{aligned}$$

We estimate the last integral with the remark above.

4.4 Choice of the scaling coefficient κ

Following the theory of GMM, we denote $m(X, \theta)$ a function of the data, where X is distributed with respect to a distribution $\mathbb{P}_{\theta_0}$, which satisfies the *moment conditions* $g(\theta) = \mathbb{E}[m(X, \theta)] = 0$ if and only if $\theta = \theta_0$, the parameter θ_0 being the *ground truth.* For $x_1, \dots, x_N$ observed copies of X, we denote $\widehat{g}_i(\theta) = m(x_i, \theta)$, the usual choice of weighting matrix is $\widehat{W}_N(\theta) = \frac{1}{N}\sum_{i=1}^N \widehat{g}_i(\theta)\widehat{g}_i(\theta)^\top$, and the objective to minimize is then

$$\left(\frac{1}{N}\sum_{i=1}^N \widehat{g}_i(\theta)\right)\left(\widehat{W}_N(\theta_1)\right)^{-1}\left(\frac{1}{N}\sum_{i=1}^N \widehat{g}_i(\theta)\right), \tag{15}$$

where θ_1 is a constant vector. Instead of computing the inverse weighting matrix, we rather use its projection on $\{\alpha \boldsymbol{I_d} : \alpha \in \mathbb{R}\}$. It can be shown that the projection choses α as the mean eigenvalue of $\widehat{W}_N(\theta_1)$. We can easily compute the sum of its eigenvalues:

$$\mathrm{Tr}(\widehat{W}_N(\theta_1)) = \frac{1}{N}\sum_{i=1}^N \mathrm{Tr}(\widehat{g}_i(\theta_1)\widehat{g}_i(\theta_1)^\top) = \frac{1}{N}\sum_{i=1}^N \mathrm{Tr}(\widehat{g}_i(\theta_1)^\top \widehat{g}_i(\theta_1)) = \frac{1}{N}\sum_{i=1}^N ||\widehat{g}_i(\theta_1)||_2^2.$$

In our case, $\widehat{g}(\boldsymbol{R}) = \left[\mathbf{vec}[\widehat{\boldsymbol{K^c}} - \boldsymbol{K^c}(\boldsymbol{R})], \mathbf{vec}[\widehat{\boldsymbol{C}} - \boldsymbol{C}(\boldsymbol{R})]\right]^\top \in \mathbb{R}^{2d^2}$. Considering a block-wise weighting matrix, one block for $\widehat{\boldsymbol{K^c}} - \boldsymbol{K^c}(\boldsymbol{R})$ and the other for $\widehat{\boldsymbol{C}} - \boldsymbol{C}(\boldsymbol{R})$, the sum of the eigenvalues of the first block becomes $\|\widehat{\boldsymbol{K^c}} - \boldsymbol{K^c}(\boldsymbol{R})\|_2^2$, and $\|\widehat{\boldsymbol{C}} - \boldsymbol{C}(\boldsymbol{R})\|_2^2$ for the second. We compute the previous terms with $\boldsymbol{R}_1 = 0$. All together, the objective function to minimize is

$$\frac{1}{\|\widehat{\boldsymbol{K^c}}\|_2^2}\|\boldsymbol{K^c}(\boldsymbol{R}) - \widehat{\boldsymbol{K^c}}\|_2^2 + \frac{1}{\|\widehat{\boldsymbol{C}}\|_2^2}\|\boldsymbol{C}(\boldsymbol{R}) - \widehat{\boldsymbol{C}}\|_2^2. \tag{16}$$

Dividing this function by $\left(1/\|\widehat{\boldsymbol{K^c}}\|_2^2 + 1/\|\widehat{\boldsymbol{C}}\|_2^2\right)^{-1}$, and setting $\kappa = \|\widehat{\boldsymbol{K^c}}\|_2^2/(\|\widehat{\boldsymbol{K^c}}\|_2^2 + \|\widehat{\boldsymbol{C}}\|_2^2)$, we obtaind the loss function given in Equation (10).

4.5 Proof of the Theorem

The main difference with the usual Generalized Method of Moments, see [Han82], relies in the relaxation of the moment conditions, since we have $\mathbb{E}[\widehat{g}_T(\theta_0)] = m_T \neq 0$. We adapt the proof of consistency given in [NM94].

We can relate the integral of the Hawkes process's kernels to the integrals of the cumulant densities, from [JHR15]. Our cumulant matching method would fall into the usual GMM framework if we could estimate - without bias - the integral of the covariance on $\mathbb{R}$, and the integral of the skewness on $\mathbb{R}^2$. Unfortunately, we can't do that easily. We can however estimate without bias $\int f_t^T C_t^{ij}\, dt$ and $\int f_t^T K_t^{ijk}\, dt$ with f^T a compact supported function on $[-H_T, H_T]$ that weakly converges to 1, with $H_T \longrightarrow \infty$. In most cases we will take $f_t^T = \mathbb{1}_{[-H_T,H_T]}(t)$. Denoting $\widehat{C}^{ij,(T)}$ the estimator of $\int f_t^T C_t^{ij}\, dt$, the term $\mid \mathbb{E}[\widehat{C}^{ij,(T)}] - C^{ij} \mid = \mid \int f_t^T C_t^{ij}\, dt - C^{ij} \mid$ can be considered a proxy to the *distance to the classical GMM*. This distance has to go to zero to make the rest of GMM's proof work: the estimator $\widehat{C}^{ij,(T)}$ is then asymptotically unbiased towards C^{ij} when T goes to infinity.

4.5.1 Notations

We observe the multivariate point process $(\boldsymbol{N}_t)$ on $\mathbb{R}^+$, with Z^i the events of the i^{th} component. We will often write covariance / skewness instead of integrated covariance / skewness. In the rest of the document, we use the following notations.

Hawkes kernels' integrals $\quad \boldsymbol{G}^{\text{true}} = \int \boldsymbol{\Phi}_t dt = (\int \phi_t^{ij} dt)_{ij} = \boldsymbol{I_d} - (\boldsymbol{R}^{\text{true}})^{-1}$

Theoretical mean matrix $\quad \boldsymbol{L} = \text{diag}(\Lambda^1, \ldots, \Lambda^d)$

Theoretical covariance $\quad \boldsymbol{C} = \boldsymbol{R}^{\text{true}} \boldsymbol{L} (\boldsymbol{R}^{\text{true}})^\top$

Theoretical skewness $\quad \boldsymbol{K^c} = (K^{iij})_{ij} = (\boldsymbol{R}^{\text{true}})^{\odot^2} \boldsymbol{C}^\top + 2[\boldsymbol{R}^{\text{true}} \odot (\boldsymbol{C} - \boldsymbol{R}^{\text{true}} \boldsymbol{L})](\boldsymbol{R}^{\text{true}})^\top$

Filtering function $\quad f^T \geq 0 \qquad \text{supp}(f^T) \subset [-H_T, H_T] \qquad F^T = \int f_s^T ds \qquad \widetilde{f}_t^T = f_{-t}^T$

Events sets $Z^{i,T,1} = Z^i \cap [H_T, T+H_T]$ $\qquad$ $Z^{j,T,2} = Z^j \cap [0, T+2H_T]$

Estimators of the mean $\widehat{\Lambda}^i = \frac{N^i_{T+H_T} - N^i_{H_T}}{T}$ $\qquad$ $\widetilde{\Lambda}^j = \frac{N^j_{T+2H_T}}{T+2H_T}$

Estimator of the covariance $\widehat{C}^{ij,(T)} = \frac{1}{T}\sum_{\tau\in Z^{i,T,1}} \left(\sum_{\tau'\in Z^{j,T,2}} f_{\tau'-\tau} - \widetilde{\Lambda}^j F^T\right)$

Estimator of the skewness[6]

$$\widehat{K}^{ijk,(T)} = \frac{1}{T}\sum_{\tau\in Z^{i,T,1}} \left(\sum_{\tau'\in Z^{j,T,2}} f_{\tau'-\tau} - \widetilde{\Lambda}^j F^T\right)\left(\sum_{\tau''\in Z^{k,T,2}} f_{\tau'-\tau} - \widetilde{\Lambda}^k F^T\right) - \frac{\widehat{\Lambda}^i}{T+2H_T}\sum_{\tau'\in Z^{j,T,2}} \left(\sum_{\tau''\in Z^{k,T,2}} (f^T \star \widetilde{f}^T)_{\tau'-\tau''} - \widetilde{\Lambda}^k (F^T)^2\right)$$

GMM related notations

$$\theta = \boldsymbol{R} \quad \text{and} \quad \theta_0 = \boldsymbol{R}^{\text{true}}$$

$$g_0(\theta) = \mathsf{vec}\begin{bmatrix} \boldsymbol{C} - \boldsymbol{R}\boldsymbol{L}\boldsymbol{R}^\top \\ \boldsymbol{K^c} - \boldsymbol{R}^{\odot^2}\boldsymbol{C}^\top - 2[\boldsymbol{R}\odot(\boldsymbol{C}-\boldsymbol{R}\boldsymbol{L})]\boldsymbol{R}^\top \end{bmatrix} \in \mathbb{R}^{2d^2}$$

$$\widehat{g}_T(\theta) = \mathsf{vec}\begin{bmatrix} \widehat{\boldsymbol{C}}^{(T)} - \boldsymbol{R}\widehat{\boldsymbol{L}}\boldsymbol{R}^\top \\ \widehat{\boldsymbol{K^c}}^{(T)} - \boldsymbol{R}^{\odot^2}(\widehat{\boldsymbol{C}}^{(T)})^\top - 2[\boldsymbol{R}\odot(\widehat{\boldsymbol{C}}^{(T)} - \boldsymbol{R}\widehat{\boldsymbol{L}})]\boldsymbol{R}^\top \end{bmatrix} \in \mathbb{R}^{2d^2}$$

$$Q_0(\theta) = g_0(\theta)^\top W g_0(\theta)$$

$$\widehat{Q}_T(\theta) = \widehat{g}_T(\theta)^\top \widehat{W}_T \widehat{g}_T(\theta)$$

4.5.2 Consistency

First, let's remind a useful theorem for consistency in GMM from [NM94].

Theorem 2. *If there is a function $Q_0(\theta)$ such that (i) $Q_0(\theta)$ is uniquely maximized at θ_0; (ii) Θ is compact; (iii) $Q_0(\theta)$ is continuous; (iv) $\widehat{Q}_T(\theta)$ converges uniformly in probability to $Q_0(\theta)$, then $\widehat{\theta}_T = \arg\max \widehat{Q}_T(\theta) \xrightarrow{\mathbb{P}} \theta_0$.*

We can now prove the consistency of our estimator.

Theorem 3. *Suppose that $(\boldsymbol{N_t})$ is observed on $\mathbb{R}^+$, $\widehat{W}_T \xrightarrow{\mathbb{P}} W$, and*

1. *W is positive semi-definite and $Wg_0(\theta) = 0$ if and only if $\theta = \theta_0$,*
2. *$\theta \in \Theta$, which is compact,*

[6] When $f^T_t = \mathbb{1}_{[-H_T, H_T]}(t)$, we remind that $(f^T \star \widetilde{f}^T)_t = (2H_T - |t|)^+$. This leads to the estimator we showed in the article.

3. *the spectral radius of the kernel norm matrix satisfies* $||\mathbf{\Phi}||_* < 1$,

4. $\forall i,j,k \in [d], \int f_u^T C_u^{ij} du \to \int C_u^{ij} du$ *and* $\int f_u^T f_v^T K_{u,v}^{ijk} du dv \to \int K_{u,v}^{ijk} du dv$,

5. $(F^T)^2/T \xrightarrow{\mathbb{P}} 0$ *and* $||f||_\infty = O(1)$.

Then

$$\widehat{\theta}_T \xrightarrow{\mathbb{P}} \theta_0.$$

Remark 1. *In practice, we use a constant sequence of weighting matrices:* $\widehat{W}_T = \boldsymbol{I_d}$.

Proof. Proceed by verifying the hypotheses of Theorem 2.1 from [NM94]. Condition 2.1(i) follows by (i) and by $Q_0(\theta) = [W^{1/2} g_0(\theta)]^\top [W^{1/2} g_0(\theta)] > 0 = Q_0(\theta_0)$. Indeed, there exists a neighborhood N of θ_0 such that $\theta \in N \backslash \{\theta_0\}$ and $g_0(\theta) \neq 0$ since $g_0(\theta)$ is a polynom. Condition 2.1(ii) follows by (ii). Condition 2.1(iii) is satisfied since $Q_0(\theta)$ is a polynom. Condition 2.1(iv) is harder to prove. First, since $\widehat{g}_T(\theta)$ is a polynom of θ, we prove easily that $\mathbb{E}[\sup_{\theta\in\Theta} |\widehat{g}_T(\theta)|] < \infty$. Then, by Θ compact, $g_0(\theta)$ is bounded on Θ, and by the triangle and Cauchy-Schwarz inequalities,

$$\begin{aligned}
&|\widehat{Q}_T(\theta) - Q_0(\theta)| \\
&\quad \leq |(\widehat{g}_T(\theta) - g_0(\theta))^\top \widehat{W}_T (\widehat{g}_T(\theta) - g_0(\theta))| \\
&\quad + |g_0(\theta)^\top (\widehat{W}_T + \widehat{W}_T^\top)(\widehat{g}_T(\theta) - g_0(\theta))| + |g_0(\theta)^\top (\widehat{W}_T - W) g_0(\theta)| \\
&\quad \leq \|\widehat{g}_T(\theta) - g_0(\theta)\|^2 \|\widehat{W}_T\| + 2\|g_0(\theta)\| \|\widehat{g}_T(\theta) - g_0(\theta)\| \|\widehat{W}_T\| + \|g_0(\theta)\|^2 \|\widehat{W}_T - W\|.
\end{aligned}$$

To prove $\sup_{\theta\in\Theta} |\widehat{Q}_T(\theta) - Q_0(\theta)| \xrightarrow{\mathbb{P}} 0$, we should now prove that $\sup_{\theta\in\Theta} \|\widehat{g}_T(\theta) - g_0(\theta)\| \xrightarrow{\mathbb{P}} 0$. By Θ compact, it is sufficient to prove that $\|\widehat{\boldsymbol{L}} - \boldsymbol{L}\| \xrightarrow{\mathbb{P}} 0$, $\|\widehat{\boldsymbol{C}}^{(T)} - \boldsymbol{C}\| \xrightarrow{\mathbb{P}} 0$, and $\|\widehat{\boldsymbol{K^c}}^{(T)} - \boldsymbol{K^c}\| \xrightarrow{\mathbb{P}} 0$.

Proof that $\|\widehat{\boldsymbol{L}} - \boldsymbol{L}\| \xrightarrow{\mathbb{P}} 0$

The estimator of $\boldsymbol{L}$ is unbiased so let's focus on the variance of $\widehat{\boldsymbol{L}}$.

$$\begin{aligned}
\mathbb{E}[(\widehat{\Lambda}^i - \Lambda^i)^2] &= \mathbb{E}\left[\left(\frac{1}{T}\int_{H_T}^{T+H_T} (dN_t^i - \Lambda^i dt)\right)^2\right] \\
&= \frac{1}{T^2}\int_{H_T}^{T+H_T}\int_{H_T}^{T+H_T} \mathbb{E}[(dN_t^i - \Lambda^i dt)(dN_{t'}^i - \Lambda^i dt')] \\
&= \frac{1}{T^2}\int_{H_T}^{T+H_T}\int_{H_T}^{T+H_T} C_{t'-t}^{ii} dt dt' \\
&\leq \frac{1}{T^2}\int_{H_T}^{T+H_T} C^{ii} dt = \frac{C^{ii}}{T} \longrightarrow 0
\end{aligned}$$

By Markov inequality, we have just proved that $\|\widehat{\boldsymbol{L}} - \boldsymbol{L}\| \xrightarrow{\mathbb{P}} 0$.

Proof that $\|\widehat{\boldsymbol{C}}^{(T)} - \boldsymbol{C}\| \xrightarrow{\mathbb{P}} 0$

First, let's remind that $\mathbb{E}(\widehat{\boldsymbol{C}}^{(T)}) \neq \boldsymbol{C}$. Indeed,

$$\begin{aligned}
\mathbb{E}\left(\widehat{C}^{ij,(T)}\right) &= \mathbb{E}\left(\frac{1}{T}\int_{H_T}^{T+H_T} dN_t^i \int_0^{T+2H_T} dN_{t'}^j f_{t'-t} - \widehat{\Lambda}^i\widetilde{\Lambda}^j F^T\right) \\
&= \mathbb{E}\left(\frac{1}{T}\int_{H_T}^{T+H_T} dN_t^i \int_{-t}^{T+2H_T-t} dN_{t+s}^j f_s - \Lambda^i\Lambda^j F^T\right) + \epsilon^{ij,T,H_T}F^T \\
&= \frac{1}{T}\int_{H_T}^{T+H_T}\int_{-H_T}^{H_T} f_s \mathbb{E}\left(dN_t^i dN_{t+s}^j - \Lambda^i\Lambda^j ds\right) + \epsilon^{ij,T,H_T}F^T \\
&= \int f_s C_s^{ij} ds + \epsilon^{ij,T,H_T}F^T
\end{aligned}$$

Now,

$$\begin{aligned}
\epsilon^{ij,T,H_T} &= \mathbb{E}\left(\Lambda^i\Lambda^j - \widehat{\Lambda}^i\widetilde{\Lambda}^j\right) \\
&= -\frac{1}{T^2}\int_{H_T}^{T+H_T}\int_0^{T+2H_T} \mathbb{E}\left(dN_t^i dN_{t'}^j - \Lambda^i\Lambda^j dt dt'\right) \\
&= -\frac{1}{T^2}\int_{H_T}^{T+H_T}\int_0^{T+2H_T} C_{t-t'}^{ij} dt dt' \\
&= -\frac{1}{T}\int \left(1 + \left(\frac{H_T - |t|}{T}\right)^-\right)^+ C_t^{ij} dt
\end{aligned}$$

Since f satisfies $F^T = o(T)$, we have $\mathbb{E}(\widehat{\boldsymbol{C}}^{(T)}) \longrightarrow \boldsymbol{C}$. It remains now to prove that $\|\widehat{\boldsymbol{C}}^{(T)} - \mathbb{E}(\widehat{\boldsymbol{C}}^{(T)})\| \xrightarrow{\mathbb{P}} 0$.
Let's now focus on the variance of $\widehat{C}^{ij,(T)}$: $\mathbb{V}(\widehat{C}^{ij,(T)}) = \mathbb{E}\left((\widehat{C}^{ij,(T)})^2\right) - \mathbb{E}(\widehat{C}^{ij,(T)})^2$.
Now,

$$\begin{aligned}
&\mathbb{E}\left((\widehat{C}^{ij,(T)})^2\right) \\
&= \mathbb{E}\left(\frac{1}{T^2}\sum_{(\tau,\eta,\tau',\eta')\in(Z^{i,T,1})^2\times(Z^{j,T,2})^2} (f_{\tau'-\tau} - F^T/(T+2H_T))(f_{\eta'-\eta} - F^T/(T+2H_T))\right) \\
&= \mathbb{E}\left(\frac{1}{T^2}\int_{t,s\in[H_T,T+H_T]}\int_{t',s'} dN_t^i dN_{t'}^j dN_s^i dN_{s'}^j (f_{t'-t} - F^T/(T+2H_T))(f_{s'-s} - F^T/(T+2H_T))\right) \\
&= \frac{1}{T^2}\int_{t,s\in[H_T,T+H_T]}\int_{t',s'\in[0,T+2H_T]} \mathbb{E}\left(dN_t^i dN_{t'}^j dN_s^i dN_{s'}^j\right) \\
&\qquad\qquad \cdot (f_{t'-t} - F^T/(T+2H_T))(f_{s'-s} - F^T/(T+2H_T))
\end{aligned}$$

And,

$$\begin{aligned}
&\mathbb{E}(\widehat{C}^{ij,(T)})^2 \\
&= \frac{1}{T^2}\int_{t,s\in[H_T,T+H_T]}\int_{t',s'\in[0,T+2H_T]} \mathbb{E}\left(dN_t^i dN_{t'}^j\right)\mathbb{E}\left(dN_s^i dN_{s'}^j\right) \\
&\qquad\qquad \cdot (f_{t'-t} - F^T/(T+2H_T))(f_{s'-s} - F^T/(T+2H_T))
\end{aligned}$$

Then, the variance involves the integration towards the difference of moments $\mu^{r,s,t,u} - \mu^{r,s}\mu^{t,u}$.
Let's write it as a sum of cumulants, since cumulants density are integrable.

$$\begin{aligned}\mu^{r,s,t,u} - \mu^{r,s}\mu^{t,u} &= \kappa^{r,s,t,u} + \kappa^{r,s,t}\kappa^{u}[4] + \kappa^{r,s}\kappa^{t,u}[3] + \kappa^{r,s}\kappa^{t}\kappa^{u}[6] + \kappa^{r}\kappa^{s}\kappa^{t}\kappa^{u} - (\kappa^{r,s} + \kappa^{r}\kappa^{s})(\kappa^{t,u} + \kappa^{t}\kappa^{u}) \\ &= \kappa^{r,s,t,u} \\ &+ \kappa^{r,s,t}\kappa^{u} + \kappa^{u,r,s}\kappa^{t} + \kappa^{t,u,r}\kappa^{s} + \kappa^{s,t,u}\kappa^{r} \\ &+ \kappa^{r,t}\kappa^{s,u} + \kappa^{r,u}\kappa^{s,t} \\ &+ \kappa^{r,t}\kappa^{s}\kappa^{u} + \kappa^{r,u}\kappa^{s}\kappa^{t} + \kappa^{s,t}\kappa^{r}\kappa^{u} + \kappa^{s,t}\kappa^{r}\kappa^{u}\end{aligned}$$

In the rest of the proof, we denote $a_t = \mathbb{1}_{t\in[H_T, T+H_T]}$, $b_t = \mathbb{1}_{t\in[0,T+2H_T]}$, $c_t = \mathbb{1}_{t\in[-H_T,H_T]}$, $g_t = f_t - \frac{1}{T+2H_T}F^T$
Before starting the integration of each term, let's remark that:

1. $\mathbf{\Psi}_t = \sum_{n\geq 1}\mathbf{\Phi}_t^{(\star n)} \geq 0$ since $\mathbf{\Phi}_t \geq 0$.
2. The regular parts of C_u^{ij}, $K_{u,v}^{ijk}$ (skewness density) and $M_{u,v,w}^{ijkl}$ (fourth cumulant density) are positive as polynoms of integrals of $\psi^{ab}_{\cdot}$ with positive coefficients. The integrals of the singular parts are positive as well.
3. a) $\int a_t b_{t'} f_{t'-t} dt dt' = TF^T$
   b) $\int a_t b_{t'} g_{t'-t} dt dt' = 0$
   c) $\int a_t b_{t'} |g_{t'-t}| dt dt' \leq 2TF^T$
4. $\forall t \in \mathbb{R}, a_t(b \star \widetilde{g})_t = 0$, where $\widetilde{g}_s = g_{-s}$.

Fourth cumulant We want here to compute $\int \kappa^{i,j,i,j}_{t,t',s,s'} a_t b_{t'} a_s b_{s'} g_{t'-t} g_{s'-s} dt dt' ds ds'$.
We remark that $|g_{t'-t}g_{s'-s}| \leq (||f||_\infty(1+2H_T/T))^2 \leq 4||f||_\infty^2$.

$$\begin{aligned}\left|\frac{1}{T^2}\int \kappa^{i,j,i,j}_{t,t',s,s'} a_t b_{t'} a_s b_{s'} g_{t'-t} g_{s'-s} dt dt' ds ds'\right| &\leq \left(\frac{2||f||_\infty}{T}\right)^2 \int dt a_t \int dt' b_{t'} \int ds a_s \int ds' b_{s'} M^{ijij}_{t'-t,s-t,s'-t} \\ &\leq \left(\frac{2||f||_\infty}{T}\right)^2 \int dt a_t \int dt' b_{t'} \int ds a_s \int dw M^{ijij}_{t'-t,s-t,w} \\ &\leq \left(\frac{2||f||_\infty}{T}\right)^2 \int dt a_t \int M^{ijij}_{u,v,w} du dv dw \\ &\leq \frac{4||f||_\infty^2}{T} M^{ijij} \underset{T\to\infty}{\longrightarrow} 0\end{aligned}$$

Third × First We have four terms, but only two different forms since the roles of (s,s') and (t,t') are symmetric.
First form

$$\begin{aligned}\int \kappa^{i,j,i}_{t,t',s} \Lambda^j G_{\boldsymbol{t}} d\boldsymbol{t} &= \frac{\Lambda^j}{T^2}\int \kappa^{i,j,i}_{t,t',s} a_t b_{t'} a_s b_{s'} g_{t'-t} g_{s'-s} dt dt' ds ds' \\ &= \frac{\Lambda^j}{T^2}\int \kappa^{i,j,i}_{t,t',s} a_t b_{t'} a_s (b\star\widetilde{g})_s g_{t'-t} dt dt' ds \\ &= 0 \qquad \text{since } a_s(b\star\widetilde{g})_s = 0\end{aligned}$$

Second form

$$\begin{aligned}\left|\int \kappa^{i,j,j}_{t,t',s'}\Lambda^i G_{\boldsymbol{t}} d\boldsymbol{t}\right| &= \left|\frac{\Lambda^i}{T^2}\int \kappa^{i,j,j}_{t,t',s'} a_t b_{t'} a_s b_{s'} g_{t'-t} g_{s'-s} dt dt' ds ds'\right| \\ &= \left|\frac{\Lambda^i}{T^2}\int \kappa^{i,j,j}_{t,t',s'} a_t b_{t'} g_{t'-t} b_{s'} (a \star g)_{s'} dt dt' ds'\right| \\ &\leq \frac{\Lambda^i}{T^2} 2||f||_\infty \int ds' b_{s'} (a \star |g|)_{s'} \int dt a_t \int dt' b_{t'} K^{ijj}_{t'-s',t-s'} \\ &\leq 4||f||_\infty K^{ijj}\Lambda^i \frac{F^T}{T} \underset{T\to\infty}{\longrightarrow} 0\end{aligned}$$

Second × Second
First form

$$\begin{aligned}\left|\int \kappa^{i,i}_{t,s}\kappa^{j,j}_{t',s'} G_{\boldsymbol{t}} d\boldsymbol{t}\right| &\leq \frac{2||f||_\infty}{T^2}\int C^{ii}_{t-s} C^{jj}_{t'-s'} a_t b_{t'} |g_{t'-t}| a_s b_{s'} dt dt' ds ds' \\ &\leq \frac{2||f||_\infty}{T^2} C^{ii} C^{jj} \int a_t b_{t'} |g_{t'-t}| dt dt' \\ &\leq 4||f||_\infty C^{ii} C^{jj} \frac{F^T}{T} \underset{T\to\infty}{\longrightarrow} 0\end{aligned}$$

Second form

$$\left|\int \kappa^{i,j}_{t,s'}\kappa^{i,j}_{t',s} G_{\boldsymbol{t}} d\boldsymbol{t}\right| \leq 4||f||_\infty (C^{ij})^2 \frac{F^T}{T} \underset{T\to\infty}{\longrightarrow} 0$$

Second × First × First
First form

$$\int \kappa^{i,j}_{t,t'}\Lambda^i\Lambda^j G_{\boldsymbol{t}} d\boldsymbol{t} = \frac{\Lambda^i\Lambda^j}{T^2}\int \kappa^{i,j}_{t,t'} a_t b_{t'} g_{t'-t} dt dt' \int a_s b_{s'} g_{s'-s} ds ds' = 0$$

Second form

$$\int \kappa^{i,i}_{t,s}\Lambda^j\Lambda^j G_{\boldsymbol{t}} d\boldsymbol{t} = \left(\frac{\Lambda^j}{T}\right)^2 \int \kappa^{i,i}_{t,s} a_t b_{t'} g_{t'-t} a_s (b \star \widetilde{g})_s dt dt' ds = 0$$

We have just proved that $\mathbb{V}(\widehat{\boldsymbol{C}}^{(T)}) \xrightarrow{\mathbb{P}} 0$. By Markov inequality, it ensures us that $\|\widehat{\boldsymbol{C}}^{(T)} - \mathbb{E}(\widehat{\boldsymbol{C}}^{(T)})\| \xrightarrow{\mathbb{P}} 0$, and finally that $\|\widehat{\boldsymbol{C}}^{(T)} - \boldsymbol{C}\| \xrightarrow{\mathbb{P}} 0$. ■

Proof that $\|\widehat{\boldsymbol{K}^{\boldsymbol{c}}}^{(T)} - \boldsymbol{K}^{\boldsymbol{c}}\| \xrightarrow{\mathbb{P}} 0$

The scheme of the proof is similar to the previous one. The upper bounds of the integrals involve the same kind of terms, plus the new term $(F^T)^2/T$ that goes to zero thanks to the assumption 5 of the theorem.

5 Conclusion

In this paper, we introduce a simple nonparametric method (the NPHC algorithm) that leads to a fast and robust estimation of the matrix $\boldsymbol{G}$ of the kernel integrals of a Multivariate Hawkes process that encodes Granger causality between nodes. This method relies on the matching of the integrated order 2 and order 3 empirical cumulants, which represent the simplest set of global observables containing sufficient information to recover the matrix $\boldsymbol{G}$. Since this matrix fully accounts for the self- and cross- influences of the process nodes (that can represent agents or users in applications), our approach can naturally be used to quantify the degree of endogeneity of a system and to uncover the causality structure of a network.

By performing numerical experiments involving very different kernel shapes, we show that the baselines, involving either parametric or non-parametric approaches are very sensible to model misspecification, do not lead to accurate estimation, and are numerically expensive, while NPHC provides fast, robust and reliable results. This is confirmed on the MemeTracker database, where we show that NPHC outperforms classical approaches based on EM algorithms or the Wiener-Hopf equations. Finally, the NPHC algorithm provided very satisfying results on financial data, that are consistent with well-known stylized facts in finance.

Acknowledgements

This work benefited from the support of the chair “Changing markets”, CMAP École Polytechnique and École Polytechnique fund raising - Data Science Initiative.

The authors want to thank Marcello Rambaldi for fruitful discussions on order book data’s experiments.

CHAPTER IV

Constrained optimization approach

Abstract

We want to estimate the Hawkes causality matrix *i.e.* the matrix of Hawkes kernels' integral from the mean, the integrated covariance and the minimization of a criterion. We show in this document the inference can be naturally turned into ADMM problem form.

Keywords: Hawkes processes, convex relaxation, ADMM, compressed sensing.

1 Introduction

The previous approach based on the Generalized Method of Moments need the first three cumulants to obtain enough information from the data to recover the d^2 entries of $\boldsymbol{G}$. Indeed, we want to recover d^2 independent coefficients - the entries of $\boldsymbol{G}$ - and the first two integrated cumulants give $d + d(d+1)/2$ independent terms since the integrated covariance $\boldsymbol{C}$ is a symmetric matrix. Assuming the matrix $\boldsymbol{G}$ has a certain structure, we can get rid of the third order cumulant and design another estimation procedure using only the first two integrated cumulants. The advantage of such approach lies in the convexity of the related optimization problem, on the contrary to the minimization of $\mathscr{L}_T$ from the previous chapter. The matrix we want to estimate minimize a simple criterion f convex, typically a norm, while being consistent with the first two empirical integrated cumulants.

2 Problem setting

We start from the relation between the integrated covariance $\boldsymbol{C}$ and the matrix $\boldsymbol{R}$ introduced in the previous chapter, from [JHR15] and many other references:

$$\boldsymbol{C} = \boldsymbol{R}\boldsymbol{L}\boldsymbol{R}^\top.$$

Our purpose is still to approximate $\boldsymbol{G} = \boldsymbol{I} - \boldsymbol{R}^{-1}$ from the information encoded in the integrated cumulants. The previous equation in $\boldsymbol{R}$ admits a set of roots of that is isomorphocic to orthogonal group $O_n(\mathbb{R})$, and then:

$$\boldsymbol{G} = \boldsymbol{I} - \boldsymbol{L}^{1/2}\boldsymbol{M}\boldsymbol{C}^{-1/2} \quad \text{with} \quad \boldsymbol{M} \in O_n(\mathbb{R}) \qquad \textit{i.e.} \qquad \boldsymbol{L}^{-1/2}(\boldsymbol{I} - \boldsymbol{G})\boldsymbol{C}^{1/2} \in O_n(\mathbb{R}). \tag{1}$$

The previous expression only comes from the relation on the covariance. However, two classic assuptions on the Hawkes kernel norm matrix are not yet encoded. The first one concerns the positivity of the kernels, and then the positivity of their integrals: $g^{ij} \geq 0$ for $i, j \in [d]$. Some variants of Hawkes processes allow the possibility of modeling inhibition through negative valued kernels [PSCR11], with nonlinear Hawkes processes for instance [BM96], but the closed formulas of the cumulants [JHR15] no longer stand with those variants. The other well-known assumption is linked to the stationarity of the process. The counting process $\boldsymbol{N}_t$ has asymptotically stationary increments if the spectral norm of the kernel norm matrix is smaller than one: $||\boldsymbol{G}|| < 1$.

We finally encode the structure of the matrix $\boldsymbol{G}$ via the minimization of a criterion f subject to some constraints. For the problem to be easy to solve, the criterion f will be a convex function whose proximal operator is explicit. The fit to the data encoded in Equation (1), and the two assumptions above will be regarded as constraints of our optimization problem.

All together, we formulate our problem as the following constrained optimization problem *Constrained optimization* problem:

$$\begin{aligned} \min_{\boldsymbol{G}} \quad & f(\boldsymbol{G}) \\ \text{s.t.} \quad & \boldsymbol{L}^{-1/2}(\boldsymbol{I} - \boldsymbol{G})\boldsymbol{C}^{1/2} \in O_n(\mathbb{R}) \\ & ||\boldsymbol{G}|| < 1 \\ & g^{ij} \geq 0 \end{aligned}$$

The problem above involves easy constraints on two different matrices: $\boldsymbol{G}$ and $\boldsymbol{M} = \boldsymbol{L}^{-1/2}(\boldsymbol{I} - \boldsymbol{G})\boldsymbol{C}^{1/2}$. Our first idea is to relax the previous problem to turn it into a convex optimization problem.

The objective f is convex, and the constraints $||\boldsymbol{G}|| < 1$ and $g^{ij} \geq 0$ correspond to convex sets. The constraint that involves the orthogonal group is trickier and is not classic. We prove in Section 6 that the convex hull of the orthogonal group $O_n(\mathbb{R})$ is the closed unit ball w.r.t. the ℓ_2 norm. In the rest of the chapter, we denote $\mathscr{B}$ (resp. $\overline{\mathscr{B}_2}$) the open (resp. closed) unit ball w.r.t the spectral norm (resp. the ℓ_2 norm).

Instead of the previous problem, we split the variables $\boldsymbol{G}$ and $\boldsymbol{M}$, meaning that we focus on the minimization problem both on $\boldsymbol{G}$ and $\boldsymbol{M}$. Such minimization problem on two variables x and z linked via an equation of the form $Ax + Bz = c$ can be efficiently solved with the *Alternating Direction Method of Multipliers* algorithm [GM75, GM76] detailed in Section 3.

The minimization problem we finally aim at solving writes:

$$\begin{aligned} \min_{\boldsymbol{G},\boldsymbol{M}} \quad & f(\boldsymbol{G}) + \mathbb{1}_{\overline{\mathscr{B}}_2}(\boldsymbol{M}) + \mathbb{1}_{\mathscr{B}}(\boldsymbol{G}) + \mathbb{1}_{\mathbb{R}_+^{d\times d}}(\boldsymbol{G}) \\ \text{s.t.} \quad & \boldsymbol{L}^{-1/2}\boldsymbol{G} + \boldsymbol{M}\,\boldsymbol{C}^{-1/2} = \boldsymbol{L}^{-1/2}, \end{aligned} \tag{2}$$

On the contrary to the optimization problem of the previous chapter, the problem just stated is convex. We test this procedure on numerical simulations of various Hawkes kernels and real order book data, and we show how the criterion f impact the matrices we retrieve.

3 ADMM

3.1 The ADMM algorithm

The *Alternating Direction Methods of Multipliers* (ADMM) is a widely-used minimization method to solve constrained problems of the form

$$\begin{aligned} \min_{x,z} \quad & f(x) + g(z) \\ \text{s.t.} \quad & Ax + Bz = c. \end{aligned} \tag{3}$$

The objective function is separable in (x, z) with g and h two convex functions. The constraint involves two matrices A and B, and a constant vector c. The algorithm ADMM was originally introduced in [GM76] and [GM75], and focuses on the augmented Lagrangian [Hes69, Pow67] associated to problem (3), that is:

$$L_\rho(x, z, y) := g(x) + h(z) + y^\top (Ax + Bz - c) + \frac{\rho}{2}||Ax + Bz - c||_2^2, \tag{4}$$

with $\rho > 0$ and solves the problem

$$\min_{x,z} \max_{y} \quad L_\rho(x, z, y) \tag{5}$$

instead of the initial one. The *method of multipliers* [Hes69, Pow67] (analysis in [Ber14]) applied to this problem would alternate an exact minimization step on the primal variable (x, z) and a gradient ascent step on the dual variable y. Instead of the exact minimization step on the couple (x, z), we do one pass of a Gauss-Seidel method [GVL12] and split the joint minimization into two partial minimization steps: one over x with z fixed, the other over z with x fixed. These two minimization steps can be done simultaneously, from the same initial points, or in the case of ADMM, one after the other, with an update between. Namely, ADMM algorithm iterates the following update steps:

$$\begin{aligned} x^{t+1} &= \arg\min_x L_\rho(x, z^t, y^t), \\ z^{t+1} &= \arg\min_z L_\rho(x^{t+1}, z, y_t), \\ y^{t+1} &= y^t + \rho(Ax^{t+1} + Bz^{t+1} - c). \end{aligned}$$

3.2 Convergence results

The convergence results of ADMM hold under the following two assumptions:

- The functions f and g are convex, proper[1] and closed[2].
- The (unaugmented) Lagrangian L_0 has a saddle point *i.e.* there exist (x^*, z^*, y^*) for which
$$L_0(x^*, z^*, y) \leq L_0(x^*, z^*, y^*) \leq L_0(x, z, y^*) \qquad \text{for all } x, y, z.$$

Under these two assumptions, the ADMM iterates satisfy the following convergences (a proof is given in [BPC+11]):

- *Residual convergence*: $r^t = Ax^t + Bz^t - c \to 0$ as $t \to \infty$ *i.e.* the iterates approaches feasibility.
- *Objective convergence*: $f(x^t) + g(z^t) \to \min_{x,z}\{f(x) + g(z)\}$ as $t \to \infty$ *i.e.* the objective function approaches its optimal value.
- *Dual variable convergence*: $y^t \to y^*$ as $t \to \infty$, where y^* is a dual optimal point.

3.3 Examples

The ADMM method is quite general and plenty of optimization problems can be solved with it. We show here two usual tricks to turn an optimization problem into a relevant ADMM form. The first is to introduce indicator functions and concerns for instance optimization problem constrained on a set $\mathscr{C}$:

$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & x \in \mathscr{C}. \end{aligned}$$

This problem can be equivalenty written:

$$\begin{aligned} \min_{x,z} \quad & f(x) + g(z) \\ \text{s.t.} \quad & x - z = 0 \end{aligned}$$

with g to be indicator of $\mathscr{C}$ *i.e.* to equal zero on $\mathscr{C}$ and ∞ outside.

The other trick is to introduce a variable z being equal to a linear transformation of x. We consider the problem called *total variation denoising* [ROF92]:

$$\min_x \quad ||x - b||_2^2 + \lambda \sum_{i=1}^{d-1} |x_{i+1} - x_i|.$$

[1] A proper convex function f is a convex function taking values on the extended real line such that $f(x) > -\infty$ for all x and $f(x) < +\infty$ for at least one x.

[2] A proper convex function is closed if and only if it is lower semi-continuous.

Denoting $F = (F_{ij})$ with $F_{ij} = \mathbb{1}_{j=i+1} - \mathbb{1}_{j=i}$, the previous problem can be written as:

$$\begin{aligned} \min_{x,z} \quad & ||x-b||_2^2 + \lambda ||z||_1 \\ \text{s.t.} \quad & Fx - z = 0. \end{aligned}$$

Such problem can be efficiently solved using the ADMM algorithm, since each update step of the algorithm has a closed form using proximal operators of the ℓ_2 and ℓ_1 norms.

4 Numerical results

In the previous sections, we only assumed f was a convex criterion whose proximal operator can be easily computed. Now, we exhibit three different choices for f and present the results obtained with these choices for both simulated and real-world dataset. The criteria we consider are the ℓ_1-norm $||\cdot||_1$, the squared ℓ_2-norm $||\cdot||_2^2$ and the nuclear norm $||\cdot||_*$. In the rest of the section, we refer as *Problem I* the minimization problem written in Equation (2) with $f = ||\cdot||_1$, *Problem II* when $f = ||\cdot||_2^2$ and *Problem III* for the case $f = ||\cdot||_*$. We solve those minimization problem using the ADMM algorithm whose update steps are written above. The explicit update steps are provided in Section 6.

4.1 Simulated data

We simulated multivariate Hawkes point processes with the procedure already explained in the previous chapter, and implemented in the open-source library tick. As previously, we simulated three datasets generated from three different Hawkes kernels: the exponential kernel, the power-law kernel and the rectangular one. The mean vector and the integrated covariance matrix are computed using estimators provided in the previous chapter. We then used ADMM algorithm to solve the problems *I*, *II* and *III*, and observed the same patterns for the three kernels. To ease the reading, we only show the results for the exponential kernel in dimension 100. The results from the Figure IV.1 and the Table IV.1 are consistent and shows that the solution to *Problem I* is the closest to the ground-truth matrix $\boldsymbol{G}$. Moreover, according the Figure IV.1 one observes that the solutions to the two other problems are symmetric matrices, while the ground-truth matrix is not.

Table IV.1: The solution of *Problem I* has the smallest relative error.

Problem	*I*	*II*	*III*
RelErr	**0.093**	0.130	0.131

4.2 Order book data

The numerical experiments on simulated data incites us to focus on *Problem I* if the matrix $\boldsymbol{G}$ we want to uncover is not symmetric. Such non-symmetric relationships are for instance highlighted in [RBL17], where the authors studied the interplay between orders of different

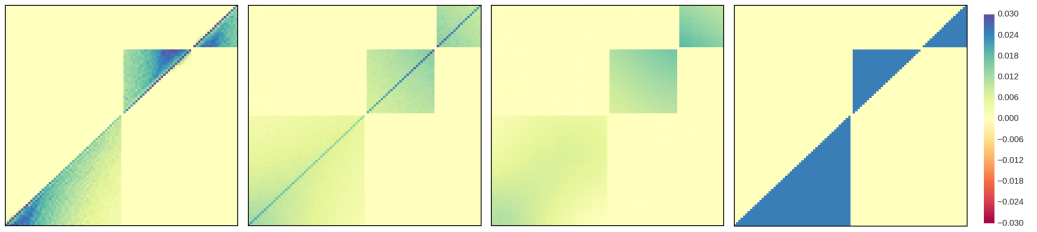


Figure IV.1: From the left to the right: solution of *Problem I*, solution of *Problem II*, solution of *Problem III*, and the ground-truth matrix $\boldsymbol{G}$. Only the solution to *Problem I* recovers the three blocks. The two other problems outputed symmetric matrices, while the ground-truth matrix is not.

sizes. Indeed, a large trade is more likely to be followed by smaller trades than the opposite.

We use the same data as the authors of [RBL17] *i.e.* high-frequency order book data of futures traded at EUREX, that are also used in the numerical part of the previous chapter, see this section for details about the dataset. Here, we use the trades' timestamps of Bund futures.

We consider here unsigned trades *i.e.* we do not distinguish between buyer initiated trades and seller initiated ones. The different dimensions of the multivariate point process correspond to different intervals of volumes: each transaction falls into only one component. We denote N_t^a the number of transactions whose volume equals a that occured before t, $N_t^{a:b}$ the number of transactions whose volume is between a and b (included), and $N_t^{a:}$ the number of transactions whose volume is greater or equal than a. We then consider the following multivariate point processes, and solve the *Problem I* for these timestamped events:

$$\boldsymbol{A}_t = (N_t^1, N_t^{2:3}, N_t^{4:20}, N_t^{21:}), \quad \boldsymbol{B}_t = (N_t^1, N_t^2, N_t^3, N_t^{4:7}, N_t^{8:20}, N_t^{21:}), \quad \boldsymbol{C}_t = (N_t^1, \dots, N_t^{20}, N_t^{21:}).$$

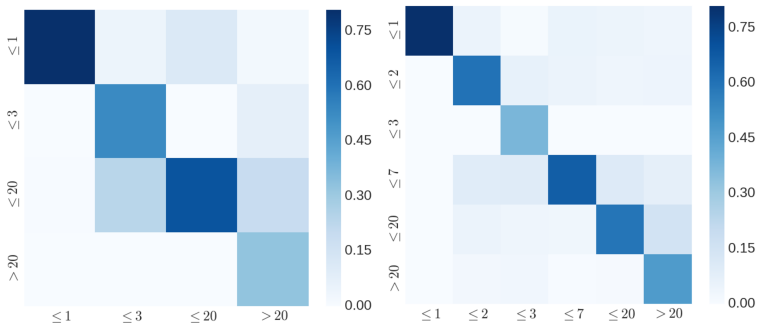


Figure IV.2: Solutions of *Problem I* for the multivariate point process $\boldsymbol{A}_t$ (left) and $\boldsymbol{B}_t$ (right). We observe a strong self-excitation. These solutions are consistent with the estimated kernel norm matrix in [RBL17].

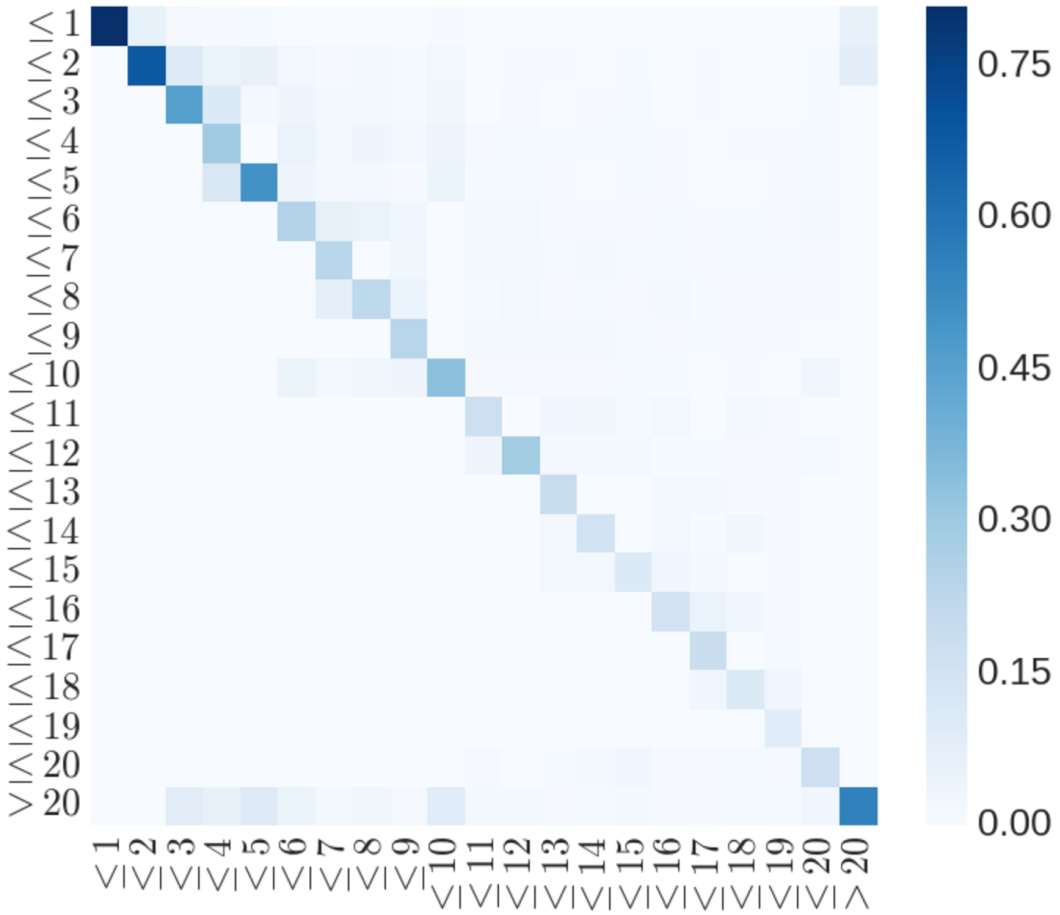


Figure IV.3: Solution of *Problem I* for the multivariate point process $\boldsymbol{C}_t$. This solution is consistent with estimtates in lower dimension.

The solutions we found share the same patterns. We observe that self-excitation is preponderant, followed in importance by the excitation from large volumes. The excitation from large volumes is however lighter when we increase the dimension, this may be a consequence of the ℓ_1 norm minimization which aims at finding sparse solutions. Our observations are consistent with the results obtained in [RBL17], see this reference for financial interpretations of the results. Note that kernel norm matrix of 21-dimensional model who have been very long to estimate with the Wieher-Hopf based method used in [RBL17], while our method has way lower complexity (comparable to the NPHC's one, see the previous chapter for a full comparison).

5 Conclusion

The approach consisting of minimizing a criterion instead of extracting the information from the third integrated cumulant seems promising. The convexity of the optimization problem is a real advantage compared to the non-convex problem one has to solve to estimate the Hawkes kernel norm matrix using NPHC, see Chapter III. The method developed in this chapter however lacks theory, especially for the choice of the criterion to minimize.

As shown in the numerical part, the solution to the *Problem I* seems to provide better solutions, compared to the two other problems. Such statement could be explained by the similarity between *Problem I* and the ℓ_1 minimization of the *compressing sensing* problem

$$\begin{aligned} \min_x \quad & ||x||_1 \\ \text{s.t.} \quad & Ax = b. \end{aligned}$$

One can indeed prove the exact recovery of the vector x under some assumptions [Don06].

6 Technical details

6.1 Convex hull of the orthogonal group

The convex hull of the orthogonal group is the unit ball for the ℓ_2 norm. This is a nice exercise that can be solved using simple tools of linear algebra. A proof can be found in [GGT04] for instance.

6.2 Updates of ADMM steps

6.2.1 Notations

We first denote the functions used in 2: $f_1(X) = f(X)$, $f_2(X) = \mathbb{1}_{\mathbb{R}_+^{d\times d}}(X)$, $f_3(X) = \mathbb{1}_{\mathscr{B}}(X)$ and $f_4(X) = \mathbb{1}_{\overline{\mathscr{B}}_2}(X)$. We also denote $A = C = \boldsymbol{L}^{-1/2}$ and $B = \boldsymbol{C}^{-1/2}$. After splitting to the right number of variables (so that the update steps of ADMM algorithm for problem 2 write with closed formula), the problem 2 becomes:

$$\begin{aligned} \min_{X_1,X_2,X_3,X_4,Y_1,Y_2} \quad & f_1(X_1) + f_2(X_2) + f_3(X_3) + f_4(X_4) \\ \text{s.t.} \quad & Y_1 + Y_2 = C \\ & X_1 - X_2 = 0 \\ & X_3 - X_2 = 0 \\ & A^{-1}Y_1 - X_1 = 0 \\ & Y_2 B^{-1} - X_4 = 0 \end{aligned}$$

6.2.2 Update steps

Now, the update steps of ADMM algorithm (using the scaled dual form, see [BPC+11]) write:

$$
\begin{aligned}
X_1^{t+1} &= \arg\min_{X_1} f_1(X_1) + (\rho/2)||X_1 - X_2^t + U_2^t||_F^2 + (\rho/2)||A^{-1}Y_1^t - X_1 + U_4^t||_F^2 \\
X_2^{t+1} &= \arg\min_{X_2} f_2(X_2) + (\rho/2)||X_1^{t+1} - X_2 + U_2^t||_F^2 + (\rho/2)||X_3^t - X_2 + U_3^t||_F^2 \\
X_3^{t+1} &= \arg\min_{X_3} f_3(X_3) + (\rho/2)||X_3 - X_2^{t+1} + U_3^t||_F^2 \\
X_4^{t+1} &= \arg\min_{X_4} f_4(X_4) + (\rho/2)||Y_2^t B^{-1} - X_4 + U_5^t||_F^2 \\
Y_1^{t+1} &= \arg\min_{Y_1} ||Y_1 + Y_2^t - C + U_1^t||_F^2 + ||A^{-1}Y_1 - X_1^{t+1} + U_4^t||_F^2 \\
Y_2^{t+1} &= \arg\min_{Y_2} ||Y_1^{t+1} + Y_2 - C + U_1^t||_F^2 + ||Y_2 B^{-1} - X_4^{t+1} + U_5^t||_F^2 \\
U_1^{t+1} &= U_1^t + (Y_1^{t+1} + Y_2^{t+1} - C) \\
U_2^{t+1} &= U_2^t + (X_1^{t+1} - X_2^{t+1}) \\
U_3^{t+1} &= U_3^t + (X_3^{t+1} - X_2^{t+1}) \\
U_4^{t+1} &= U_4^t + (A^{-1}Y_1^{t+1} - X_1^{t+1}) \\
U_5^{t+1} &= U_5^t + (Y_2^{t+1}B^{-1} - X_4^{t+1})
\end{aligned}
$$

6.2.3 Proximal operators

The previous update steps can be written using proximal operators of the functions $f_1, f_2, f_3 and f_4$.

Proximal operator of f_2 This one is straightforward:

$$\mathbf{prox}_{f_2}(X) = (X)_+ = (\max(x_{ij}, 0))_{ij}$$

Proximal operator of f_3 Using techniques given in [BV04], one easily shows that First, it is easier to compute the proximal operator of $f_3^\alpha = \mathbb{1}_{\sigma_1(\cdot)\leq\alpha}$ for $\alpha < 1$. Let's SVD some matrix $X \in \mathbb{R}^{d\times d}$: $X = USV^\top$ where $U, V \in O_d(\mathbb{R})$ and $S = \mathrm{diag}(\sigma_1, \ldots, \sigma_d)$ is diagonal. Using techniques given in [BV04], one easily shows that

$$\mathbf{prox}_{f_3^\alpha}(X) = \sum_{i=1}^{d} (\sigma_i - (\sigma_i - \alpha)_+)\, u_i v_i^\top$$

Proximal operator of f_4 This one is a well-known projection too:

$$\mathbf{prox}_{f_4}(X) = X\mathbb{1}_{\{||X||_2\leq 1\}} + \frac{X}{||X||_2}\mathbb{1}_{\{||X||_2>1\}}.$$

6.2.4 Final algorithm

$$
\begin{aligned}
X_1^{t+1} &= \mathbf{prox}_{f_1/(2\rho)}\left((X_2^t - U_2^t + A^{-1}Y_1^t + U_4^t)/2\right) \\
X_2^{t+1} &= (1/2)\left(X_1^{t+1} + U_2^t + X_3^t + U_3^t\right)_+ \\
X_3^{t+1} &= \mathbf{prox}_{f_3^\alpha}\left(X_2^{t+1} - U_3^t\right) \\
X_4^{t+1} &= \mathbf{prox}_{f_4}\left(Y_2^t B^{-1} + U_5^t\right) \\
Y_1^{t+1} &= (I_d + A^{-2})^{-1}(A^{-1}(X_1^{t+1} - U_4^t) - Y_2^t + C - U_1^t) \\
Y_2^{t+1} &= ((X_4^{t+1} - U_5^t)B^{-1} - Y_1^{t+1} + C - U_1^t)(I_d + B^{-2})^{-1} \\
U_1^{t+1} &= U_1^t + (Y_1^{t+1} + Y_2^{t+1} - C) \\
U_2^{t+1} &= U_2^t + (X_1^{t+1} - X_2^{t+1}) \\
U_3^{t+1} &= U_3^t + (X_3^{t+1} - X_2^{t+1}) \\
U_4^{t+1} &= U_4^t + (A^{-1}Y_1^{t+1} - X_1^{t+1}) \\
U_5^{t+1} &= U_5^t + (Y_2^{t+1}B^{-1} - X_4^{t+1})
\end{aligned}
$$

Part III

Capture order book dynamics with Hawkes processes

CHAPTER V

Analysis of order book dynamics using a nonparametric estimation of the branching ratio matrix

Abstract

We introduce a new non parametric method that allows for a direct, fast and efficient estimation of the matrix of kernel norms of a multivariate Hawkes process, also called branching ratio matrix. We demonstrate the capabilities of this method by applying it to high-frequency order book data from the EUREX exchange. We show that it is able to uncover (or recover) various relationships between all the first level order book events associated with some asset when mapped to a 12-dimensional process. We then scale up the model so as to account for events on two assets simultaneously and we discuss the joint high-frequency dynamics.
Keywords. Hawkes Process, Non-parametric estimation, GMM method, Order books, Market Microstructure

1 Introduction

With the large number of empirical studies devoted to high frequency finance, relying on datasets of increasing size and quality, many progresses have been made during the last decade in the modelling and understanding the microstructure of financial markets. Within this context, as evidenced by this special issue, Hawkes processes have become a very popular class of models. The main reason is that they allow one to account for the mutual influence of various types of events in a simple and parsimonious way through a conditional intensity vector. Hawkes processes have been involved in many different problems of high frequency finance ranging from the simple description of the temporal occurrence of market orders or price changes ([Bow07, HB14, FS12]), to the complex modelling of the arrival rates of various kinds of events in a full order book model ([Lar07, Tok11, JA13]). We refer to [BMM15] for a recent review.

A multivariate Hawkes model of dimension d is characterized by a $d \times d$ matrix of kernels, whose elements $\phi^{ij}(t)$ account for the influence, after a lag t, of events of type j on the arrival

rate of events of type i. The challenging issue of the statistical estimation of the shape of these excitation kernels has been addressed by many authors and various solutions have been proposed whose performances (accuracy and computational complexity) strongly depend on the empirical situation one considers. Indeed, if non-parametric methods like e.g. the EM method ([LM11]), the Wiener-Hopf method ([BM14a, BM16, BJM16]) or the contrast function method ([RBRGTM14]) can be applied in low dimensional situations with a large number of events, one has to consider parametric penalized alternatives (like e.g., in [ZZS13, YZ13]) when one has to handle a system of very large dimension with a relative low number of observed events (as, e.g., when studying events associated with the node activities of some social networks).

As far as (ultra) high frequency finance is concerned, the overall number of events can be very large. These events occur in a very correlated manner (with long-range correlations) and the system dimensionality can vary from low to moderately high. In a series of recent papers, Bacry *et al.* have shown that the non parametric Wiener-Hopf method provides reliable estimations in order to describe, within a multivariate Hawkes model, various aspects of level-I order book fluctuations: the coupled dynamics of mid-price changes, market and limit order arrivals ([BM14a, BJM16]), the impact of market orders ([BILL15]) or the interplay between book orders of different sizes ([RBL17]). However, if one wants to account for systems of larger dimensionality by considering for instance a wider class of event types or the book events associated with a basket (e.g. a couple) of assets, then the Wiener-Hopf method (or any other similar non-parametric method) may reach its limits as respect to both computational cost and estimation accuracy. On the other hand, a parametric approach can lead to strong bias in the estimated influences between components.

For this reason, in the present paper, we propose to estimate Hawkes models of order book data using the faster and simpler non-parametric approach introduced in [ABG+17]. This method focuses only on the global properties of the Hawkes process. More precisely, it aims at estimating directly the matrix of the kernel norms (also called the *branching ratio matrix*) without paying attention to the precise shape of these kernels. As recalled in the next section, this matrix does not bring all the information about the process dynamics, but is sufficient to disentangle the complex interactions between various type of events and estimate the magnitude of their self- and cross- excitations. Moreover, it allows one to estimate the amplitude of fluctuations of endogenous origin as compared to those of exogenous sources. The method we propose can be considered as the multivariate extension of the approach pioneered by [HB14] that proposed to estimate the kernel norm of a one-dimensional Hawkes model directly from the integral of the empirical correlation function. Unfortunately their approach cannot be immediately extended to a multivariate framework because it does not bring a sufficient number of constraints as compared to the number of unknown parameters. The method of [ABG+17] circumvents this difficulty by taking into account the first three integrated cumulant tensors of Hawkes process.

The paper is organized as follows: in Section 2 we provide the main definitions and properties of multivariate Hawkes processes and we introduce the main notations we use all along the paper. The cumulant method of Achab et al. is described and illustrated in Section 3. In Section 4 we estimate the matrix of kernel of Hawkes models for level-I book

events associated with 4 differents very liquid assets, namely DAX, Euro-Stoxx, Bund and Bobl future contracts. We first consider the 8-dimensional model proposed in [BJM16] in order to compare our method to the former results obtained with a computationally more complex Wiener-Hopf method. We then show that the cumulant approach can easily be extended to a 12-dimensional model where all types of level-I book events are considered. Within this model, we uncover all the relationships between these types of events and we study the daily amplitude variations of exogenous intensities. In Section 5 we investigate the correlation between two assets by considering the events of their order book within a 16-dimensional model. This allows us to discuss the influence of both their tick size and their degree of reactivity with respect to the impact of their book events on each other. Section 6 contains concluding remarks while some technical details are provided in Appendix.

2 Hawkes processes: definitions and properties

In this section we provide the main definitions and properties of multivariate Hawkes processes and set the notations we need all along the paper.

2.1 Multivariate Hawkes processes and the branching ratio matrix G

A multivariate Hawkes process of dimension d is a d-dimensional counting processes $\boldsymbol{N}_t$ with a conditional intensity vector $\boldsymbol{\lambda_t}$ that is a linear function of past events. More precisely,

$$\lambda_t^i = \mu^i + \sum_{j=1}^{d} \int_{-\infty}^{t} \phi^{ij}(t-s)\, dN_s^j \tag{1}$$

where μ^i represents the baseline intensity while the kernel $\phi^{ij}(t)$ quantifies the excitation rate of an event of type j on the arrival rate of events of type i after a time lag t. In general it is assumed that each kernel is causal and positive, meaning that Hawkes processes can only account for mutual excitation effects since the occurrence of some event can only increase the future arrival intensity of other events. In order to consider the possibility of inhibition effects, one can allow kernels to take negative values. In that case, we have to consider expression (1) only when it provides a positive result while the conditional intensity is assumed to be zero otherwise. Rigorously speaking, such non-linear variant of Eq. (1) cannot be handled as simply as the original Hawkes process ([BM96]) but, as empirically shown in e.g. [RBRGTM14] or [BM16], if the probability that $\lambda_t^i < 0$ is small enough, one can safely consider the model as linear so that all standard expressions provide accurate results. In the following we will suppose that we are in this case and we don't necessarily impose that the kernels $\phi^{ij}(t)$ are positive functions.

Let us define the matrix $\boldsymbol{G}$ as the matrix whose coefficients are the integrals of the kernels $\phi^{ij}(t)$ (that are supported by $\mathbb{R}^+$):

$$G^{ij} = \int_0^{+\infty} \phi^{ij}(t)\, dt\,. \tag{2}$$

Let us remark that, as it can directly be seen from the cluster representation of Hawkes processes ([HO74]), G^{ij} represents the mean total number of events of type i directly triggered by an event of type j. For that reason, in the literature, the matrix $\boldsymbol{G}$ is also referred to as the *branching ratio matrix* ([HB14]). Notice that since the kernels $\phi^{ij}(t)$ are not necessarily non negative functions, G^{ij} does not in general correspond to the L^1 norm of ϕ^{ij}. For the sake of simplicity, though this is not technically correct, we shall often refer to the matrix $\boldsymbol{G}$ as the "matrix of kernel norms" or more simply the "norm matrix".

If $\|\boldsymbol{G}\|$ stands for the largest eigenvalue of $\boldsymbol{G}$, it is well known that a sufficient condition for the intensity process $\boldsymbol{\lambda_t}$ to be stationary is that $\|\boldsymbol{G}\| < 1$. In the following we will always consider this condition satisfied. One can then define the matrix $\boldsymbol{R}$ as:

$$\boldsymbol{R} = (\boldsymbol{I_d} - \boldsymbol{G})^{-1}, \tag{3}$$

where $\boldsymbol{I_d}$ denotes the identity matrix of dimension d.

Let $\boldsymbol{\Lambda}$ denote the mean intensity vector:

$$\boldsymbol{\Lambda} = \mathbb{E}(\boldsymbol{\lambda_t}), \tag{4}$$

so that the ratio $\frac{\mu^i}{\Lambda^i}$ represents the fraction of events of type i that are of exogenous origin. One can easily prove that $\boldsymbol{\Lambda}$ and $\boldsymbol{\mu}$ are related as:

$$\boldsymbol{\Lambda} = \boldsymbol{R}\,\boldsymbol{\mu} \tag{5}$$

If one defines the matrix $\boldsymbol{\Psi}$ as:

$$\boldsymbol{\Psi} = \boldsymbol{G}\boldsymbol{R} = \boldsymbol{R} - \boldsymbol{I_d}, \tag{6}$$

then Ψ^{ij} represents the average number of events of type i triggered (directly or indirectly) by an exogenous event of type j. When one analyzes empirical data within the framework of Hawkes processes, the previous remarks allow one to quantify causal relationships between events in the sense of Granger, i.e., within a well defined mathematical model. In that respect, the coefficients of the matrices $\boldsymbol{G}$ or $\boldsymbol{\Psi}$ can be read as (Granger-)causality relationships between various types of events and used as a tool to disentangle the complexity of the observed flow of events occurring in some experimental situations ([EDD17]). Let us emphasize that such causal implications are just a matter of interpretation of data within a specific model (namely a Hawkes model) and should simply be considered as a convenient and parsimonious way to represent that data. They should not, in any way, be understood as a "physical" causality reflecting their "real nature".

2.2 Integrated Cumulants of Hawkes Process

The NPHC algorithm developed in [ABG+17] and described in Sec. 3 below, enables the direct estimation of the matrix $\boldsymbol{G}$ from a single or several realizations of the process. It relies on the computation of low order cumulant functions whose expressions are recalled below.

Given $1 \leq i, j, k \leq d$, the first three integrated cumulants of the Hawkes process can be, thanks to stationarity, defined as follows:

$$\Lambda^i dt = \mathbb{E}(dN_t^i) \tag{7}$$

$$C^{ij} dt = \int_{\tau \in \mathbb{R}} \Big(\mathbb{E}(dN_t^i dN_{t+\tau}^j) - \mathbb{E}(dN_t^i)\mathbb{E}(dN_{t+\tau}^j) \Big) \tag{8}$$

$$\begin{aligned} K^{ijk} dt = \iint_{\tau,\tau' \in \mathbb{R}^2} \Big(& \mathbb{E}(dN_t^i dN_{t+\tau}^j dN_{t+\tau'}^k) + 2\mathbb{E}(dN_t^i)\mathbb{E}(dN_{t+\tau}^j)\mathbb{E}(dN_{t+\tau'}^k) \\ & - \mathbb{E}(dN_t^i dN_{t+\tau}^j)\mathbb{E}(dN_{t+\tau'}^k) - \mathbb{E}(dN_t^i dN_{t+\tau'}^k)\mathbb{E}(dN_{t+\tau}^j) - \mathbb{E}(dN_{t+\tau}^j dN_{t+\tau'}^k)\mathbb{E}(dN_t^i) \Big), \end{aligned} \tag{9}$$

where Eq. (7) is the mean intensity of the Hawkes process, the second-order cumulant (8) refers to the integrated covariance density matrix and the third-order cumulant (9) measures the skewness of $\boldsymbol{N}_t$. Using the martingale representation ([BM16]) or the Poisson cluster process representation ([JHR15]), one can obtain an explicit relationship between these integrated cumulants and the matrix $\boldsymbol{R}$ (and therefore the matrix $\boldsymbol{G}$ thanks to Eq. (3)). Some straightforward computations (see [ABG+17]) lead to the following identities:

$$\Lambda^i = \sum_{m=1}^{d} R^{im} \mu^m \tag{10}$$

$$C^{ij} = \sum_{m=1}^{d} \Lambda^m R^{im} R^{jm} \tag{11}$$

$$K^{ijk} = \sum_{m=1}^{d} (R^{im} R^{jm} C^{km} + R^{im} C^{jm} R^{km} + C^{im} R^{jm} R^{km} - 2\Lambda^m R^{im} R^{jm} R^{km}). \tag{12}$$

3 The NPHC method

In this section we briefly recall the main lines of the recent non parametric method proposed in [ABG+17] that leads to a fast and robust direct estimation of the branching ratio matrix $\boldsymbol{G}$ without estimating the shape of the kernel functions. This method is based on the remark that, as shown in [JHR15] and as it can be seen in Eqs. (10), (11) and (12), the integrated cumulants of a Hawkes process can be explicitly written as functions of $\boldsymbol{R}$. The NPHC method is a moment method that consists in directly exploiting these equations to recover $\boldsymbol{R}$ and thus $\boldsymbol{G}$.

3.1 Estimation of the integrated cumulants

Let us first introduce explicit formulas to estimate the three moment-based quantities listed in the previous section, namely, $\boldsymbol{\Lambda}$, $\boldsymbol{C}$ and $\boldsymbol{K}$. In what follows, we assume there exists $H > 0$ such that the truncation from $(-\infty, +\infty)$ to $[-H, H]$ of the domain of integration of the quantities appearing in Eqs. (8) and (9) introduces only a small error. This amounts to neglecting tail effects in the covariance density and in the skewness density, and it corresponds to a good approximation if (i) each kernel $\phi^{ij}(t)$ is essentially supported by $[0, H]$ and (ii) the spectral norm $\|\boldsymbol{G}\|$ is less than 1.

In this case, given a realization of a stationary Hawkes process $\{\boldsymbol{N}_t : t \in [0,T]\}$, as shown in [ABG+17], we can write the estimators of the first three cumulants (7), (8) and (9) as

$$\widehat{\Lambda}^i = \frac{1}{T}\sum_{\tau\in Z^i} 1 = \frac{N_T^i}{T} \tag{13}$$

$$\widehat{C}^{ij} = \frac{1}{T}\sum_{\tau\in Z^i}\left(N^j_{\tau+H} - N^j_{\tau-H} - 2H\widehat{\Lambda}^j\right) \tag{14}$$

$$\begin{aligned}\widehat{K}^{ijk} = \frac{1}{T}\sum_{\tau\in Z^i}\left(N^j_{\tau+H} - N^j_{\tau-H} - 2H\widehat{\Lambda}^j\right)\cdot\left(N^k_{\tau+H} - N^k_{\tau-H} - 2H\widehat{\Lambda}^k\right)\\ - \frac{\widehat{\Lambda}^i}{T}\sum_{\tau\in Z^j}\sum_{\tau'\in Z^k}(2H - |\tau'-\tau|)^+ + 4H^2\widehat{\Lambda}^i\widehat{\Lambda}^j\widehat{\Lambda}^k.\end{aligned} \tag{15}$$

In practice, the filtering parameter H is selected by (i) computing estimates of the covariance density at several points t [1], (ii) assessing the characteristic time τ_c after which the covariance density is negligible, and (iii) setting a multiple of τ_c for H, for instance $H = 5\tau_c$.

3.2 The NPHC algorithm

The covariance $\boldsymbol{C}$ only provides $d(d+1)/2$ independent coefficients and is therefore not sufficient to uniquely identify the d^2 coefficients of the matrix $\boldsymbol{G}$. In order to set a sufficient number of constraints, the NPHC approach relies on using all the covariance $\boldsymbol{C}$ along with a restricted number of the $(d^3+3d^2+2d)/6$ third-order independent cumulant components, namely the d^2 coefficients $\boldsymbol{K^c} = \{K^{iij}\}_{1\le i,j\le d}$. Thus, we define the estimator of $\boldsymbol{R}$ as $\widehat{\boldsymbol{R}} \in \operatorname{argmin}_{\boldsymbol{R}} \mathscr{L}(\boldsymbol{R})$, where

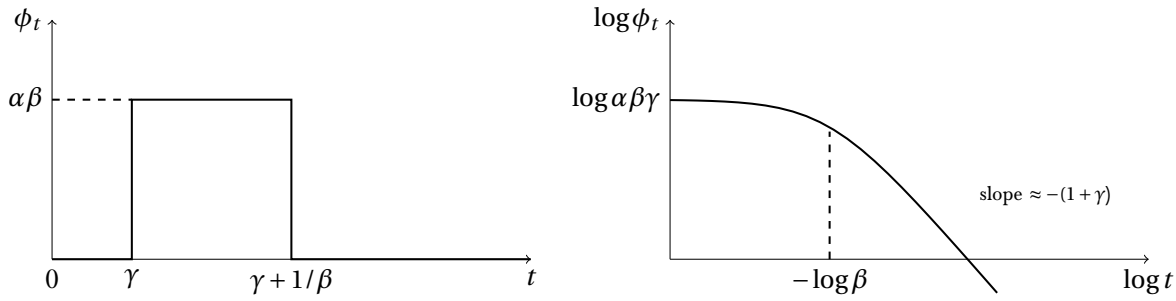
$$\mathscr{L}(\boldsymbol{R}) = (1-\kappa)\|\boldsymbol{K^c}(\boldsymbol{R}) - \widehat{\boldsymbol{K^c}}\|_2^2 + \kappa\|\boldsymbol{C}(\boldsymbol{R}) - \widehat{\boldsymbol{C}}\|_2^2, \tag{16}$$

where $\|\cdot\|_2$ stands for the Frobenius norm, while $\widehat{\boldsymbol{K^c}}$ and $\widehat{\boldsymbol{C}}$ are the respective estimators of $\boldsymbol{C}$ and $\boldsymbol{K^c}$ as defined in Equations (14), (15) above. It is noteworthy that the above mean square error approach can be seen as a particular instance of Generalized Method of Moments (GMM), see [Hal05], [Han82]. Though this framework allows to determine the optimal weighting matrix involved in the loss function, in practice this approach is unusable, as the associated complexity is too high. Indeed, since we have d^2 parameters, this matrix has d^4 coefficients and GMM calls for computing its inverse leading to a $O(d^6)$ complexity. Thus, instead, we choose to use the loss function (16) in which, so as to be of the same order, the two terms are rescaled using $\kappa = \|\widehat{\boldsymbol{K^c}}\|_2^2/(\|\widehat{\boldsymbol{K^c}}\|_2^2 + \|\widehat{\boldsymbol{C}}\|_2^2)$. We refer to Appendix 1 for an explanation of how κ is related to the weighting matrix. Finally the estimator of $\boldsymbol{G}$ is straightforwardly obtained as

$$\widehat{\boldsymbol{G}} = \boldsymbol{I_d} - \widehat{\boldsymbol{R}}^{-1},$$

from the inversion of Eq. (2). The authors of [ABG+17] proved the consistency of the so-obtained estimator $\widehat{\boldsymbol{G}}$, i.e. the convergence in probability to the true value, when the observation time T goes to infinity.

[1] the pointwise covariance density at t can be estimated with $\frac{1}{hT}\sum_{\tau\in Z^i}\left(N^j_{\tau+t+h} - N^j_{\tau+t} - h\widehat{\Lambda}^j\right)$ for a small h



(a) Rectangular kernel

(b) Power-law kernel on log-log scale

Figure V.1: The two different kernels used to simulate the datasets.

Let us mention that, when applied to financial time-series, the number of events is generally large as compared with d (i.e., $n = \max_i |Z^i| \gg d$), thus the matrix inversion in the previous formula is not the bottleneck of the algorithm. Indeed, it has a complexity $O(d^3)$ which is cheap as compared with the computation of the cumulants which is $O(nd^2)$. Thus, assuming the loss function (16) is minimized after N_{iter} iterations, the overall complexity of the algorithm is $O(nd^2 + N_{\text{iter}}d^3)$. The authors of [ABG+17] compared the complexity of their algorithm with other state-of-the-art methods' ones, namely the ordinary differential equations based (ODE) algorithm in [ZZS13], the Sum of Gaussians based algorithm in [XFZ16], the ADM4 algorithm in [ZZS13], and the Wiener-Hopf-based algorithm in [BM16]. The complexity of NPHC is smaller, because the algorithm NPHC directly estimates the kernels' integrals while other methods go through the estimation of the kernel functions themselves.

3.3 Numerical experiments

As mentioned above, the NPHC algorithm is non parametric and provides an estimation of the integral of the kernels regardless of their shapes. In order to illustrate the stability of our method with respect to the shape of the kernels, we simulated two datasets with Ogata's Thinning algorithm introduced in [Oga81] using the open-source library `tick`[2]. Each dataset corresponds to a different kernel shape (but with the same norm), a rectangular kernel and a power-law kernel, both represented in Figure V.1:

$$\text{rectangular kernel: } \phi(t) = \alpha\beta\mathbb{1}_{[0,1/\beta]}(t-\gamma) \tag{17}$$

$$\text{power law kernel: } \phi(t) = \alpha\beta\gamma(1+\beta t)^{-(1+\gamma)} \tag{18}$$

In both cases, α corresponds the integral of the kernel, $1/\beta$ can be regarded as a characteristic time-scale, and γ corresponds to the scaling exponent for the power law kernel and a delay parameter for the rectangular one. We consider a non-symmetric 10-dimensional block-matrix $\boldsymbol{G}$ with 3 non-zero blocks, and where the parameters $\alpha = 1/6$ and $\gamma = 1/2$ take the

[2] https://github.com/X-DataInitiative/tick

same constant values on these blocks. Three different β_0, β_1 and β_2 are used in the different blocks, with $\beta_2/\beta_1 = \beta_1/\beta_0 = 10$ and $\beta_0 = 0.1$. The number of events is roughly equal to 10^5 on average over the nodes. We thus obtain two datasets, the first one referred to as Rect10 corresponding to the rectangular kernels and the second one referred to as PLaw10 corresponding to the power law kernels. We run on these two datsets the NPHC algorithm and the ADM4 algorithm from [ZZS13], which calibrates a single exponential kernel $t \to \alpha\beta e^{-\beta t}$ with constant β, and for which we provided the intermediate true value $\beta = \beta_1$. The results are shown in Figure V.2. These figures clearly illustrate that parametric methods can lead to very poor results when the parametrization does not represent well the data, while NPHC method gives better solutions without knowing scaling parameters β.

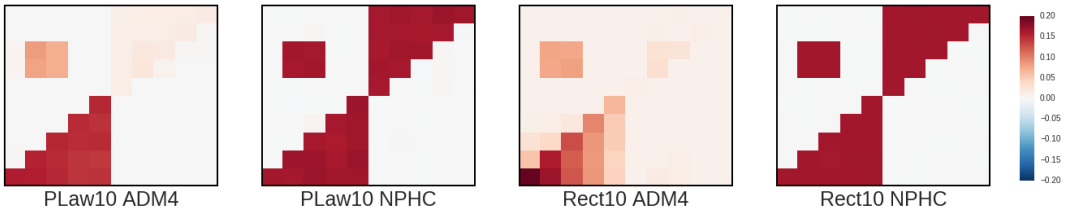


Figure V.2: Estimated matrices $\boldsymbol{G}$ with our NPHC algorithm and the ADM4 algorithm from [ZZS13] on the two datasets Rect10 and PLaw10. NPHC shows significantly better results on these two datasets.

4 Single-asset model

In this section we apply the NPHC method to high-frequency financial data. First we describe our dataset, then we compare the results of the NPHC method with those obtained with the Wiener-Hopf method of [BM16] on the 8-dimensional model of single asset level-I book order events proposed in [BJM16]. We finally discuss the NPHC estimation of the norm matrix associated with a "complete version" (i.e. 12-dimensional) of this model.

4.1 Data

In this paper we use level-I order book data provided by `QuantHouse EUROPE/ASIA`[3] for four future contracts traded on the Eurex exchange, namely the futures on the DAX and Euro Stoxx 50 equity indices, and the Bund and Bobl futures. The DAX and Euro Stoxx 50 indices track the largest stock by market capitalization in Germany and the Euro area respectively, while the Bund and Bobl are German interest rate futures on the 8.5 -10.5 years and the 4.5-5.5 years horizon respectively. The data span a period of 338 trading days from July 2013 to October 2014. For each asset, a line with the current status of the first levels of the order book is added to the database every time there is a change (price, volume or both). Moreover, an additional line is added in the case the change is caused by a market

[3] http://www.quanthouse.com

	T^+	T^-	L^+	L^-	C^+	C^-	T^a	T^b	L^a	L^b	C^a	C^b
DAX	11.9	11.9	21.8	21.9	10.1	10.1	11.6	11.7	80.0	79.5	97.3	96.1
ESXX	2.6	2.6	3.5	3.6	0.9	0.9	16.4	16.5	176.0	174.7	172.4	170.8
Bund	3.2	3.2	4.0	4.0	0.8	0.8	14.5	14.7	125.4	125.0	111.5	110.7
Bobl	1.1	1.1	1.5	1.5	0.5	0.5	6.1	6.1	86.5	86.8	81.6	81.4

Table V.1: Average number of events in thousands per type in a trading day (from open at 08:00 to closing at 22:00 Frankfurt time) for the four assets considered.

order and a trade is generated. It is therefore possible to obtain a list of the orders that were submitted complete with their time, type (limit, cancel or market order), volume and price. The timestamp precision is one microsecond and the timestamps are set directly by the exchange.

In this work we are interested in disentangling the interactions of different types of events occurring at the first level of the order book. To this end, we will distinguish the following event types:

- T^+ (T^-) : upwards (downwards) mid price movement triggered by a market order;
- L^+ (L^-) : upwards (downwards) mid price movement triggered by a limit order;
- C^+ (C^-) : upwards (downwards) mid price movement triggered by a cancel order;
- T^a (T^b) : market order at the ask (bid) that does not move the mid price;
- L^a (L^b) : limit order at the ask (bid) that does not move the mid price;
- C^a (C^b) : cancellation order at the ask (bid) that does not move the mid price.

Additionally, we introduce the symbols P^+ (P^-) to denote an upwards (downwards) mid price movement irrespectively of its origin. In Table V.1 we report the average number of events per day (from 08:00 am to 10:00 pm) for each asset and each type. We remark that all four assets are extremely active securities with an average of more than 300.000 events per day.

One characteristic that strongly influences the order book dynamics at short time scales is the tick size to average spread ratio. When this ratio is close to one (resp. much smaller than one), the asset is said to be a "large tick asset" (resp. a "small tick asset") (see, e.g., [DR15]). In our dataset, all assets are large-tick assets (the spread is equal to one tick in more than 95% of the times) except for the DAX future, which is a small-tick one. As evidenced by Table V.1, the price changes much less frequently on large tick assets. One can also remark that the quantity available at the best quotes tends do be proportionally much larger on large tick assets. These microstructural characteristics will be reflected by our analysis.

4.2 Revising the 8-dimensional mono-asset model of [BJM16] : A sanity check

In [BM14a, BM16], the authors outlined a method for non-parametric estimation of the Hawkes kernel functions based the infinitesimal covariance density and the numerical solution of a Wiener-Hopf system of integral equations that links the covariance matrix and the kernel matrix. Their method has been applied to high-frequency financial data in [BM14a, BJM16], and [RBL17].

The aim of this section is to compare the newly proposed NPHC methodology with the Wiener-Hopf method mentioned above in order to assess the reliability of the new NPHC method. To this end, we reproduce the results obtained in [BJM16].

As it was done there, we consider the DAX and Bund futures data[4] and for each asset we separate Level-I order book events into 8 categories as defined above: P^+, P^-, T^a, T^b, L^a, L^b, and C^a, C^b. Note that here a price move can be of any type. We then consider the timestamp associated with all events as a realization of a 8-dimensional Hawkes process and we use both the NPHC method outlined in Section 3 and the Wiener-Hopf method of [BM16] to estimate the integrated kernel interaction matrix $\boldsymbol{G}$ from the data. For the Wiener-Hopf method, we follow the same procedure as [BJM16] and in particular we estimate the covariance density up to a maximum lag of $\approx 1000s$ using a log-linear spaced grid[5], while for the NPHC method we follow the steps outlined in Section 3 and we fix $H = 500s$ so to be on a comparable scale with the Wiener-Hopf method. Let us note that this scale is several orders of magnitude larger than the typical inter-event time. Indeed, on the assets considered median inter-event times are of the order of $300\mu s$ (the mean being $\approx 50ms$), with minimum time distances in the tens of microseconds.

In Figure V.3, we compare the kernel integral matrices $\boldsymbol{G}$ obtained with the NPHC method (left) with those obtained with the Wiener-Hopf approach (right) on the DAX future. Although the precise values of the matrix entries differ somewhat, as it is difficult to tune the estimation parameters of the two methods as to produce the exact same numerical results, we note that the two methods produce very consistent results. Indeed, they recover the same interaction structure and thus lead to the same interpretation of the underlying system dynamics. In our view, this represents a good sanity check for the proposed NPHC methodology. Analogous results are obtained for the Bund future. Let us also point out that the small asymmetries between symmetric interactions (such as e.g. $T^+ \to T^-$ and $T^- \to T^+$) can be used get a rough measure of the estimation error. In the case presented here, the average absolute difference between symmetric interactions kernels is 0.03, which means relative error of a few percent on the most relevant interactions.

We do not comment here the features emerging from the kernel norm matrices presented in this section since they have been already discussed at length in [BJM16] and some of them will be further discussed in the next sections. Instead, here we highlight that the results of this section provide a strong case for the use of the NPHC method over the Wiener-Hopf

[4] Note that we use the very same dataset as in [BJM16]

[5] As was done in [BJM16], for the estimation of the covariance density we take a linearly spaced grid at short time lags (until a lag of $1ms$) and we switch to a log-spaced one for longer time lags. This allows to estimate the covariance on several orders of magnitude in time.

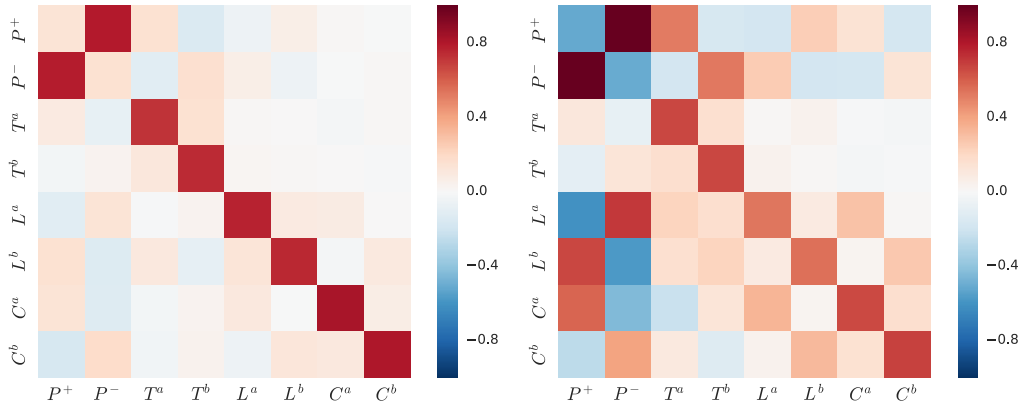


Figure V.3: Kernel norm matrix $\boldsymbol{G}$ estimated with the NPHC method for the DAX future (left) and with the Wiener-Hopf method of [BM16] (right) when the 8-dimensional model described in Section 4.2 is considered.

method when the focus is solely on the kernel interaction matrix. Indeed, in order to estimate the kernel norm matrix with the Wiener-Hopf method, the full kernel functions have to be estimated first and then numerically integrated. The NPHC method thus represents a much faster alternative, as it does not require the estimation of d^2 functions but directly estimates their integrals. Besides the speed gain, the gain in complexity allows NPHC to scale much better when increasing the dimension, i.e., when using more detailed models.

4.3 A 12-dimensional mono-asset model

By estimating directly the norm of the kernels and not the whole kernel function, the NPHC method can be used to investigate systems of greater dimension. In this section we extend the model of Section 4.2 to 12 dimensions by separating the type of events that lead to a price move. The 12 even types we consider are thus T^+ (T^-), L^+ (L^-), C^+ (C^-), T^a (T^b), L^a (L^b), C^a (C^b). We then apply the NPHC algorithm to estimate the branching ratio matrix. When not otherwise specified, we set $H = 500s$. To further assess the validity of our methodology and the impact of time-of-day effects, we first estimate the model using different time slots within the trading day. In Section 4.3.2 we also check the robustness of our results as respect to the choice of the parameter H.

4.3.1 Kernel stability during the trading day

We ran our method for the DAX future on the 12-dimensional point process detailed above on different subintervals of the trading day. More precisely, we divided each trading day into 7 slots with edges at 08:00 am, 10:00 am, 12:00 am, 02:00 pm, 04:00 pm, 06:00 pm and 10:00 pm. We then estimated the 12-dimensional model described above on each slot separately, averaging over all 338 trading days available in our dataset.

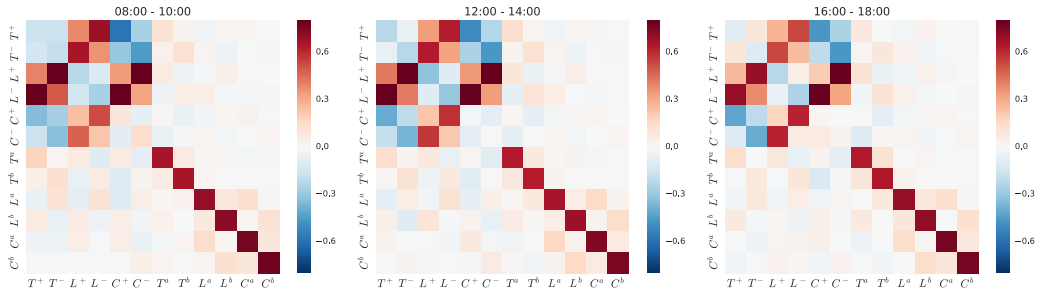


Figure V.4: Kernel norm matrix $\boldsymbol{G}$ for the DAX future estimated (using NPHC) at three different times: between 08:00 and 10:00 (left), between 12:00 and 14:00 (middle) and between 16:00 and 18:00 (right).

In Figure V.4 we show the estimated branching ratio matrix $\boldsymbol{G}$ on three different slots. The results are remarkable in that the kernel norm matrix appears to be very stable during the trading day. (we checked that this is also true if we set $H = 1s$).

The NPHC method outputs the estimated matrix $\widehat{\boldsymbol{R}}$ (and then $\widehat{\boldsymbol{G}}$) from which one can obtain an estimate of $\boldsymbol{\mu}$ using the relation (5) that links $\boldsymbol{R}$ and the mean intensity $\boldsymbol{\Lambda}$, namely $\widehat{\boldsymbol{\mu}} = \widehat{\boldsymbol{R}}^{-1}\widehat{\boldsymbol{\Lambda}}$.

In the right panel of Figure V.5 we plot the values of $\widehat{\mu}$ as obtained using the above relation for the $T^{a/b}$, $L^{a/b}$ and $C^{a/b}$ components. We consider the kernel norm matrix as constant in each two hours slot and we estimate the average intensity on 15 minutes non-overlapping windows. Moreover, for each type of events we show the average of the bid/ask components. For comparison, in the left panel of Figure V.5 we show the empirical intraday pattern obtained for each component. We remark that the values of μ obtained with our procedure vary during the day and roughly follow the intraday curve of the respective components. Let us notice that μ^i/Λ^i, the fraction of exogenous events, is of the order of a few percent. This is fully consistent with what was found in [BJM16] and means, within the Hawkes framework, that most of the observed order book dynamics is strongly endogenous. For the price moving components the values of Λ are of the order of $1s^{-1}$, while results for μ are more noisy, similarly to those of $T^{a/b}$.

This analysis confirms the result formerly observed in [BM14a] that the kernels are stable during the day, and that time-of-day effects are well captured by the baseline intensity, at least as long as we are mainly concerned with the high frequency dynamics on a very liquid asset as is the case here.

4.3.2 Analysis of the G matrix: Unveiling mutual interactions between book events

Having established that the estimated kernel matrix is stable with respect to time of the day effects, we now examine more in-depth its structure. In Figure V.6 is represented the result of the estimation of the matrix $\boldsymbol{G}$ over the whole trading day for the DAX future. The branching ratio matrix on the left panel is estimated with $H = 1s$ while the right panel corresponds to

$H = 500s$. Let us recall that both horizons are several orders of magnitude larger than the typical inter-event time.

Concerning the differences between the two matrices, we note that certain inhibitory effects that are visible for $H = 1s$ are less intense or disappear when $H = 500s$ is used. This most notably happens for the elements $T^+ \to T^+$ and $T^+ \to T^-$ and similarly for $L^+ \to L^{+/-}$ and $C^+ \to C^{+/-}$, which suggests that when we look at longer scale correlation the self-exciting behavior (i.e. trades are followed by more trades) tends to prevail on the high frequency mean reverting effect.

Apart from these differences, we can make some observations that are valid in both cases. In particular, we note that two main interaction blocks stand out. The first is the upper left corner which concerns interactions between price-moving events, where two anti-diagonal bands are prominent. The second is the bottom right corner, which has a strong diagonal structure. The blocks involving interactions between price-moving and non-price moving events present instead much smaller values. In what follows, we first discuss more in depth the effects of price movements on other events, then those of non-price-moving ones. We also remark that the spectral norm of the estimated matrices $\boldsymbol{G}$ is close to 1 while being inferior (e.g. 0.98 for the DAX with $H = 500s$). This is in line with what was found in [BJM16], and the criticality of financial markets highlighted in [HBB13].

Before entering into more details, let us remark that in both cases the expected symmetry up/down $(+/-)$ and bid-ask (b/a) is well recovered in our results. Therefore, to make notation lighter and facilitate the exposition, we will comment only on one side. More precisely, when discussing the effects of price moves we will refer only to the upwards ones $(T^+/L^+/C^+)$ and when discussing effects of liquidity changes we will focus on ask side events $(T^a/L^a/C^a)$.

Effect of price-moving events As we noted above, the most relevant interactions involving T^+ are the $T^+ \to L^+$ and $T^+ \to L^-$ ones, the mean reverting one $(T^+ \to L^-)$ being more intense.

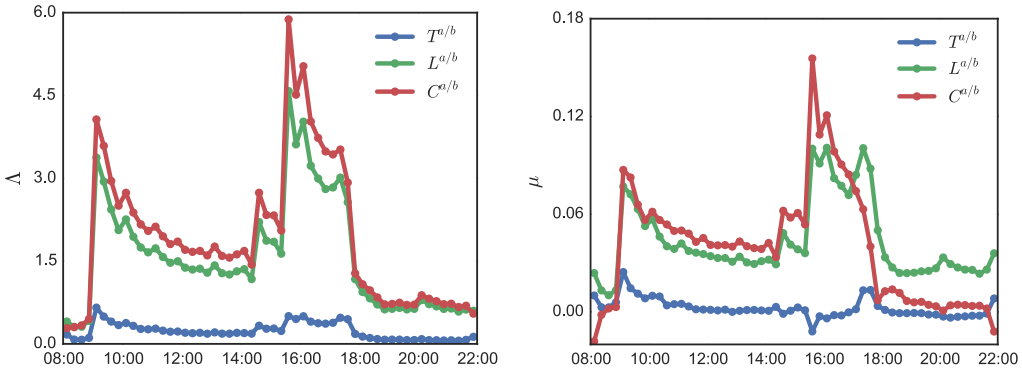


Figure V.5: Estimation of the baseline intensities of each event type within a trading day for the DAX future using 15 min slots. Left panel: Empirical intraday pattern measured using market, limit and cancel orders that do not move the price. Right panel: μ values estimated using the NPHC method. All quantities are expressed in s^{-1}.

When a market order consumes the liquidity available at the best ask, two main scenarios can occur for the mid price to change again, either the consumed liquidity is replaced, reverting back the price (mean-reverting scenario, highly probable) or the price moves up again and a new best bid is created.

Market orders that move the mid price have also an inhibitory effects at short time scales on subsequent price-moving trades ($T^+ \to T^+$ is negative for $H = 1s$). Indeed, once a market order consumes the liquidity available at the best quote, it is unlikely that the price will be moved in the same direction by other market orders as the price becomes more unfavorable. We also note a generally inhibitory effect of T^+ on price-moving cancel orders which can be linked to a mechanical effect, liquidity that has been consumed by the market order cannot be canceled anymore.

The same kind of dynamics is at play also in the interactions $L^+ \to T^+$ and $L^+ \to T^-$ with the roles inverted. Again, the mean reverting effect $L^+ \to T^-$ appears to be much more probable. A strong mean-reverting effect is found in the block $L^+ \to C^-$. This is possibly the signature of high-frequency strategies whereby agents place limit orders in the spread and cancel them shortly thereafter.

Concerning C^+ events, the main feature lies in the block $C^+ \to L^-$, where we notice the same anti-diagonal dominance found for the block $L^+ \to C^-$. Again, we can suppose that when a limit order in the spread is removed it is often quickly replaced by market participants.

Finally, the effect of price moving events on non-price moving ones can be summarized in two main effects. The first is a trend-following/order splitting effect by which e.g. trades at the ask are likely to be followed by more trades in the same direction ($T^+ \to T^a$) and similarly for limit ($L^+ \to L^b$) and cancel ($C^+ \to C^a$) orders. The second is the shift in liquidity triggered by a price change. A trade at the ask that moves upward the mid price triggers limit orders on the opposite side ($T^+ \to L^b$). This can be understood using a latent price argument ([RR10]), as it is well known that there are more limit orders far from the latent price. Right after the mid price goes up, the latent price is expected to be closer to the newly best ask price than to the best bid price, thus limit order flow is expected to be higher at best bid than at best ask.

Effect of non-price-moving events For all events T^a, L^a and C^a the most visible feature is the strong self-exciting interaction. This has been confirmed in several works ([BJM16], [RBL17]) and can be traced to order-splitting strategies and herding behaviors. Signatures of typical trading patterns can be seen also in the kernels $L^a \to C^a$, $L^a \to C^b$, where the positive value of the kernel arises form agents canceling and replacing their limit orders with or without switching sides.

We also note the positive effects $T^a \to T^+$, $L^a \to T^-$ and $C^a \to T^+$. All these effects, as well as the analogous ones on $C^{+/-}/L^{+/-}$, reflect the fact that changes in the imbalance have an influence on the probability of a subsequent price move. So when the queue at the best ask decrease an upward price move becomes more likely and vice-versa. These effects are much more relevant on a small tick asset (DAX) than on a large tick asset (Bund) where, the size of the queues being larger, their influence is marginal.

We performed the same analysis on the Bund (see Figure V.7). The main differences as compared to the DAX are that the effects between events that move the price are much more

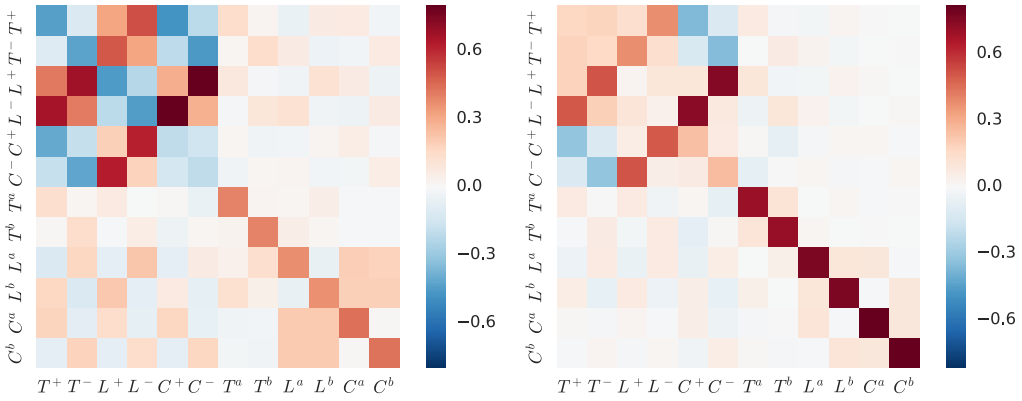


Figure V.6: Kernel norm matrix $\boldsymbol{G}$ estimated with the NPHC method for the DAX future with $H = 1s$ (left) and $H = 500s$ (right).

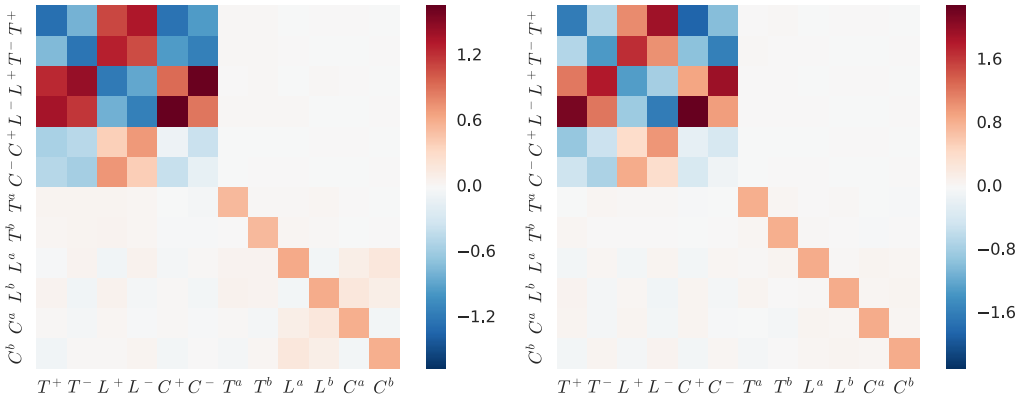


Figure V.7: Kernel norm matrix $\boldsymbol{G}$ estimated with the NPHC method for the Bund future with $H = 1s$ (left) and $H = 500s$ (right).

intense while the effects of events that do not move the price on those that do move the price (and vice-versa) are much less pronounced, indeed they are barely visible in Figure V.7. This can be basically seen as a simple consequence of the Bund future being large tick assets, while the DAX is a small tick one. Therefore, price movements on the former are much less frequent but when they happen their effects are more marked.

4.3.3 Analysis of the Ψ matrix: the fingerprint of meta-orders

As discussed in Section 2, the elements of the matrix $\boldsymbol{\Psi}$ quantifies the total effect, direct and indirect, of an event of type j on events of type i. More precisely, thanks to the branching process structure, we can interpret ψ^{ij} as the mean number of events of type i generated by a single exogenous ancestor of type j. We plot the estimated matrices $\boldsymbol{\Psi}$ for the DAX and

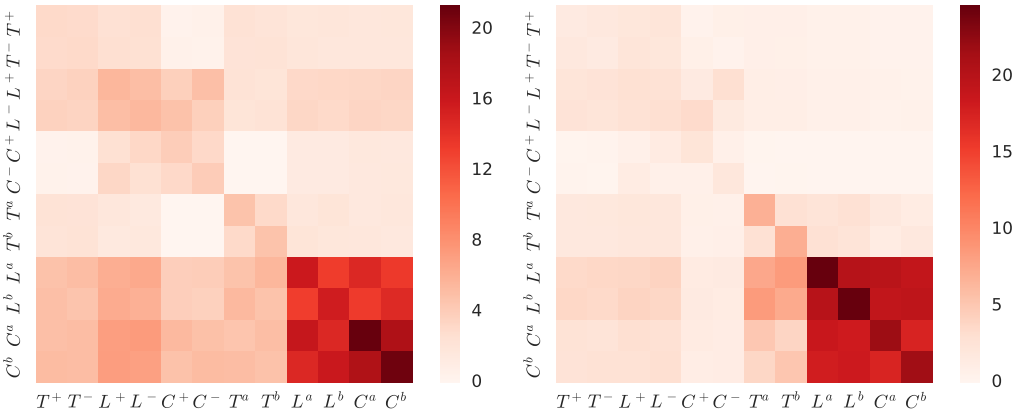


Figure V.8: $\mathbf{\Psi}$ matrix of eq. (6) estimated with the NPHC method for the DAX future (left) and the Bund future (right) with $H = 500s$.

Bund futures in Figure V.8. The main feature that appears for both assets is the set of strong values found in the bottom right corner, namely in the columns and lines associated with $L^{a/b}$ and $C^{a/b}$. We note that an exogenous limit or cancel event generates a large number of limit and cancel events and, to a lesser extent, trade events. This can be read as the signature of meta-orders. Indeed, if an agent wants to sell a large number of contracts[6], he will place a meta-order, i.e., he will optimize the overall cost by dividing this large order into several smaller orders. The overall optimization will result in many limit/cancel sell orders L^a, C^a and, as less as possible, of sell market orders T^b (the cost of a market order is on average higher than that of a limit order). The same description can be applied to understand why an exogenous sell market order T^b generates mainly limit and cancel sell orders L^a, C^a as well as other sell market orders T^b.

Due to the much lower values of the exogenous intensities for price moving events, the left part of the $\mathbf{\Psi}$ matrix is more noisy. Nevertheless, at least in the DAX case, we note also for the price moving components the prevalence of the $L^+ \to L^+$ and $L^+ \to C^-$ elements, which are the price-moving counterparts of the effect described for L^a.

Finally, we also remark that although we noted several inhibition effects in the matrices $\boldsymbol{G}$, the elements of $\mathbf{\Psi}$ are non negative. This suggests that most inhibition effects are short lived and the effect of an event arrival is towards an increase of the overall intensity. This is in line with what was found in [BJM16] and [RBL17], where the inhibitions effects were shown to be mostly concentrated around the typical market reaction time.

Within the branching ratio representation of Hawkes processes, $\frac{\mu^j}{\Lambda^i}\psi^{ij}$ represents the fraction of events of type i that has a type j as primary ancestor. Along the same line, we can estimate the fraction of aggressive orders (i.e. all T), as opposed to passive orders (L or

[6] Let us recall that, in our discussion, we only address half of the matrix coefficients since the discussion on the other half can be obtained using the symmetries ask/bid, buy/sell, price up/price down. Following these lines, we only consider here the case of a selling meta-order.

C), that is ultimately generated by another aggressive order, as:

$$\frac{1}{\sum_{i=\{T^{+/-},T^{a/b}\}}\Lambda^i}\sum_{j=\{T^{+/-},T^{a/b}\}}\sum_{i=\{T^{+/-},T^{a/b}\}}\psi^{ij}\mu^j\,. \tag{19}$$

We find that for both assets this fraction is about 10%, which means that the large majority of market orders have a “passive order" (L or C) oldest ancestor. We compute the analogous fraction for passive orders and we find that for both assets more than 96% of the passive orders (L or C) have an oldest ancestor that is itself a passive (L or C) order. This fact is in line with the idea that meta-orders would be at the origin of most of the trading activity within the order book.

5 Multi-asset model

Studying and quantifying the interactions and comovements within a basket of assets is an important topic in finance. Most of these studies focus on the return correlations properties in relationship with portfolio theory. At very high frequency, the discrete nature of price variations and the asynchronous occurrence of price change events make the correlation analysis trickier and, in order to avoid well known bias (like the Epps effect) one has to use specific techniques like the estimator proposed by [HY+05]. Hawkes processes, being naturally defined in continuous time, can represent a complementary tool for the investigation of high-frequency cross-asset dynamics.

The idea of capturing the joint dynamic of multiple assets via Hawkes processes has only been considered in few recent papers. Let us mention the work proposed by [BCT+15] which models the simultaneous cojumps of different assets using a one-dimensional Hawkes process, and a more recent work ([DFZ17]) which focuses on the correlation and lead-lag relationships between the price changes of two assets, in the spirit of [BDHM13].

In this section, we aim at unveiling a more precise structure of the high-frequency cross-asset dynamics by pushing further the dimensionality of the model to include simultaneously events on two assets. We first consider the pair DAX-EURO STOXX and then the one Bobl-Bund. The pairs of assets considered here are tightly related, as they share exposure to the same risk factors and, in the case of DAX-EURO STOXX, also because the underlying indices actually share a significant part of their components. This is confirmed also by Table V.2 where we report 5 minutes return correlations among the considered assets.

In this section we consider the same kind of events as in Section 4.2 and we have therefore a 16-dimensional model (2×8) corresponding to 256 possible interactions. Let us point out that this is quite a large dimension value for a non parametric methodology.

5.1 The DAX - EURO STOXX model

In the following, we will denote the events of the DAX order book with the subscript D while we will use the subscript X for the events of EURO STOXX order book. The obtained branching ratio matrix is displayed in Figure V.9. We observe that the mono-asset submatrices (the two 8×8 block matrices along the diagonal), which present the most relevant effects,

	DAX	ESXX	Bobl	Bund
DAX	1.00	0.89	-0.18	-0.22
ESXX	0.89	1.00	-0.19	-0.22
Bobl	-0.18	-0.19	1.00	0.85
Bund	-0.22	-0.22	0.85	1.00

Table V.2: Five minutes return correlation coefficients for the examined assets.

have the same structure as the ones which have already been commented on in detail in Section 4.2. Consequently, in this section, we shall focus our discussion on the non diagonal 8×8 submatrices that correspond to the interactions between the two assets. These two submatrices are shown in Figure V.10. Note that colors have been rescaled to highlight their structure. To keep the notation lighter, we will comment only on effect of upwards price moves and ask events as it was done in the previous section, since we find the symmetries $+/-$ and a/b to be well respected. The most striking feature emerging from Figure V.10 is the very intense relation between same-sign price movements on the two assets. Albeit present in both directions, the norms $P_X^+ \to P_D^+$ attain larger values.

Another notable aspect is the different effects of price moves and liquidity changes of one asset on events on the other asset. Price moves on the DAX have also an effect on the flow of limit orders on EURO STOXX ($P_D^+ \to L_X^b$ and $P_D^+ \to C_X^a$), whereas EURO STOXX price moves triggers mainly DAX price moves in the same direction ($P_X^+ \to P_D^+$). An important aspect for understanding this result is the different perceived tick sizes on the two assets.

In the following, whenever it is convenient, we shall place the discussion withing the framework of latent price models (e.g., [RR10]). Within this framework, the latent price refers to an underlying efficient price representing at any time some average opinion of market participants about the value of the asset. As noted in Section 4.1, the DAX future is a small-tick asset, while the EURO STOXX future is a large-tick one ([EBK12]). As a consequence, an upward move in the DAX price (P_D^+), while signaling that the market latent price has moved slightly upwards, is not sufficient to move the EURO STOXX price by a full tick. However, this move can be perceived in the EURO STOXX through the L_X^b and C_X^a flows that are increasing. Indeed, as already mentioned in Section 4.2, it is well known ([RR10]) that there are more limit orders far from the latent price. The latent price went up, so it is now closer to the best ask, and hence the flow of the limit (resp. cancel) orders on the best bid (resp. ask) is increasing.

In the opposite direction, a change in EURO STOXX price is perceived as “large" and triggers price changes in the same direction on the DAX. Interestingly, we can also note that changes in the latent price on the EURO STOXX triggers price movements on the DAX. For instance, a shift of liquidity at the bid, namely an increase of the arrival flow of limit orders at the bid, that signals that the latent price has moved upwards, has a direct effect on upward price moves on the DAX. This can be seen from the interactions $T_X^a \to P_D^+$, $L_X^b \to P_D^+$ and $C_X^a \to P_D^+$.

We can summarize our results by saying that price changes and liquidity changes on the

DAX mainly influence liquidity (latent price) on the EURO STOXX, while price changes and liquidity changes on the EURO STOXX tend to trigger price moves on the DAX.

Finally, let us note that the above effects are even more pronounced when we estimate the interaction matrices with a smaller H. In particular the effects of DAX price movements on T, L, C on the EURO STOXX become more relevant compared with those on prices. At the same time, while the effect of EURO STOXX price moves on DAX's ones is still strong, the effect of liquidity movements on DAX price movements is comparatively stronger with smaller H. This suggests that these effects are mainly localized at short time scales, while the $P^+ \to P^+$ ones have much slower decay in time.

5.2 Bobl - Bund

We perform the same analysis on the asset pair Bobl-Bund futures. Here both assets are large tick assets, however the Bund is much more actively traded than the Bobl in the sense that all the order flows are of higher intensity. The cross-asset submatrices are depicted in Figure V.11. As in the previous case, we remark that the elements $P^+_L \to P^+_M$ and $P^+_M \to P^+_L$ reflect the strong correlation observed between the two assets. Price changes in the Bund have also a noticeable effect on limit/cancel order flows in the Bobl, while price changes in the Bobl have little to no effect on the Bund except for the mentioned $P^+_M \to P^+_L$ interaction. At the same time, T^a, L^a, C^a events on the Bobl impact prices on the Bund, while the corresponding event on the Bund have little effect.

Comparing this with the case of the DAX-EURO STOXX pair, we can liken the effect of the Bund on the Bobl to that of the DAX over the EURO STOXX and vice-versa. We argue that the difference in trading frequency between the Bobl and Bund contracts has a similar effect of that of a different tick size that we observed in the previous case. As before, we have an asset, the Bund, which is more "reactive" (the limit/cancel order flows are higher than those of the Bobl) than the Bobl, thus a price change of the Bund indicating a change of the latent price impacts the limit/cancel flows of the Bobl. In the previous case, the higher "reactivity" of the DAX was due to its smaller tick size.

6 Conclusion and prospects

In the context of Hawkes processes, the estimation of the matrix kernel norms is essential, as it gives a clear overview of the dependencies involved in the underlying dynamics. In the context of high-frequency financial time-series non-parametric estimation of the matrix kernel norms has already shown to be very fruitful ([BM14a, BJM16]), since it provides a very rich summary of the system interactions, and it can thus be a valuable tool in understanding a system where many different types of events are present. However, its estimation is a computationally demanding process since these estimations are computed from a non-parametric pre-estimation of the kernels themselves, i.e., their entire shape and not only their norm. The resulting complexity prevents the estimations from being performed when the dataset is too heavy or (more important) when the dimension of the Hawkes process (i.e., the number of considered different event types) is too large.

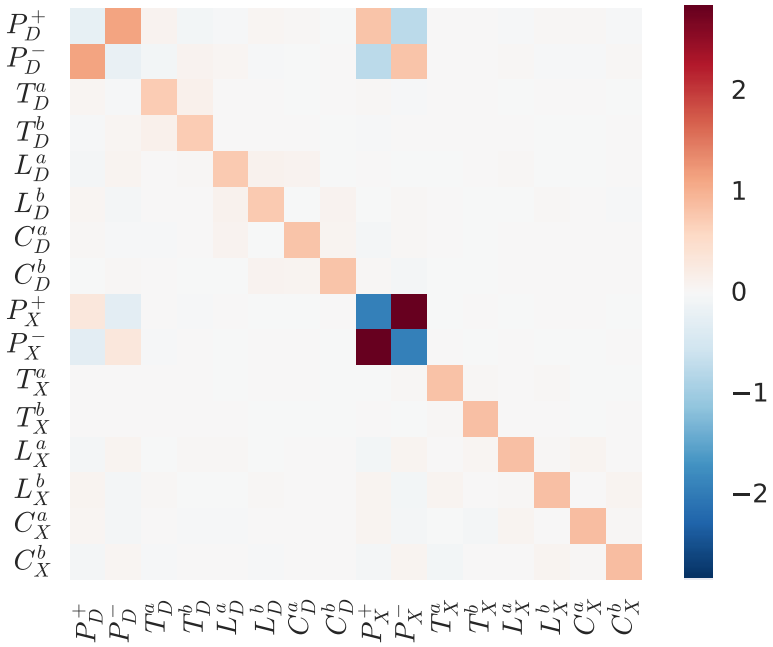


Figure V.9: Hawkes kernel norm matrix obtained when the DAX and EURO STOXX futures are considered simultaneously in a 16D model. DAX events are denoted with the D subscript, EURO STOXX ones with the X subscript.

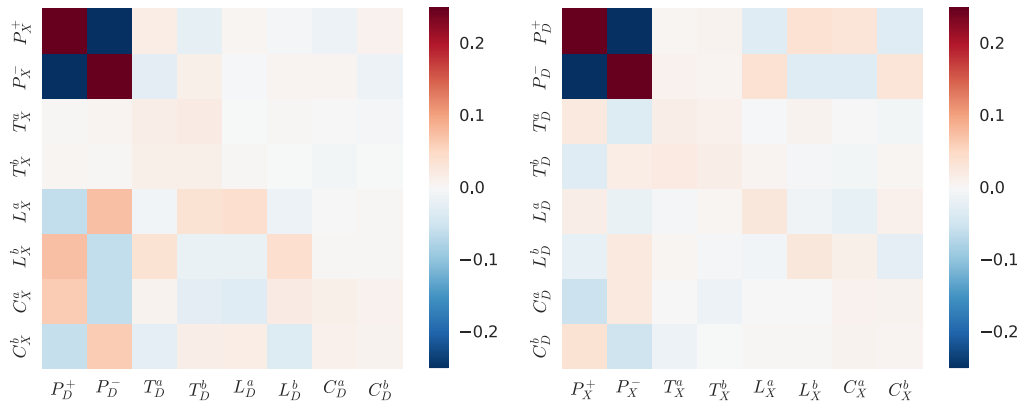


Figure V.10: Submatrices of the Kernel norm matrix $\boldsymbol{G}$ corresponding to the effect of DAX events on EUROSTOXX STOXX events (left) and vice versa (right). These two submatrices correspond to the ones lying on the antidiagonal on the Figure V.9

In this work, we presented the newly developed NPHC algorithm ([ABG+17]) that allows to *directly* estimate non-parametrically the kernel norms matrix of a multidimensional Hawkes process, that is without going through the kernel shapes pre-estimation step. As of today, it is the only *direct* non-parametric estimation procedure available in the academic literature. This method can be seen as a Generalized Method of Moments (GMM) that relies on second-order and third-order integrated cumulants. This paper shows that this method successfully reveals the various dynamics between the different (first level) order flows involved in order books. In a context of a single-asset 8-dimensional Hawkes process, we have shown (as a "sanity check") that it is able to reproduce former results obtained using "indirect" methods. Moreover, the so-obtained gain in complexity allowed us to run a much more detailed analysis (increasing the dimension to 12), separating the different types of events that lead to a mid-price move. This in turn allowed us to have a very precise picture of the high frequency order book dynamics, revealing, for instance, the different interactions that lead to the high-frequency price mean reversion or those between liquidity takers and liquidity makers as well as the influence of the tick-size of these dynamics. Not the least, through the analysis of the matrix $\mathbf{\Psi}$ we also detected the signature of meta-orders. We have also successfully used the NPHC algorithm in a multi-asset 16-dimensional framework. It allowed us to unveil very precisely the high-frequency joint dynamics of two assets that share exposure to the same risk factors but that have different characteristics (e.g., different tick sizes or different degrees of reactivity). It is noteworthy that our methodology can efficiently highlight these types of dynamics, especially since cross-asset effects are second order effects compared to mono-asset's.

We conclude by noting that our study left out some relevant information such as the volume of the orders and the size of the jumps in the mid-price. This will be the objective of future works. Moreover, within the methodology presented in this paper, an analysis of baskets of assets (with more than two assets) as well as multi-agent high-frequency interactions are currently under progress.

Acknowledgments

This research benefited from the support of the Chair "Changing Markets", under the aegis of Louis Bachelier Finance and Sustainable Growth laboratory, a joint initiative of Ecole Polytechnique, Université d'Evry Val d'Essonne and Fédération Bancaire Française and from the chair of the Risk Foundation: Quantitative Management Initiative.

1 Origin of the scaling coefficient κ

Following the theory of GMM, we denote $m(X,\theta)$ a function of the data, where X is distributed with respect to a distribution $\mathbb{P}_{\theta_0}$, which satisfies the *moment conditions* $g(\theta) = \mathbb{E}[m(X,\theta)] = 0$ if and only if $\theta = \theta_0$, the parameter θ_0 being the *ground truth*. For $x_1, \dots, x_N$ observed copies of X, we denote $\widehat{g}_i(\theta) = m(x_i,\theta)$, the usual choice of weighting matrix is

$\widehat{W}_N(\theta) = \frac{1}{N}\sum_{i=1}^{N} \widehat{g}_i(\theta)\widehat{g}_i(\theta)^\top$, and the objective to minimize is then

$$\left(\frac{1}{N}\sum_{i=1}^{N} \widehat{g}_i(\theta)\right)\left(\widehat{W}_N(\theta_1)\right)^{-1}\left(\frac{1}{N}\sum_{i=1}^{N} \widehat{g}_i(\theta)\right), \tag{20}$$

where θ_1 is a constant vector. Instead of multiplying by the inverse weighting matrix, we have decided to divide by the sum of its eigenvalues, which is easily computable:

$$\begin{aligned}\mathrm{Tr}(\widehat{W}_N(\theta)) &= \frac{1}{N}\sum_{i=1}^{N} \mathrm{Tr}(\widehat{g}_i(\theta)\widehat{g}_i(\theta)^\top) \\ &= \frac{1}{N}\sum_{i=1}^{N} \mathrm{Tr}(\widehat{g}_i(\theta)^\top\widehat{g}_i(\theta)) \\ &= \frac{1}{N}\sum_{i=1}^{N} ||\widehat{g}_i(\theta)||_2^2\end{aligned}$$

In our case, $\widehat{g}(\boldsymbol{R}) = \left[\mathbf{vec}[\widehat{\boldsymbol{K}^{\boldsymbol{c}}} - \boldsymbol{K}^{\boldsymbol{c}}(\boldsymbol{R})], \mathbf{vec}[\widehat{\boldsymbol{C}} - \boldsymbol{C}(\boldsymbol{R})]\right]^\top \in \mathbb{R}^{2d^2}$. Assuming the associated weighting matrix is block-wise, one block for $\widehat{\boldsymbol{K}^{\boldsymbol{c}}} - \boldsymbol{K}^{\boldsymbol{c}}(\boldsymbol{R})$ and the other for $\widehat{\boldsymbol{C}} - \boldsymbol{C}(\boldsymbol{R})$, the sum of the eigenvalues of the first block becomes $\|\widehat{\boldsymbol{K}^{\boldsymbol{c}}} - \boldsymbol{K}^{\boldsymbol{c}}(\boldsymbol{R})\|_2^2$, and $\|\widehat{\boldsymbol{C}} - \boldsymbol{C}(\boldsymbol{R})\|_2^2$ for the second. We compute the previous terms with $\boldsymbol{R}_1 = 0$. All together, the objective function to minimize is

$$\frac{1}{\|\widehat{\boldsymbol{K}^{\boldsymbol{c}}}\|_2^2}\|\boldsymbol{K}^{\boldsymbol{c}}(\boldsymbol{R}) - \widehat{\boldsymbol{K}^{\boldsymbol{c}}}\|_2^2 + \frac{1}{\|\widehat{\boldsymbol{C}}\|_2^2}\|\boldsymbol{C}(\boldsymbol{R}) - \widehat{\boldsymbol{C}}\|_2^2, \tag{21}$$

which equals the loss function given in 16, up to a constant.

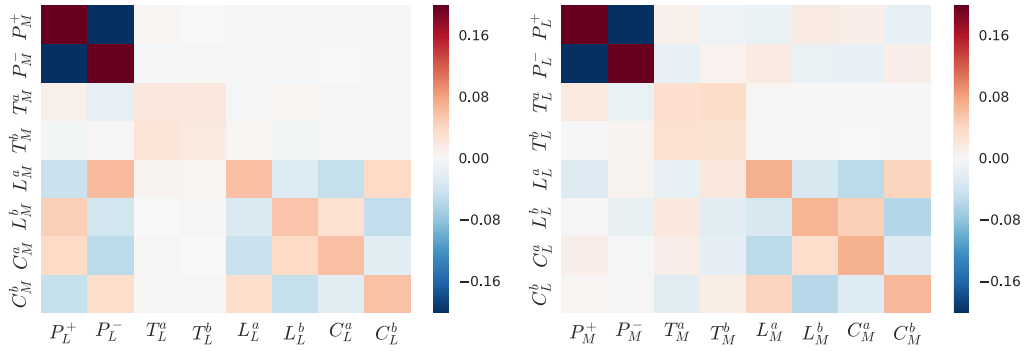


Figure V.11: Submatrices of the Kernel norm matrix $\boldsymbol{G}$ corresponding to the effect of Bund (L) events on Bobl (M) events (left) and vice-versa (right).

Bibliography

[AAB+16] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[ABG+17] M. Achab, E. Bacry, S. Gaïffas, I. Mastromatteo, and J.-F. Muzy. Uncovering causality from multivariate hawkes integrated cumulants. *Proceedings of the International Conference on Machine Learning*, 2017.

[ABGK12] P. K. Andersen, O. Borgan, R. D. Gill, and N. Keiding. *Statistical models based on counting processes*. Springer Science & Business Media, 2012.

[AED+00] A. A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu, et al. Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, 403(6769):503–511, 2000.

[AFM17] Y. F. Atchadé, G. Fort, and E. Moulines. On perturbed proximal gradient algorithms. *Journal of Machine Learning Research*, 18(10):1–33, 2017.

[ASCDL10] Y. Aït-Sahalia, J. Cacho-Diaz, and R. J. Laeven. Modeling financial contagion using mutually exciting jump processes. Technical report, National Bureau of Economic Research, 2010.

[B+15] S. Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[BBH12] C. Blundell, J. Beck, and K. A. Heller. Modelling reciprocating relationships with hawkes processes. In *Advances in Neural Information Processing Systems*, pages 2600–2608, 2012.

[BCN16] L. Bottou, F. E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *arXiv preprint arXiv:1606.04838*, 2016.

[BCT+15] G. Bormetti, L. Calcagnile, M. Treccani, F. Corsi, S. Marmi, and F. Lillo. Modelling systemic price cojumps with Hawkes factor models. *Quantitative Finance*, 15(7):1137–1156, 2015.

[BDHM13] E. Bacry, S. Delattre, M. Hoffmann, and J.-F. Muzy. Modelling microstructure noise with mutually exciting point processes. *Quantitative Finance*, 13(1):65–77, 2013.

[Ber14] D. P. Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

[BG96] M. Broadie and P. Glasserman. Estimating security price derivatives using simulation. *Management science*, 42(2):269–285, 1996.

[BGM15] E. Bacry, S. Gaïffas, and J.-F. Muzy. A generalization error bound for sparse and low-rank multivariate hawkes processes. *arXiv preprint arXiv:1501.00725*, 2015.

[BHG07] D. Blatt, A. O. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2007.

[BILL15] E. Bacry, A. Iuga, M. Lasnier, and C.-A. Lehalle. Market impacts and the life cycle of investors orders. *Market Microstructure and Liquidity*, 01(02):1550009, 2015.

[BJM16] E. Bacry, T. Jaisson, and J.-F. Muzy. Estimation of slowly decreasing hawkes kernels: application to high-frequency order book dynamics. *Quantitative Finance*, 16(8):1179–1201, 2016.

[BM96] P. Brémaud and L. Massoulié. Stability of nonlinear hawkes processes. *The Annals of Probability*, pages 1563–1588, 1996.

[BM13] F. Bach and E. Moulines. Non-strongly-convex smooth stochastic approximation with convergence rate o (1/n). In *Advances in neural information processing systems*, pages 773–781, 2013.

[BM14a] E. Bacry and J.-F. Muzy. Hawkes model for price and trades high-frequency dynamics. *Quantitative Finance*, 14(7):1147–1166, 2014.

[BM14b] E. Bacry and J.-F. Muzy. Second order statistics characterization of hawkes processes and non-parametric estimation. *arXiv preprint arXiv:1401.0903*, 2014.

[BM16] E. Bacry and J.-F. Muzy. First-and second-order statistics characterization of hawkes processes and non-parametric estimation. *IEEE Transactions on Information Theory*, 62(4):2184–2202, 2016.

[BMM15] E. Bacry, I. Mastromatteo, and J.-F. Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01):1550005, 2015.

[Bot98] L. Bottou. Online learning and stochastic approximations. *On-line learning in neural networks*, 17(9):142, 1998.

[Bot10] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[Bow07] C. G. Bowsher. Modelling security market events in continuous time: Intensity based, multivariate point process models. *Journal of Econometrics*, 141(2):876–912, 2007.

[BPC+11] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[BT09] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[BV04] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[BvdBS+15] M. Bogdan, E. van den Berg, C. Sabatti, W. Su, and E. J. Candès. Slope—adaptive variable selection via convex optimization. *The annals of applied statistics*, 9(3):1103, 2015.

[Cau47] A. Cauchy. Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538, 1847.

[CCS12] H. Chen, R. H. Chiang, and V. C. Storey. Business intelligence and analytics: From big data to big impact. *MIS quarterly*, 36(4), 2012.

[CH79] D. R. Cox and D. V. Hinkley. *Theoretical statistics*. CRC Press, 1979.

[CHM+15] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun. The loss surfaces of multilayer networks. In *AISTATS*, 2015.

[Cox75] D. R. Cox. Partial likelihood. *Biometrika*, 62(2):269–276, 1975.

[CPH05] M. A. Carreira-Perpinan and G. E. Hinton. On contrastive divergence learning. In *Aistats*, volume 10, pages 33–40, 2005.

[CS08] R. Crane and D. Sornette. Robust dynamic classes revealed by measuring the response function of a social system. *Proceedings of the National Academy of Sciences*, 105(41), 2008.

[Dav72] C. R. David. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society*, 34:187–220, 1972.

[DBLJ14] A. Defazio, F. Bach, and S. Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, pages 1646–1654, 2014.

[DFZ14] J. Da Fonseca and R. Zaatour. Hawkes process: Fast calibration, application to trade clustering, and diffusive limit. *Journal of Futures Markets*, 34(6):548–579, 2014.

[DFZ17] J. Da Fonseca and R. Zaatour. Correlation and lead–lag relationships in a hawkes microstructure model. *Journal of Futures Markets*, 37(3):260–285, 2017.

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

[Don06] D. L. Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

[DR15] K. Dayri and M. Rosenbaum. Large tick assets: implicit spread and optimal tick size. *Market Microstructure and Liquidity*, 1(01):1550003, 2015.

[DS09] J. Duchi and Y. Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10(Dec):2899–2934, 2009.

[Dur97] E. Durkheim. *Le suicide: étude de sociologie.* F. Alcan, 1897.

[DVJ07] D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure.* Springer Science & Business Media, 2007.

[DXH+14] B. Dai, B. Xie, N. He, Y. Liang, A. Raj, M.-F. F. Balcan, and L. Song. Scalable kernel methods via doubly stochastic gradients. In *Advances in Neural Information Processing Systems*, pages 3041–3049, 2014.

[EBK12] Z. Eisler, J.-P. Bouchaud, and J. Kockelkoren. The price impact of order book events: market orders, limit orders and cancellations. *Quantitative Finance*, 12(9):1395–1419, 2012.

[EDD17] M. Eichler, R. Dahlhaus, and J. Dueck. Graphical modeling for multivariate hawkes processes with nonparametric link functions. *Journal of Time Series Analysis*, 38(2):225–242, 2017.

[EL14] L. Einav and J. Levin. Economics in the age of big data. *Science*, 346(6210):1243089, 2014.

[FB12] E. D. Feigelson and G. J. Babu. Big data in astronomy. *Significance*, 9(4):22–25, 2012.

[FHT08] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[FL01] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001.

[FM+03] G. Fort, E. Moulines, et al. Convergence of the monte carlo expectation maximization for curved exponential families. *The Annals of Statistics*, 31(4):1220–1259, 2003.

[FS12] V. Filimonov and D. Sornette. Quantifying reflexivity in financial markets: Toward a prediction of flash crashes. *Physical Review E*, 85(5):056108, 2012.

[Fu06] M. C. Fu. Gradient estimation. *Handbooks in operations research and management science*, 13:575–616, 2006.

[FWR+15] M. Farajtabar, Y. Wang, M. G. Rodriguez, S. Li, H. Zha, and L. Song. Coevolve: A joint point process model for information diffusion and network co-evolution. In *Advances in Neural Information Processing Systems*, pages 1954–1962, 2015.

[GBC16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[GGT04] G. Giorgi, A. Guerraggio, and J. Thierfelder. *Mathematics of optimization: smooth and nonsmooth case*. Elsevier, 2004.

[Gla13] P. Glasserman. *Monte Carlo methods in financial engineering*, volume 53. Springer Science & Business Media, 2013.

[GM75] R. Glowinski and A. Marroco. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *Revue française d'automatique, informatique, recherche opérationnelle. Analyse numérique*, 9(2):41–76, 1975.

[GM76] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.

[GM98] A. Gelman and X.-L. Meng. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical science*, pages 163–185, 1998.

[Goe10] J. J. Goeman. L1 penalized estimation in the cox proportional hazards model. *Biometrical journal*, 52(1):70–84, 2010.

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

[GRLS13] M. Gomez-Rodriguez, J. Leskovec, and B. Schölkopf. Modeling information propagation with survival theory. In *International Conference on Machine Learning*, pages 666–674, 2013.

[GTPV14] S. Gopakumar, T. Tran, D. Phung, and S. Venkatesh. Stabilizing sparse cox model using clinical structures in electronic medical records. *arXiv preprint arXiv:1407.6094*, 2014.

[GVL12] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[Hal05] A. R. Hall. *Generalized method of moments*. Oxford University Press, 2005.

[Han82] L. P. Hansen. Large sample properties of generalized method of moments estimators. *Econometrica: Journal of the Econometric Society*, pages 1029–1054, 1982.

[HAV+15] R. Harikandeh, M. O. Ahmed, A. Virani, M. Schmidt, J. Konečnỳ, and S. Sallinen. Stopwasting my gradients: Practical svrg. In *Advances in Neural Information Processing Systems*, pages 2251–2259, 2015.

[Haw71a] A. G. Hawkes. Point spectra of some mutually exciting point processes. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 438–443, 1971.

[Haw71b] A. G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, pages 83–90, 1971.

[HB14] S. J. Hardiman and J.-P. Bouchaud. Branching-ratio approximation for the self-exciting hawkes process. *Physical Review E*, 90(6):062807, 2014.

[HBB13] Hardiman, Stephen J., Bercot, Nicolas, and Bouchaud, Jean-Philippe. Critical reflexivity in financial markets: a hawkes process analysis. *Eur. Phys. J. B*, 86(10):442, 2013.

[Hes69] M. R. Hestenes. Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320, 1969.

[Hin02] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[HO74] A. G. Hawkes and D. Oakes. A cluster process representation of a self-exciting process. *Journal of Applied Probability*, 11(03):493–503, 1974.

[HPK09] C. Hu, W. Pan, and J. T. Kwok. Accelerated gradient methods for stochastic optimization and online learning. In *Advances in Neural Information Processing Systems*, pages 781–789, 2009.

[HRBR+15] N. R. Hansen, P. Reynaud-Bouret, V. Rivoirard, et al. Lasso and probabilistic inequalities for multivariate point processes. *Bernoulli*, 21(1):83–143, 2015.

[HS06] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[HT90] T. Hastie and R. Tibshirani. *Generalized additive models*. Wiley Online Library, 1990.

[HTF09] T. Hastie, R. Tibshirani, and J. Friedman. Overview of supervised learning. In *The elements of statistical learning*, pages 9–41. Springer, 2009.

[HY+05] T. Hayashi, N. Yoshida, et al. On covariance estimation of non-synchronously observed diffusion processes. *Bernoulli*, 11(2):359–379, 2005.

[ISG13] T. Iwata, A. Shah, and Z. Ghahramani. Discovering latent influence in online social activities via shared cascade poisson processes. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 266–274. ACM, 2013.

[JA13] A. Jedidi and F. Abergel. On the stability and price scaling limit of a Hawkes process-based order book model. *Available at SSRN: https://ssrn.com/abstract=2263162*, 2013.

[JHR15] S. Jovanović, J. Hertz, and S. Rotter. Cumulants of hawkes point processes. *Physical Review E*, 91(4):042802, 2015.

[JN+11] A. Juditsky, A. Nemirovski, et al. First order methods for nonsmooth convex large-scale optimization, ii: utilizing problems structure. *Optimization for Machine Learning*, pages 149–183, 2011.

[JZ13] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.

[Kal57] N. Kaldor. A model of economic growth. *The economic journal*, 67(268):591–624, 1957.

[KLRT16] J. Konečnỳ, J. Liu, P. Richtárik, and M. Takáč. Mini-batch semi-stochastic gradient descent in the proximal setting. *IEEE Journal of Selected Topics in Signal Processing*, 10(2):242–255, 2016.

[KR13] J. Konecnỳ and P. Richtárik. Semi-stochastic gradient descent methods. *arXiv preprint arXiv:1312.1666*, 2(2.1):3, 2013.

[KW+52] J. Kiefer, J. Wolfowitz, et al. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.

[Lan12] G. Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1):365–397, 2012.

[Lar07] J. Large. Measuring the resiliency of an electronic limit order book. *Journal of Financial Markets*, 10(1):1–25, 2007.

[LHKD+07] S. Loi, B. Haibe-Kains, C. Desmedt, F. Lallemand, A. M. Tutt, C. Gillet, P. Ellis, A. Harris, J. Bergh, J. A. Foekens, et al. Definition of clinically distinct molecular subtypes in estrogen receptor–positive breast carcinomas through genomic grade. *Journal of clinical oncology*, 25(10):1239–1246, 2007.

[LJ17] L. Lei and M. Jordan. Less than a single pass: Stochastically controlled stochastic gradient. In *Artificial Intelligence and Statistics*, pages 148–156, 2017.

[LM11] E. Lewis and G. Mohler. A nonparametric em algorithm for multiscale hawkes processes. *Journal of Nonparametric Statistics*, 2011.

[LMH15] H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems*, pages 3384–3392, 2015.

[LMP+01] J. Lafferty, A. McCallum, F. Pereira, et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[LN89] D. C. Liu and J. Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528, 1989.

[LV14] R. Lemonnier and N. Vayatis. Nonparametric markovian learning of triggering kernels for mutually exciting and mutually inhibiting multivariate hawkes processes. In *Machine Learning and Knowledge Discovery in Databases*, pages 161–176. Springer, 2014.

[MB+12] A. McAfee, E. Brynjolfsson, et al. Big data: the management revolution. *Harvard business review*, 90(10):60–68, 2012.

[MD13] T. B. Murdoch and A. S. Detsky. The inevitable application of big data to health care. *Jama*, 309(13):1351–1352, 2013.

[MMCB13] S. Mittal, D. Madigan, J. Q. Cheng, and R. S. Burd. Large-scale parametric survival analysis. *Statistics in medicine*, 32(23):3955–3971, 2013.

[MSB+11] G. O. Mohler, M. B. Short, P. J. Brantingham, F. P. Schoenberg, and G. E. Tita. Self-exciting point process modeling of crime. *Journal of the American Statistical Association*, 2011.

[Mur12] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[Nit14] A. Nitanda. Stochastic proximal gradient descent with acceleration techniques. In *Advances in Neural Information Processing Systems*, pages 1574–1582, 2014.

[NJLS09] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

[NM94] W. K. Newey and D. McFadden. Large sample estimation and hypothesis testing. *Handbook of econometrics*, 4:2111–2245, 1994.

[NN94] Y. Nesterov and A. Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

[Oga81] Y. Ogata. On lewis' simulation method for point processes. *IEEE Transactions on Information Theory*, 27(1):23–31, 1981.

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

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

[PB11] J. Peters and J. A. Bagnell. Policy gradient methods. In *Encyclopedia of Machine Learning*, pages 774–776. Springer, 2011.

[PBLJ15] A. Podosinnikova, F. Bach, and S. Lacoste-Julien. Rethinking lda: moment matching for discrete ica. In *Advances in Neural Information Processing Systems*, pages 514–522, 2015.

[PH07] M. Y. Park and T. Hastie. L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):659–677, 2007.

[Pow67] M. J. Powell. *" A method for non-linear constraints in minimization problems"*. UKAEA, 1967.

[PSCR11] V. Pernice, B. Staude, S. Cardanobile, and S. Rotter. How structure determines correlations in neuronal networks. *PLoS computational biology*, 7(5):e1002059, 2011.

[RBL17] M. Rambaldi, E. Bacry, and F. Lillo. The role of volume in order book dynamics: a multivariate hawkes process analysis. *Quantitative Finance*, 17(7):999–1020, 2017.

[RBRGTM14] P. Reynaud-Bouret, V. Rivoirard, F. Grammont, and C. Tuleau-Malot. Goodness-of-fit tests and nonparametric adaptive estimation for spike train analysis. *The Journal of Mathematical Neuroscience*, 4(1):3, 2014.

[RBS10] P. Reynaud-Bouret and S. Schbath. Adaptive estimation for hawkes processes; application to genome analysis. *The Annals of Statistics*, 38(5):2781–2822, 2010.

[Ric12] S. Richards. A handbook of parametric survival models for actuarial use. *Scandinavian Actuarial Journal*, 2012(4):233–257, 2012.

[RM51] H. Robbins and S. Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[RMW14] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

[Rob04] C. P. Robert. *Monte carlo methods*. Wiley Online Library, 2004.

[Rod13] M. G. Rodriguez. *Structure and Dynamics of Diffusion Networks*. PhD thesis, Stanford University, 2013.

[ROF92] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

[RR10] C. Y. Robert and M. Rosenbaum. A new approach for the dynamics of ultra-high-frequency data: The model with uncertainty zones. *Journal of Financial Econometrics*, 9(2):344–366, 2010.

[RSB12] N. L. Roux, M. Schmidt, and F. R. Bach. A stochastic gradient method with an exponential convergence _rate for finite training sets. In *Advances in Neural Information Processing Systems*, pages 2663–2671, 2012.

[RZ11] J.-J. Ren and M. Zhou. Full likelihood inferences in the cox model: an empirical likelihood approach. *Annals of the Institute of Statistical Mathematics*, 63(5):1005–1018, 2011.

[SAD+16] V. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, and F. Menczer. The darpa twitter bot challenge. *Computer*, 49(6):38–46, 2016.

[SBA+15] M. Schmidt, R. Babanezhad, M. Ahmed, A. Defazio, A. Clifton, and A. Sarkar. Non-uniform stochastic average gradient method for training conditional random fields. In *Artificial Intelligence and Statistics*, pages 819–828, 2015.

[SFHT11] N. Simon, J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for cox's proportional hazards model via coordinate descent. *Journal of statistical software*, 39(5):1, 2011.

[She15] P.-s. Shen. Full likelihood inference in the cox model with ltrc data when covariates are discrete. *Statistics*, 49(3):602–613, 2015.

[SKJP09] I. Sohn, J. Kim, S.-H. Jung, and C. Park. Gradient lasso for cox proportional hazards model. *Bioinformatics*, 25(14):1775–1781, 2009.

[SLRB17] M. Schmidt, N. Le Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2):83–112, 2017.

[SSZ12] S. Shalev-Shwartz and T. Zhang. Proximal stochastic dual coordinate ascent. *arXiv preprint arXiv:1211.2717*, 2012.

[SSZ13] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb):567–599, 2013.

[T+97] R. Tibshirani et al. The lasso method for variable selection in the cox model. *Statistics in medicine*, 16(4):385–395, 1997.

[TA14] P. Toulis and E. M. Airoldi. Implicit stochastic gradient descent for principled estimation with large datasets. *ArXiv e-prints*, 2014.

[Tib96] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[Tok11] I. M. Toke. "Market making" in an order book model and its impact on the spread. In *Econophysics of Order-driven Markets*, pages 49–64. Springer, 2011.

[Vap13] V. Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[VDVHVV+02] M. J. Van De Vijver, Y. D. He, L. J. Van't Veer, H. Dai, A. A. Hart, D. W. Voskuil, G. J. Schreiber, J. L. Peterse, C. Roberts, M. J. Marton, et al. A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medicine*, 347(25):1999–2009, 2002.

[VNMN14] R. Van Noorden, B. Maher, and R. Nuzzo. The top 100 papers. *Nature*, 514(7524):550, 2014.

[WT09] D. M. Witten and R. Tibshirani. Survival analysis with high-dimensional covariates. *Statistical methods in medical research*, 2009.

[XFZ16] H. Xu, M. Farajtabar, and H. Zha. Learning granger causality for hawkes processes. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1717–1726, 2016.

[Xia10] L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11(Oct):2543–2596, 2010.

[XZ14] L. Xiao and T. Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

[YZ12] Y. Yang and H. Zou. A cocktail algorithm for solving the elastic net penalized cox's regression in high dimensions. *Statistics and its Interface*, 6(2):167–173, 2012.

[YZ13] S.-H. Yang and H. Zha. Mixture of mutually exciting processes for viral diffusion. In *Proceedings of the International Conference on Machine Learning*, 2013.

[ZO00] T. Zhang and F. Oles. The value of unlabeled data for classification problems. In *Proceedings of the Seventeenth International Conference on Machine Learning,(Langley, P., ed.)*, pages 1191–1198, 2000.

[Zou06] H. Zou. The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429, 2006.

[ZZS13] K. Zhou, H. Zha, and L. Song. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *AISTATS*, volume 31, pages 641–649, 2013.

université PARIS-SACLAY | École doctorale de mathématiques Hadamard (EDMH)

Titre : Apprentissage statistique pour séquences d'évènements à l'aide de processus ponctuels

Mots-clefs : processus ponctuels, causalité, optimisation convexe, finance quantitative.

Résumé : Cette thèse est divisée en trois parties. La première introduit un nouvel algorithme d'optimisation qui permet d'estimer le vecteur de paramètre de la régression de Cox lorsque le nombre d'observations est très important. Notre algorithme est basé sur l'algorithme SVRG et utilise une méthode MCMC pour approximer la direction de descente. Nous avons prouvé des vitesses de convergence pour notre algorithme et avons montré sa performance numérique sur des jeux de données simulés et issus de monde réel. La deuxième partie montre que la causalité au sens de Hawkes peut être estimée de manière non-paramétrique grâce aux cumulants intégrés du processus ponctuel multivarié. Nous avons développé deux méthodes d'estimation des intégrales des noyaux du processus de Hawkes, sans faire d'hypothèse sur la forme de ces noyaux. Nos méthodes sont plus rapides et plus robustes, vis-à-vis de la forme des noyaux, par rapport à l'état de l'art. Nous avons démontré la consistence statistique de la première méthode, et avons montré que la deuxième peut être réduite à un problème d'optimisation convexe. La dernière partie met en lumière les dynamiques de carnet d'ordre grâce à la première méthode d'estimation non-paramétrique introduite dans la partie précédente. Nous avons utilisé des données du marché à terme EUREX, défini de nouveaux modèles de carnet d'ordre et appliqué la méthode d'estimation sur ces processus ponctuels. Les résultats obtenus sont très satisfaisants et cohérents avec une analyse économétrique, et prouve que la méthode que nous avons développée permet d'extraire une structure à partir de données complexes telles que celles issues de la finance haute-fréquence.

Title : Learning from Sequences with Point Processes

Keywords : point processes, causality, convex optimization, quantitative finance.

Abstract : This thesis is divided into three parts. The first focuses on a new optimization algorithm we have developed. It allows to estimate the parameter vector of the Cox regression when the number of observations is very important. Our algorithm is based on the SVRG algorithm and uses a MCMC method to approximate the descent direction. We have proved convergence rates for our algorithm and have shown its numerical performance on simulated and real world data sets. The second part shows that the Hawkes causality can be estimated in a non-parametric way from the integrated cumulants of the multivariate point process. We have developed two methods for estimating the integrals of the kernels of the Hawkes process, without making any hypothesis about the shape of these kernels. Our methods are faster and more robust, with respect to the shape of the kernel, compared to the state-of-the-art. We have demonstrated the statistical consistency of the first method, and have shown that the second method can be reduced to a convex optimization problem. The last part highlights the dynamics of the order book thanks to the first non-parametric estimation method introduced in the previous section. We used EUREX futures data, defined new order book models and applied the estimation method on these point processes. The results obtained are very satisfactory and consistent with an econometric analysis, and prove that the method that we have developed makes it possible to extract a structure from data as complex as those resulting from high-frequency finance.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France