Time-oriented statistical machine learning

NIST ITL Talk, Summer 2017

Stéphane Gaïffas

Example 1. Social networks. Understand who is influencing **twitter**: based on the timestamps patterns of messages, **web-data**: publication activity of websites/blogs

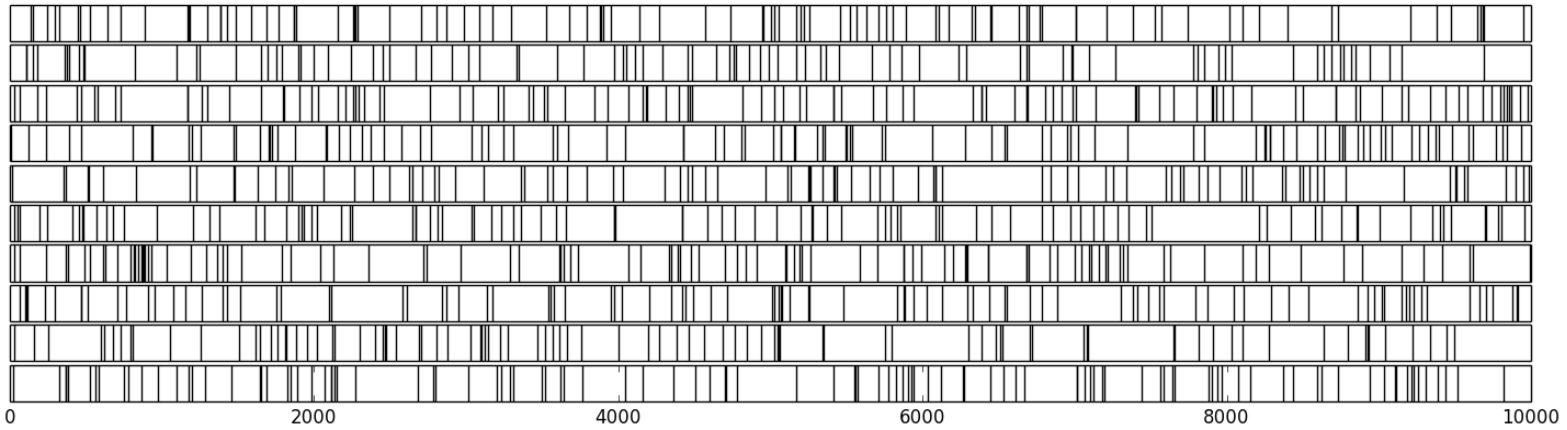
Example 2. High Frequency Finance.
From zoomed financial signals ($\Delta t \approx 1$ms, upward / downward price proves and other order book features), build a "causality map"

Example 3 (??). Arrival time of queries for items on a network
These are not homogeneous, related to human behavior (tendencies, memes, etc.)

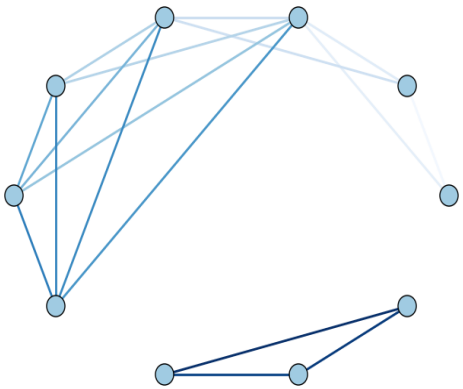
Example 4. Health-care. Impact of some health events to other health events (all being timestamped, longitudinal data)

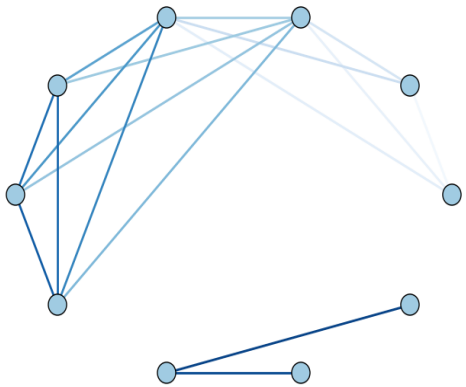
Introduction

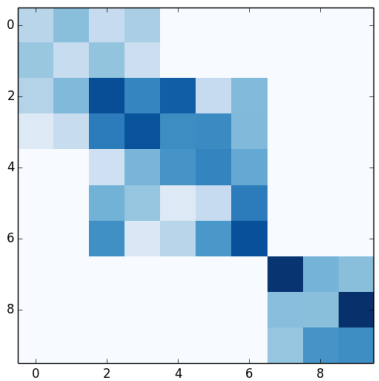
From:



Build:







Setting

- For each node $i \in I = \{1, \ldots, d\}$ we have a set Z^i of *events*
- Any $\tau \in Z^i$ is the occurence time of an event related to i

Counting process

- Put $N_t = [N_t^1 \cdots N_t^d]^\top$
- $N_t^i = \sum_{\tau \in Z^i} \mathbf{1}_{\tau \leq t}$

Intensity

- Stochastic intensities $\lambda_t = [\lambda_t^1 \cdots \lambda_t^d]^\top$, $\lambda_t^i =$ intensity of N_t^i

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

- $\lambda_t^i =$ instantaneous rate of event occurence at time t for node i
- λ_t characterizes the distribution of N_t [Daley et al. 2007]
- Patterns can be captured by *putting structure* on λ_t

Scaling

- We observe N_t on $[0, T]$. "Asymptotics" in $T \to +\infty$. d is "large"

The Hawkes process

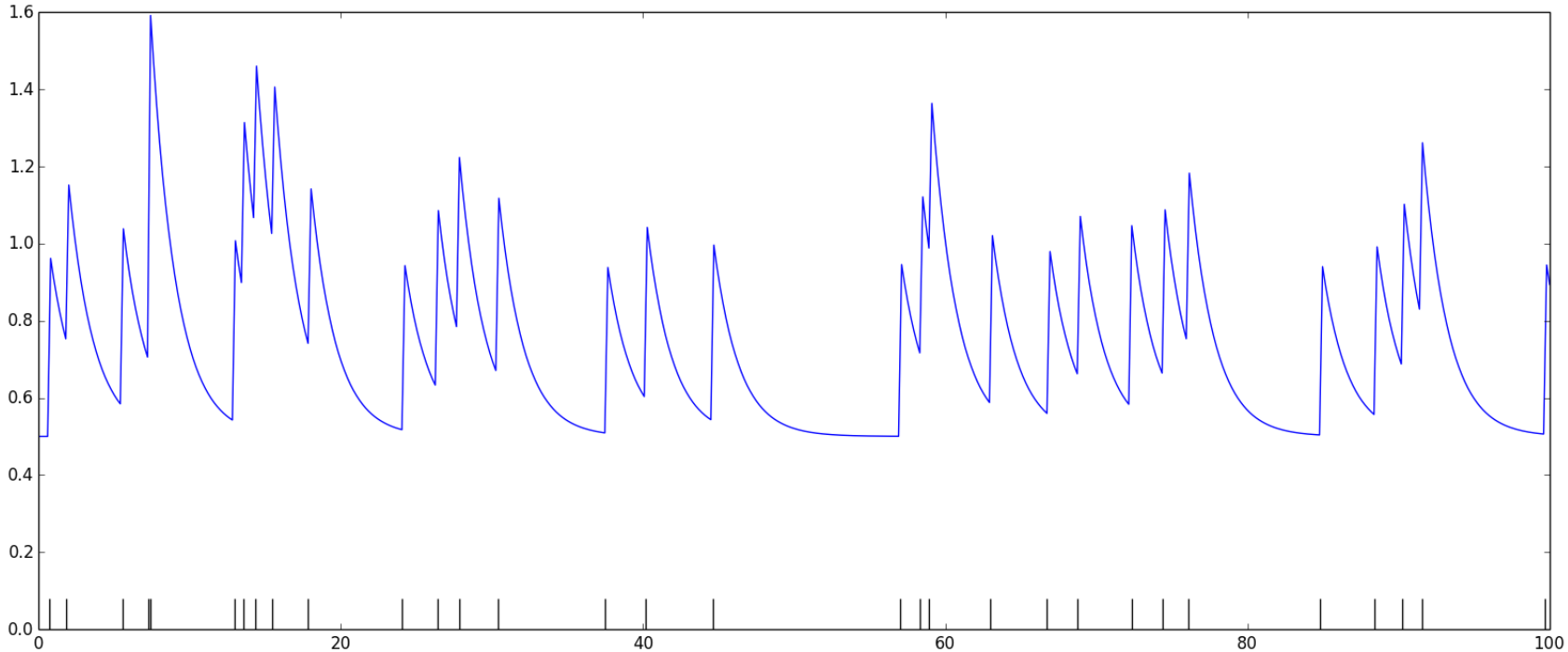
- A particular structure for λ_t: auto-regression
- N_t is called a *Hawkes* process [Hawkes 1971] if

$$\lambda_t^i = \mu_i + \sum_{j=1}^{d} \int_0^t \varphi^{ij}(t - t') dN_{t'}^j = \mu_i + \sum_{j=1}^{d} \sum_{t' \in Z_j : t' < t} \varphi^{ij}(t - t')$$

- $\mu_i \in \mathbb{R}^+$ exogenous intensity
- φ^{ij} non-negative integrable and causal (support $\mathbb{R}_+$) functions
- φ^{ij} are called *kernels*. Encodes the impact of an action by node j on the activity of node i
- Captures *auto-excitation* and *cross-excitation* across nodes, a phenomenon observed in social networks [Crane et al. 2008]

A simple parametrization of the MHP

For $d = 1$, $K = 1$ and $\varphi^{11}(t) = e^{-1}$, intensity $\lambda_{\theta,t}$ looks like:



Stability condition

- Introduce

$$\boldsymbol{G}^{ij} = \int_0^{+\infty} \varphi^{ij}(t)dt$$

- Spectral norm must satisfy $\|\boldsymbol{G}\| < 1$ to ensure stability and stationarity of the process

Sum of exponentials **parametric** model:

$$\lambda^i_{\theta,t} = \mu_i + \int_{(0,t)} \sum_{j=1}^{d} \sum_{k=1}^{K} a^k_{ij} \times \alpha_k e^{-\alpha_k (t-s)} dN^j_s$$

for $i \in \{1, \dots, d\}$ with $\alpha_1, \dots, \alpha_K > 0$ given and parameters to infer are $\theta = [\mu, \boldsymbol{A}]$ with

- baselines $\mu = [\mu_1 \cdots \mu_d]^\top \in \mathbb{R}^d_+$
- interactions $\boldsymbol{A} = [a_{ij}]_{1 \le i,j \le d} \in \mathbb{R}^{d \times d}_+ =$ "adjacency matrix"

Brief history

- Introduced in Hawkes 1971
- Earthquakes and geophysics [Kagan and Knopoff 1981], [Zhuang et al. 2012]
- Genomics [Reynaud-Bouret and Schbath 2010]
- High-frequency Finance [Bacry et al. 2013]
- Terrorist activity [Mohler et al. 2011, Porter and White 2012]
- Neurobiology [Hansen et al. 2012]
- Social networks [Carne and Sornette 2008], [Zhou et al.2013]
- And even FPGA-based implementation [Guo and Luk 2013]

A brief history of MHP

THE GENESIS BLOCK

Digital currency research and data

HOME | NEWS | MINING | TRADING | ECONOMICS | REGULATION | BUSINESSES | BI

Home / Bitcoin 201 / Analyzing Trade Clustering To Predict Price Movement In Bitcoin Trading

Analyzing Trade Clustering To Predict Price Movement In Bitcoin Trading

Sep 19, 2013 Posted By Jonathan Heusser In Bitcoin 201, Economics, Featured, News, Trading Tagged Analysis, Bitcoin Trading, Hawkes Process, Jonathan Heusser, London, Price, Trading

What do we want to do?

- Deal with large number of events and large dimension d (number of nodes)
- End up with a *tractable* and *scalable* optimization problem

Goodness-of-fit functionals. Two choices: **minus** log-**likelihood**

$$-\ell_T(\theta) = \frac{1}{T}\sum_{i=1}^{d}\Big\{\int_0^T \lambda^i_{\theta,t}dt - \int_0^T \log\lambda^i_{\theta,t}dN^i_t\Big\}$$

or **least-squares**

$$R_T(\theta) = \frac{1}{T}\sum_{i=1}^{d}\Big\{\int_0^T (\lambda^i_{\theta,t})^2dt - 2\int_0^T \lambda^i_{\theta,t}dN^i_t\Big\}$$

Paper. E. Bacry, S. G., J.-F. Muzy, *A generalization error bound for sparse and low-rank multivariate Hawkes processes*, in revision in Journal of Machine Learning

- Parametric setting $\varphi^{ij}(t) = (\boldsymbol{A})_{ij} \times h(t)$
- Low-rank and sparsity inducing penalization on $\boldsymbol{A}$
- Introduces a sharp tuning of the penalizations using data-driven weights
- Leads to optimal error bounds for penalized least-squares (sharp sparse oracle inequality)

Prior assumptions

- Users are basically inactive and react mostly if stimulated:

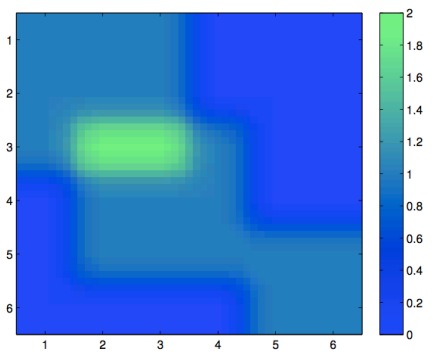
  μ is sparse

- Everybody does not interact with everybody:

  $\boldsymbol{A}$ is sparse

- Interactions have community structure, possibly overlapping, a small number of factors explain interactions:

  $\boldsymbol{A}$ is low-rank



Standard convex relaxations
(Tibshirani (01), Srebro et al. (05), Bach (08), Candès & Tao (09), etc.)

- Convex relaxation of $\|\boldsymbol{A}\|_0 = \sum_{ij} \mathbf{1}_{\boldsymbol{A}_{ij}>0}$ is ℓ_1-norm:

$$\|\boldsymbol{A}\|_1 = \sum_{ij} |\boldsymbol{A}_{ij}|$$

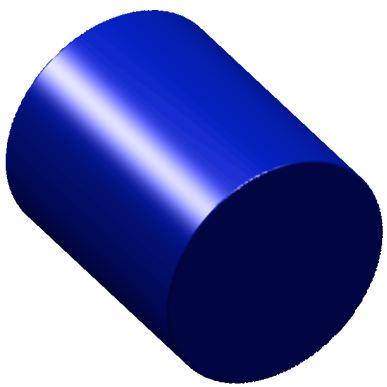
- Convex relaxation of rank is trace-norm:

$$\|\boldsymbol{A}\|_* = \sum_{j} \sigma_j(\boldsymbol{A}) = \|\sigma(\boldsymbol{A})\|_1$$

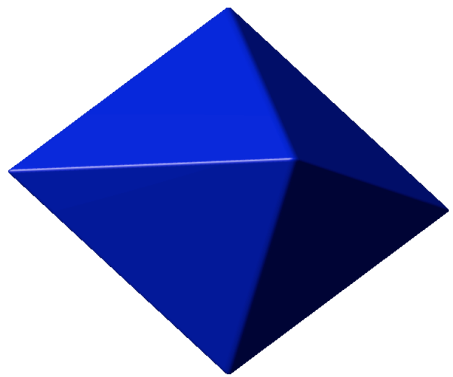
where $\sigma_1(\boldsymbol{A}) \geq \cdots \geq \sigma_d(\boldsymbol{A})$ singular values of $\boldsymbol{A}$

We use the following penalizations

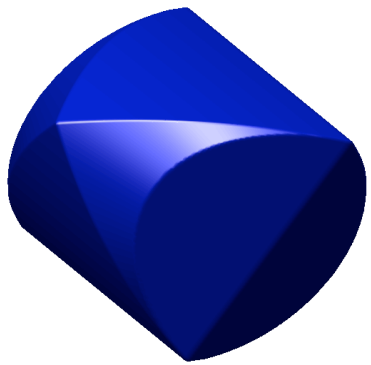
- Use ℓ_1 penalization on μ
- Use ℓ_1 penalization on $\boldsymbol{A}$
- Use trace-norm penalization on $\boldsymbol{A}$



$\{\boldsymbol{A} : \|A\|_* \leq 1\}$



$\{\boldsymbol{A} : \|\boldsymbol{A}\|_1 \leq 1\}$



$\{\boldsymbol{A} : \|\boldsymbol{A}\|_1 + \|\boldsymbol{A}\|_* \leq 1\}$

Balls are on the set of 2×2 symmetric matrices identified with $\mathbb{R}^3$.

Leads to

$$\hat{\theta} = (\hat{\mu}, \hat{\boldsymbol{A}}) \in \underset{\theta=(\mu,\boldsymbol{A})\in\mathbb{R}_+^d\times\mathbb{R}_+^{d\times d}}{\operatorname{argmin}} \big\{R_T(\theta) + \operatorname{pen}(\theta)\big\},$$

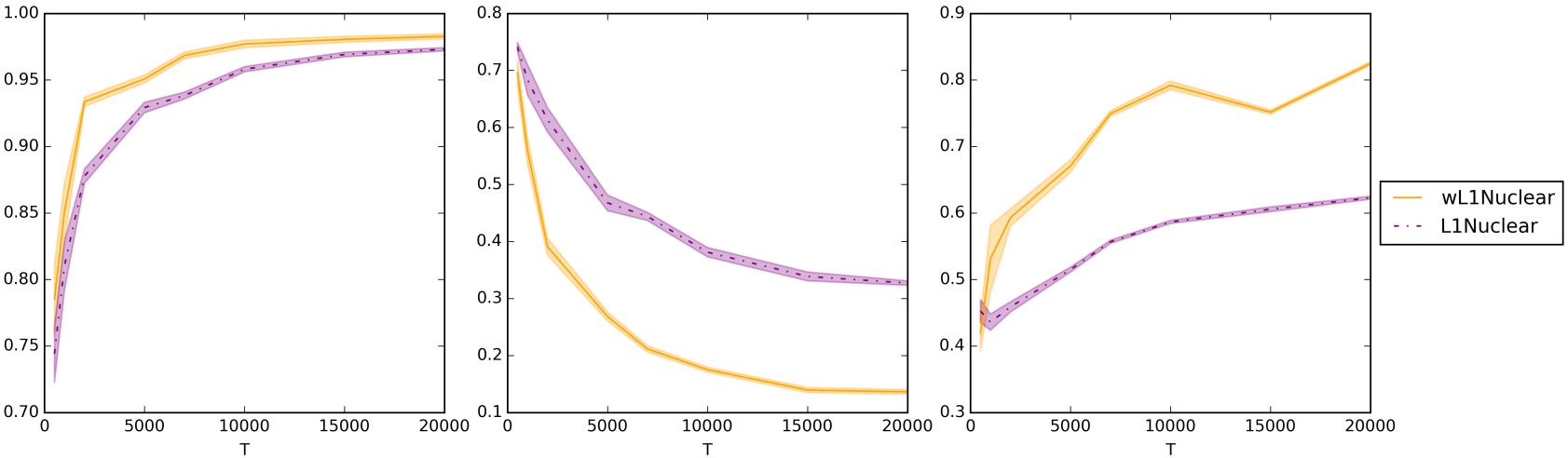
with penalization

$$\operatorname{pen}(\theta) = \tau_1\|\mu\|_1 + \gamma_1\|\boldsymbol{A}\|_1 + \gamma_*\|\boldsymbol{A}\|_*$$

The **features scaling problem**

- Features scaling is necessary for "linear approaches" in supervised learning
- No features and labels here!
- We solve this by sharp data-driven tuning of the penalization terms
- Required a new theory for random matrices with entries that are continuous-time martingales

Contribution 1. Dimension reduction for MHP



Left: AUC; *Middle*: Estimation error; *Right*: Kandall rank correlation

A strong theoretical guarantee

- Recall $\langle \lambda_1, \lambda_2 \rangle_T = \frac{1}{T} \sum_{i=1}^{d} \int_0^T \lambda_{1,t}^i \lambda_{2,t}^i dt$ and $\|\lambda\|_T^2 = \langle \lambda, \lambda \rangle_T$
- Assume RE in our setting (Restricted Eigenvalues, Compressed Sensing literature)

Theorem. We have

$$\|\lambda_{\hat{\theta}} - \lambda^*\|_T^2 \leq \inf_{\theta} \left\{ \|\lambda_\theta - \lambda^*\|_T^2 + \kappa(\theta)^2 \left(\frac{5}{4} \|(\hat{w})_{\text{supp}(\mu)}\|_2^2 + \frac{9}{8} \|(\hat{\boldsymbol{W}})_{\text{supp}(\boldsymbol{A})}\|_F^2 + \frac{9}{8} \hat{w}_*^2 \, \text{rank}(\boldsymbol{A}) \right) \right\}$$

with a probability larger than $1 - 146e^{-x}$.

Roughly, $\hat{\theta}$ achieves an optimal tradeoff between approximation and complexity given by

$$\frac{\|\mu\|_0 \log d}{T} \max_i \bar{N}^i([0,T]) + \frac{\|\boldsymbol{A}\|_0 \log d}{T} \max_{ij} \hat{v}_T^{ij} + \frac{\text{rank}(A) \log d}{T} \lambda_{\max}(\hat{\boldsymbol{V}}_T)$$

- Complexity measured both by sparsity and rank
- Convergence has shape $(\log d)/T$, where $T =$ length of the observation interval
- Terms are balanced by "empirical variance" terms

Paper. E. Bacry, S. G. and J-F Muzy, *Concentration inequalities for matrix martingales in continuous time*, Probability Theory and Related Fields (2017)

Consider a $m \times n$ matrix-martingale given by

$$\boldsymbol{Z}_t = \int_0^t \mathbb{T}_s \circ d\boldsymbol{M}_s,$$

with $(\boldsymbol{M}_t)_{t\geq 0}$ "white" Brownian random matrix and $(\mathbb{T}_t)_{t\geq 0}$ rank-4 preditable tensor. Then

$$\mathbb{P}\left[\|\boldsymbol{Z}_t\|_{\mathrm{op}} \geq \sqrt{2v(x+\log(m+n))}\ ,\ \sigma^2(\boldsymbol{Z}_t) \leq v\right] \leq e^{-x}$$

for any $v, x > 0$ with

$$\sigma^2(\boldsymbol{Z}_t) = \max\left(\Big\|\sum_{j=1}^n \langle \boldsymbol{Z}_{\bullet,j}\rangle_t\Big\|_{\mathrm{op}}, \Big\|\sum_{j=1}^m \langle \boldsymbol{Z}_{j,\bullet}\rangle_t\Big\|_{\mathrm{op}}\right).$$

- Strong generalization of previously known inequalities to continuous time (Tropp 2011)
- Very different approach (random matrix tools + stochastic calculus)
- Also the "Poissonian" case: martingale with sub-exponential jumps (counting process, Hawkes processes)

Interesting particular case (previously unknown!). Consider $\boldsymbol{P} = [P_{ij}]$ a $n \times m$ random matrix where P_{ij} is Poisson(λ_{ij}) and put $\boldsymbol{\lambda} = [\lambda_{ij}]$. Then

$$\mathbb{P}\left(\|\boldsymbol{N} - \boldsymbol{\lambda}\|_{\mathrm{op}} \geq \sqrt{2(\|\boldsymbol{\lambda}\|_{1,\infty} \vee \|\boldsymbol{\lambda}\|_{\infty,1})x} + \frac{x}{3} \right) \leq (n+m)e^{-x}$$

for any $x > 0$, where $\|\boldsymbol{\lambda}\|_{1,\infty}$ (resp. $\|\boldsymbol{\lambda}\|_{\infty,1}$) stands for the maximum ℓ_1-norm of rows (resp. columns)

Paper. Achab et al. *Uncovering Causality from Multivariate Hawkes Integrated Cumulants*, International Conference on Machine Learning (2017) and Journal of Machine Learning (2017)

A reminder

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

Idea

- Direct estimation of $(\boldsymbol{G})_{ij} = \int \varphi^{ij}$ without estimation of φ^{ij}
- Actually, $\boldsymbol{G}$ encodes Granger causality between nodes!

Cumulant matching method for estimation of $\boldsymbol{G}$

- Compute estimates of the third order cumulants of the process
- Find $\boldsymbol{G}$ that matches these empirical cumulants
- Highly non-convex problem: polynomial or order 10 with respect to the entries of $(\boldsymbol{I} - \boldsymbol{G})^{-1}$
- Not so hard, local minima turns out to be good (deep learning literature)
- We prove statistical consistency of the method

Why order **three** and not two?

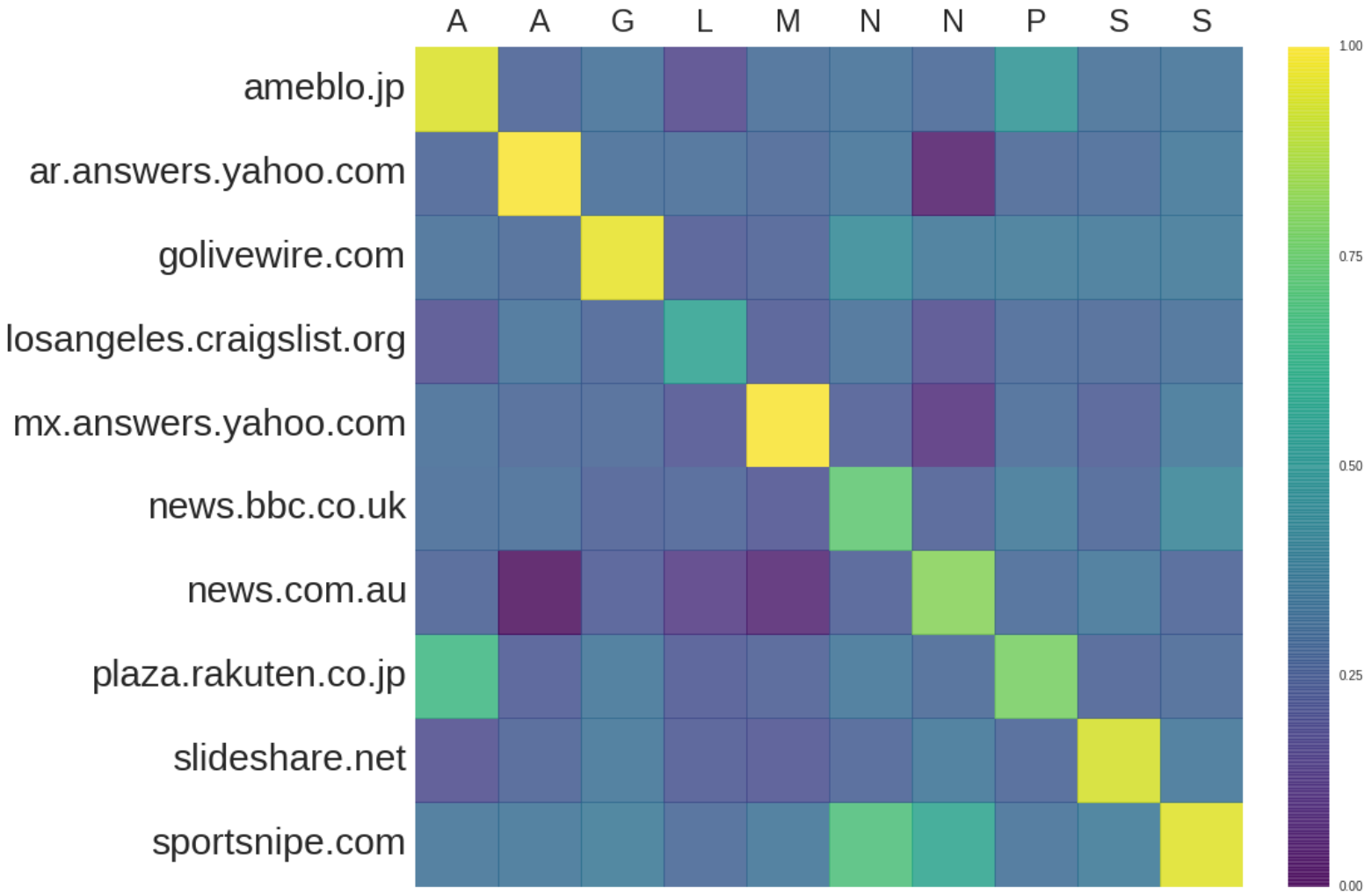
- integrated covariance (order two) contains only symmetric information, and is thus unable to provide causal information
- the skewness of the process breaks the symmetry between past and future so to uniquely fix $\boldsymbol{G}$

Experiment with MemeTracker dataset

- keep the 200 most active sites
- contains publication times of articles in many websites/blogs, with hyperlinks
- $\approx$ 8 millions events
- Use hyperlinks to establish an estimated ground truth for the matrix $\boldsymbol{G}$

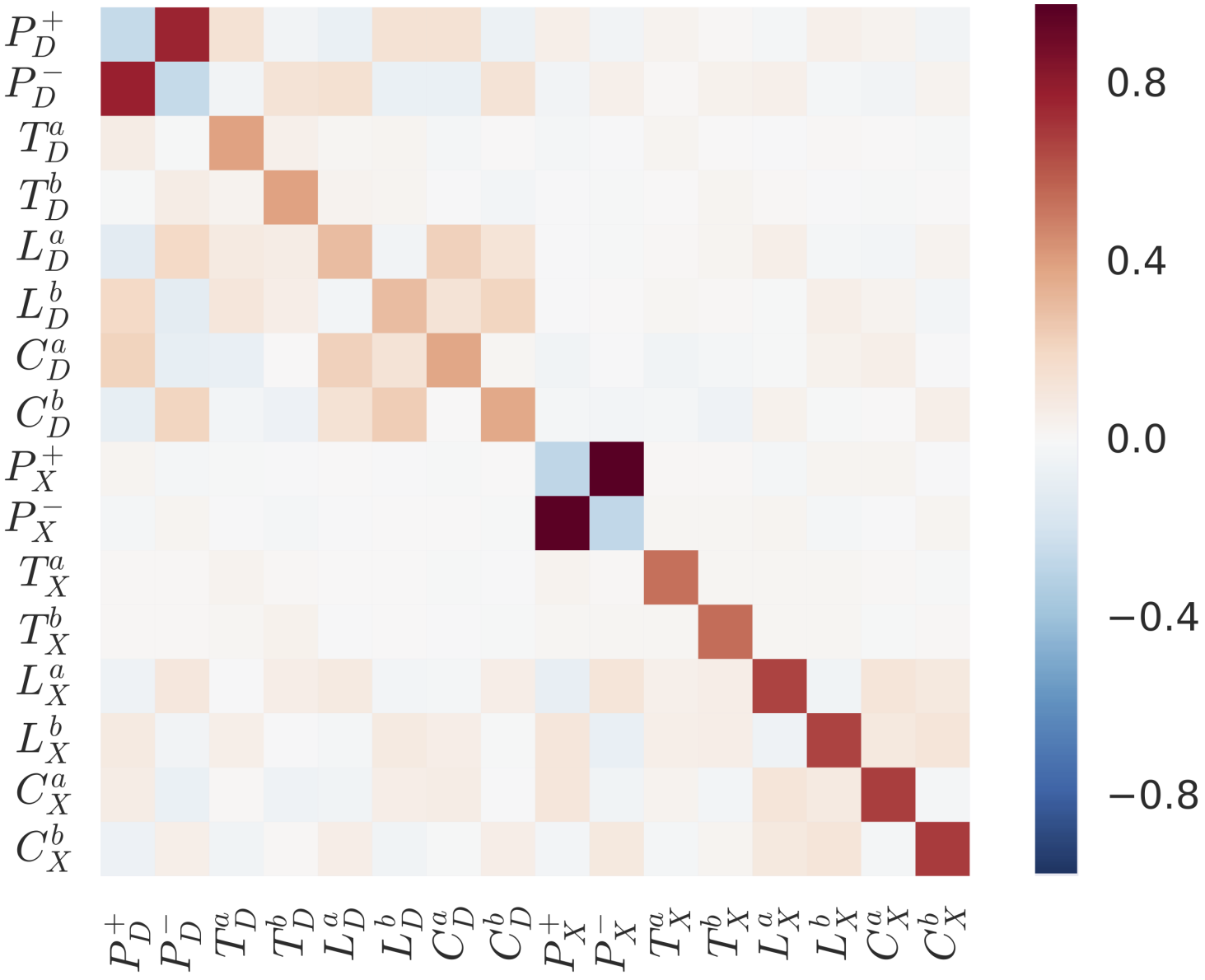
Method	ODE	GC	ADM4	NPHC
RelErr	0.162	0.19	0.092	**0.071**
MRankCorr	0.07	0.053	0.081	**0.095**
Time (s)	2944	2780	2217	**38**

Experiment with MemeTracker dataset



$\widehat{G}$

High-frequency financial data (DAX order book dynamics)



1. Motivation and examples

2. Hawkes processes
   - Introduction
   - Dimension reduction for MHP
   - Random matrix theory
   - Causality maps
   - Accelerating training time

3. Software

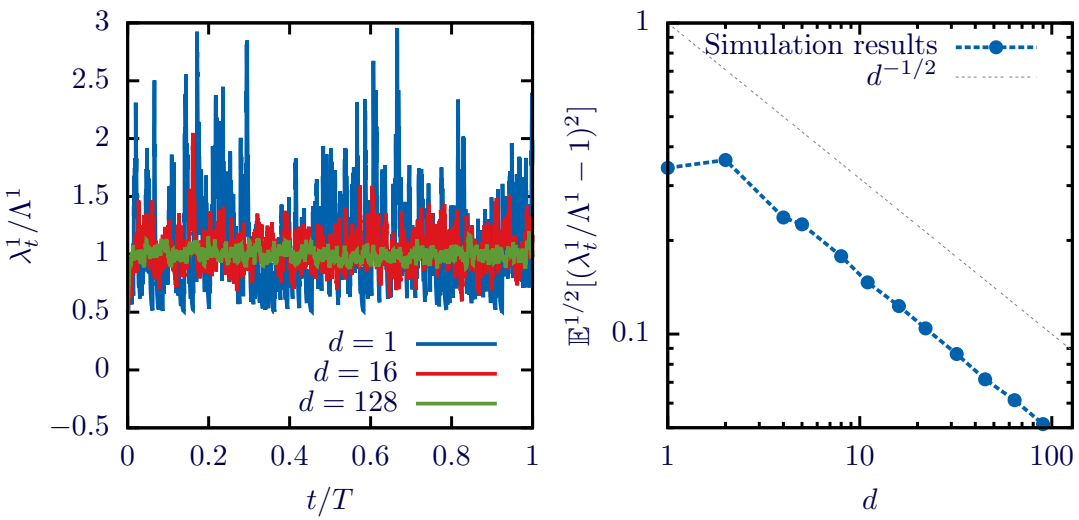
4. Healthcare
   - The CNAM project
   - A new infrastructure
   - A new methodology

Contribution 4. Accelerating training time of MHP

Paper. E. Bacry, S. G., J.-F. Muzy, I. Mastromatteo, *Mean-field inference of Hawkes point processes*, Journal of Physics A, 2016

- Dedicated optimization algorithm for the Hawkes MLE with large number of nodes
- Based on a mean-field approximation
- Partially understood (proof on toy cases)
- Improves state-of-the-art by orders of magnitude

Mean-Field approximation (large number of nodes d helps!)

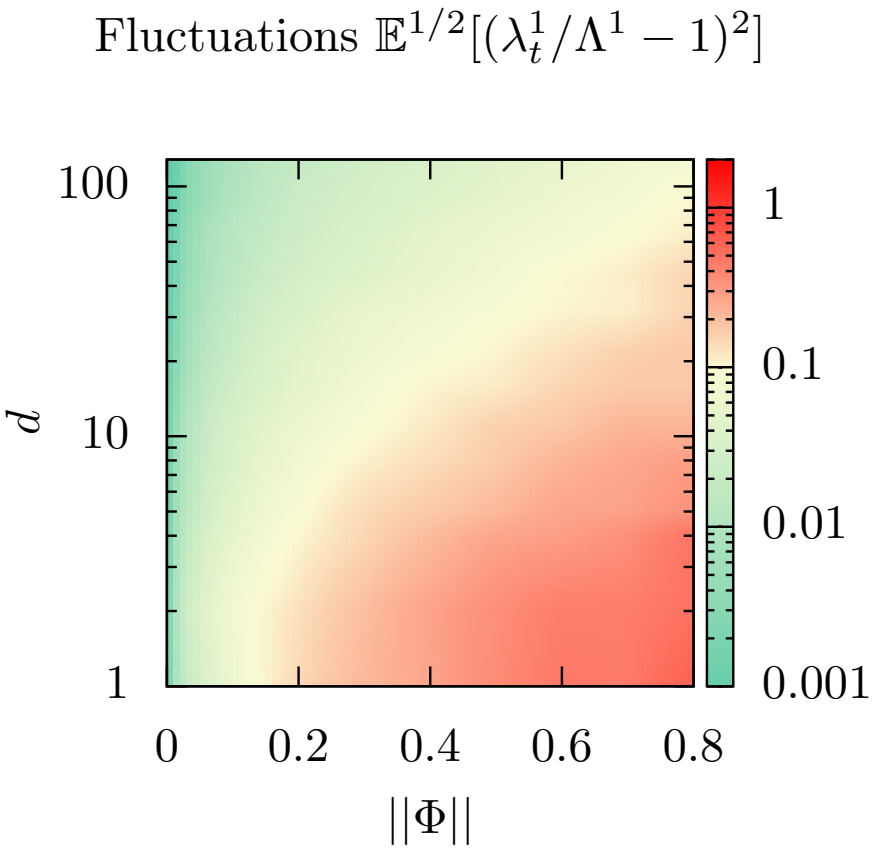


Use the quadratic approximation

$$\log \lambda_t^i \approx \log \Lambda^i + \frac{\lambda_t^i - \Lambda^i}{\Lambda^i} - \frac{(\lambda_t^i - \Lambda^i)^2}{2(\Lambda^i)^2}$$

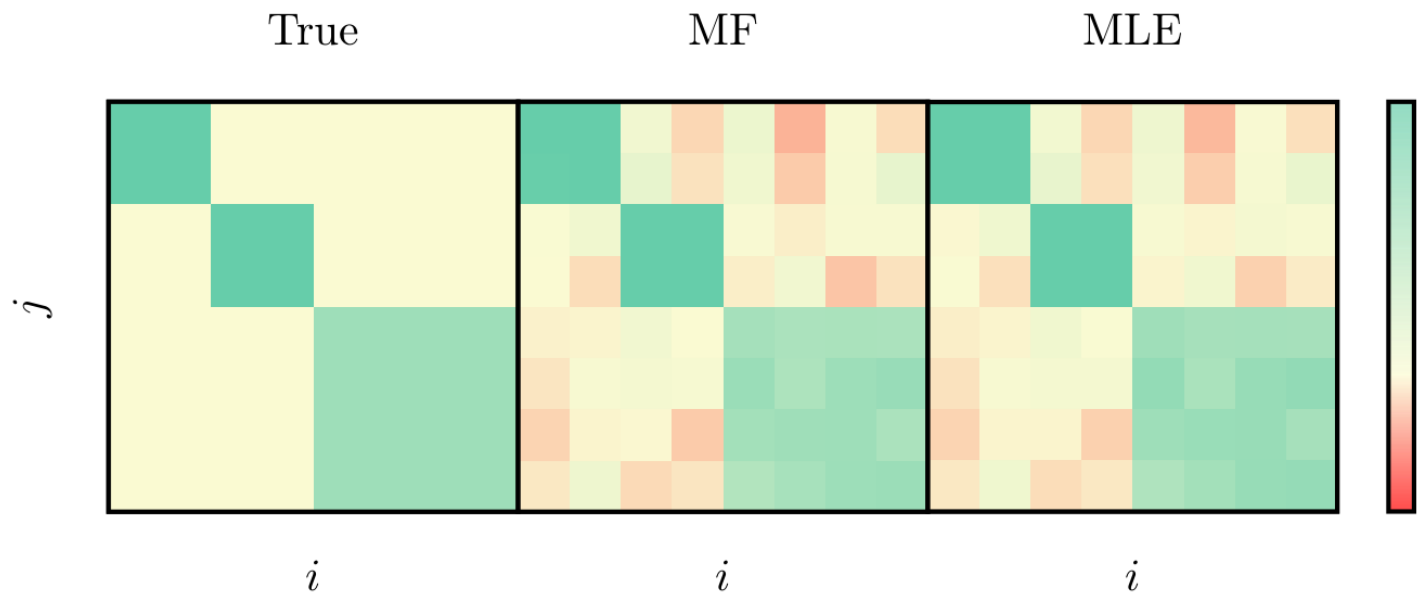
in the log-likelihood

$\Rightarrow$Reduces inference to linear systems



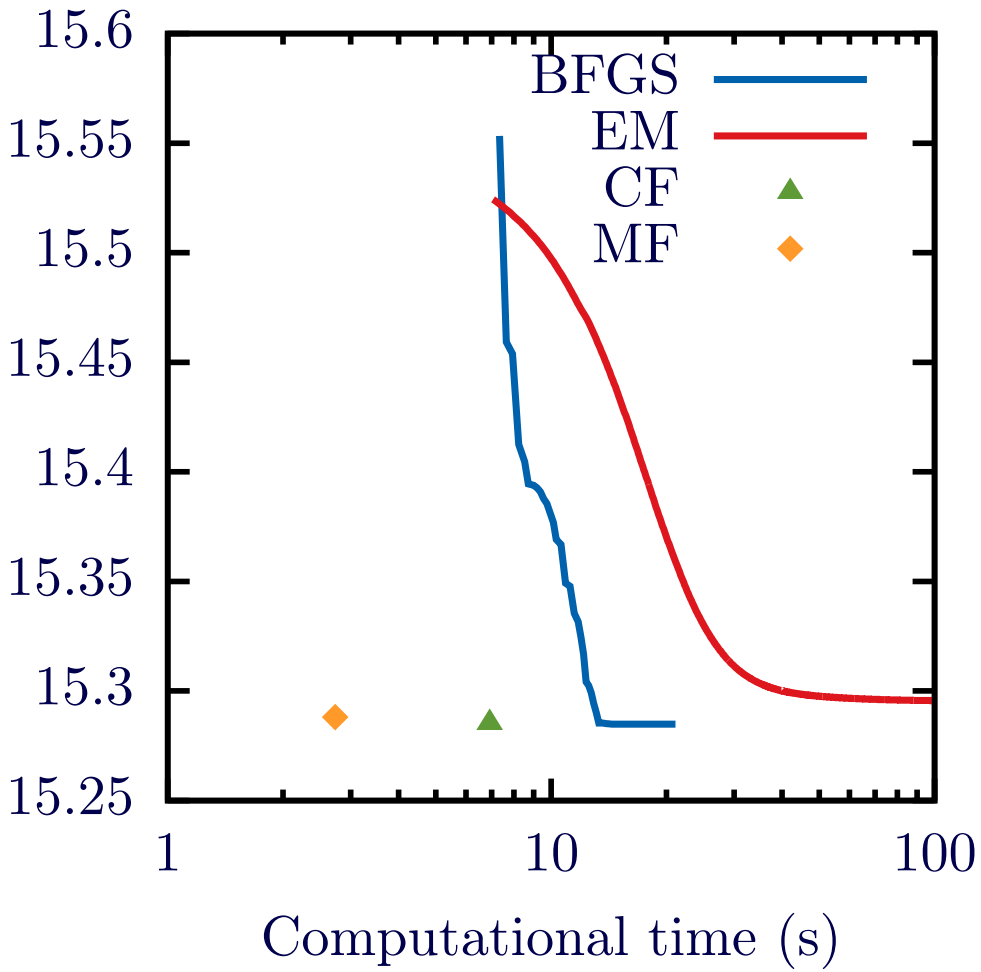
Contribution 4. Accelerating training time of MHP

No clean proof yet (only on toy example) but works very well empirically



Contribution 4. Accelerating training time of MHP

Faster by *several order of magnitude* than state-of-the-art solvers



Take-home message

- Hawkes Process for "time-oriented" machine learning
- Surprisingly relevant to fit real-word phenomena (auto-excitation, user influence)
- Very flexible: intensity can depend on features, other processes, etc.

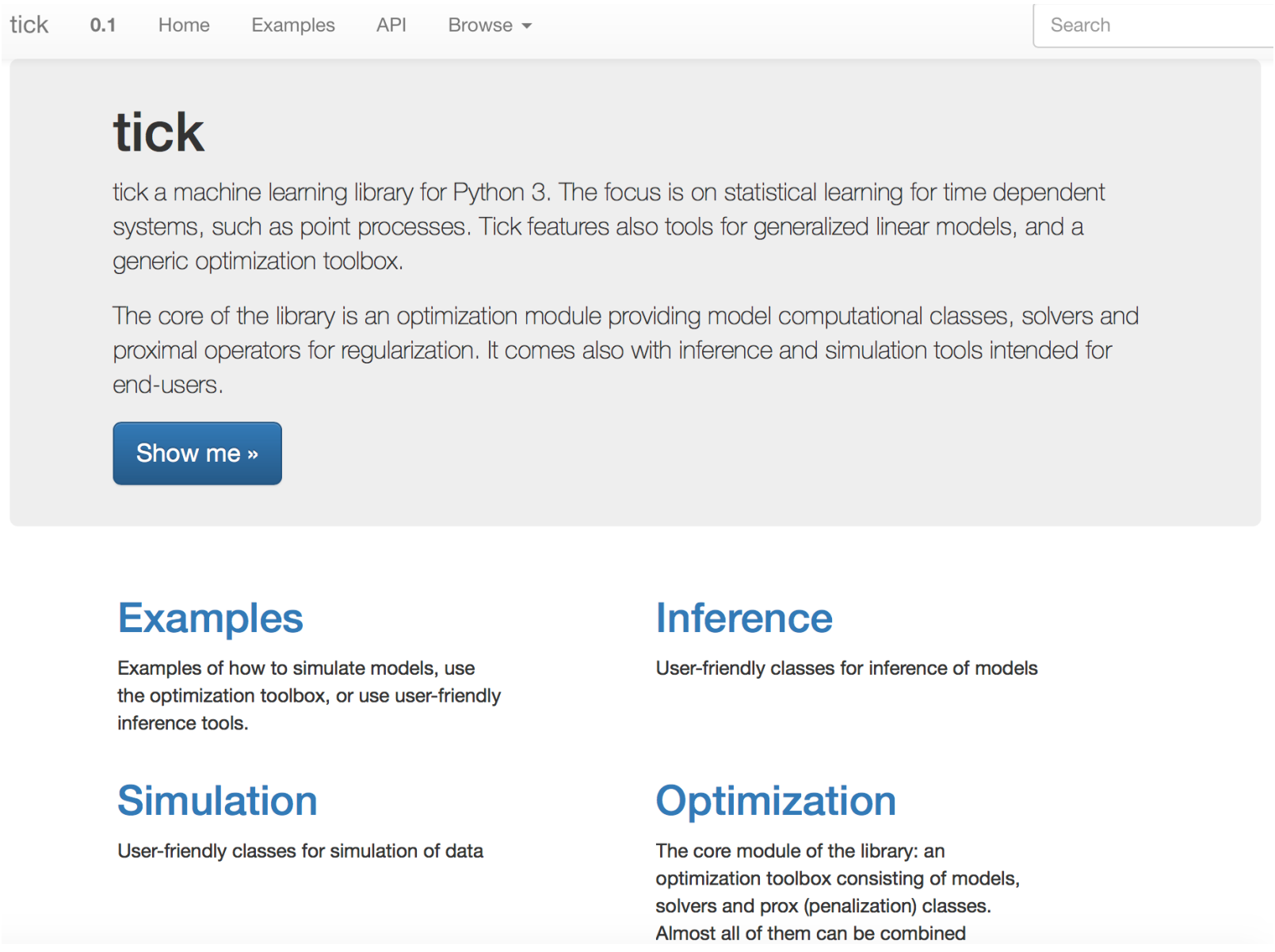
Main contributions

- Sharp **theoretical guarantees for low-rank inducing penalization** for Hawkes models
- New results about **concentration of matrix-martingales** in continuous time
- Go beyond the parametric approach: **unveil causality using integrated cumulants** matching
- **Improved training time** of the Hawkes model using a "mean-field" approximation

Software: `tick` library

- Python 3 et C++11
- Open-source (BSD-3 License)
- `pip install tick` (on MacOS and Linux...)
- `https://x-datainitiative.github.io/tick`
- Statistical learning for time-dependent models
- Point processes (Poisson, Hawkes), Survival analysis, GLMs (parallelized, sparse, etc.)
- A strong simulation and optimization toolbox
- Partnership with Intel (use-case for new processors with 256 cores)
- Contributors welcome!

Software: `tick` library



The CNAM project

What is CNAM?

- CNAM is the global social security in France
- About 60 million citizens!
- One of the world's largest electronic health-care database
- Contains all reimbursed health events: pharmacy, hospital, doctor, with quantities, diagnosis, etc.

About the project

- 3-years research partnership between Ecole polytechnique and CNAM (2015–2017), principal investigators: E. Bacry (Ecole polytechnique) and S. G.
- Renewed for 3 years by the end of 2017
- Test and develop the **potential of big data and machine learning approaches on their database**
- Pharmacovigilance, fraud detection

While this database is not made for this at all!

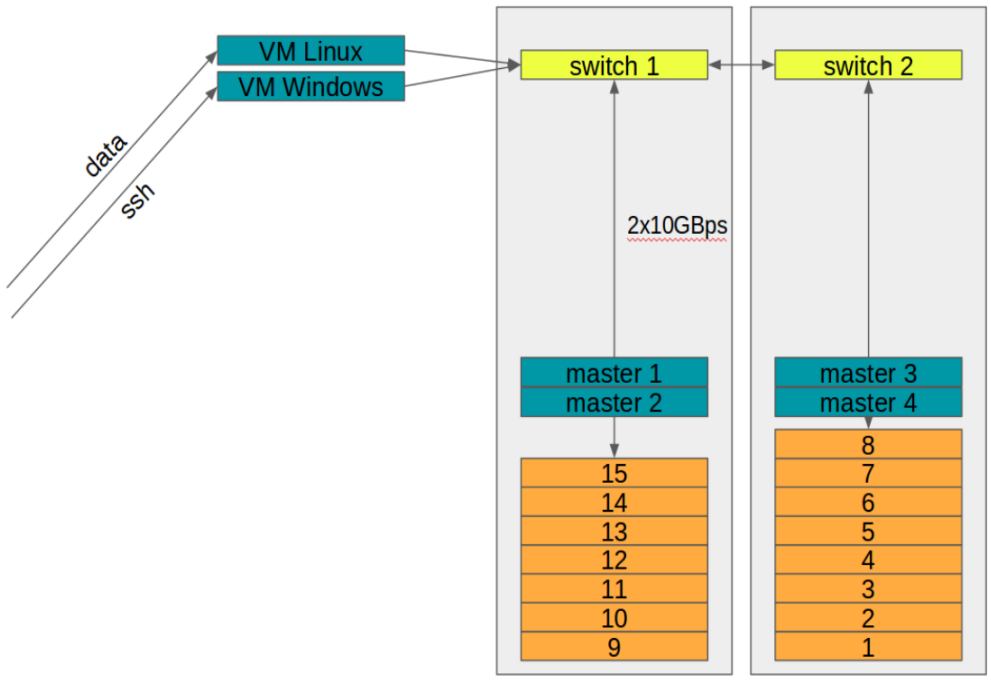
1. Motivation and examples
2. Hawkes processes
   - Introduction
   - Dimension reduction for MHP
   - Random matrix theory
   - Causality maps
   - Accelerating training time
3. Software
4. Healthcare
   - The CNAM project
   - A new infrastructure
   - A new methodology

Existing "vertical" infrastructure

- Hardware: IBM's Exadata
- Software: Oracle relational SQL (about 800 tables, about 100TB), and SAS software
- Completely closed proprietary architecture
- Not appropriate for large scale batch data processing
- **Not appropriate for methodological research**

A new “horizontal” infrastructure

- Scalable architecture: **distributed** data and processing
- 4 masters
- 15 slaves
- 240 cores
- 1.9To RAM
- 480To (120 DD)
- HDFS (triple replication)
- Spark, Scala
- Only open-source technology

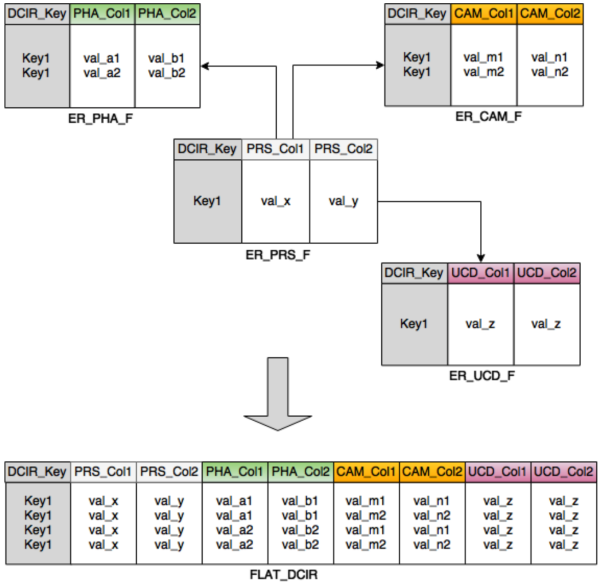


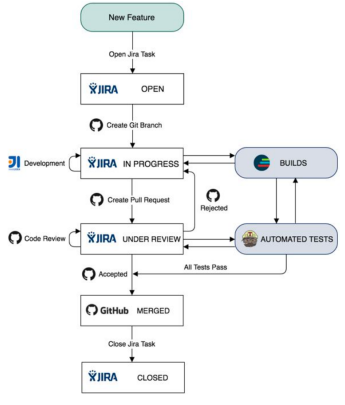
A new infrastructure

Flattening

- Understanding
- Organization
- Column-oriented format distributed files `parquet.apache.org`
- Open source
- Agile development
- ETL data processing

Crucial step : reorganize data from an ideal format for SQL (random access) to an ideal format for batch processing)





A new infrastructure

Featuring. Transformation of the data in a huge matrix ready for machine learning

- **scala** code
- Based on **spark**
- Many versions, depending on the ML algorithm

Machine Learning.

- `R` software for classical methods and benchmarks
- Home-made `tick` library for our new methodology

Example of a training dataset

Type-2 diabetic citizens with bladder cancer

Aim: a "screening" method to detect potential side-effects (user-defined adverse events)

- $\neq$ hypothesis validation (standard biostatistical methods)
- Strong simplification of cohort preparation

$\rightarrow$ Detection by screening of Pioglitazone (removed from the market in 2011)

Some figures

- 2.5 millions citizens, 4 years of data
- 1.3 To, 2 billion lines
- Flattening of the database: $\simeq$ 20 minutes (since spark 2.1)
- Featuring $\simeq$ 10 minutes

1. Motivation and examples

2. Hawkes processes
   - Introduction
   - Dimension reduction for MHP
   - Random matrix theory
   - Causality maps
   - Accelerating training time

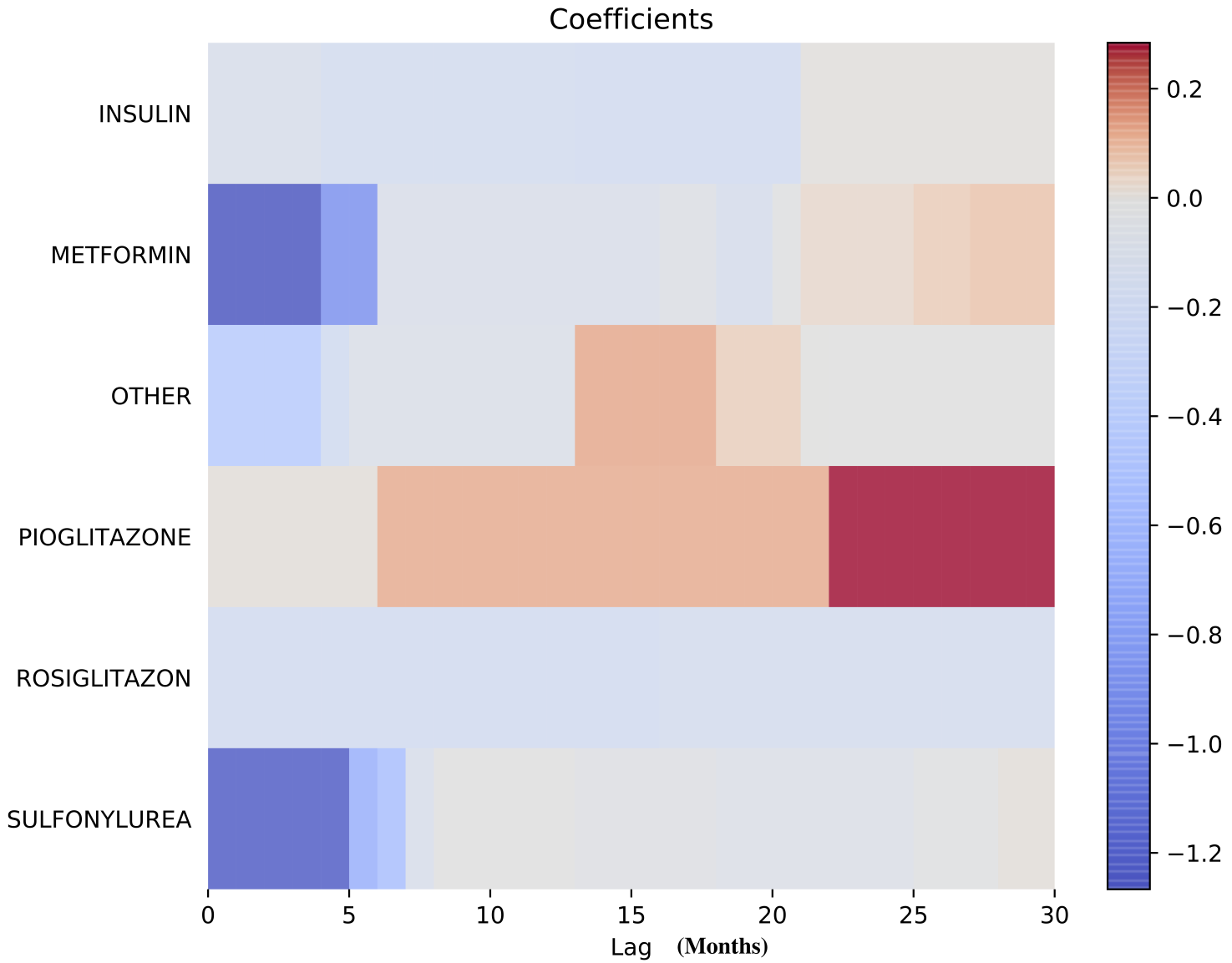
3. Software

4. Healthcare
   - The CNAM project
   - A new infrastructure
   - A new methodology

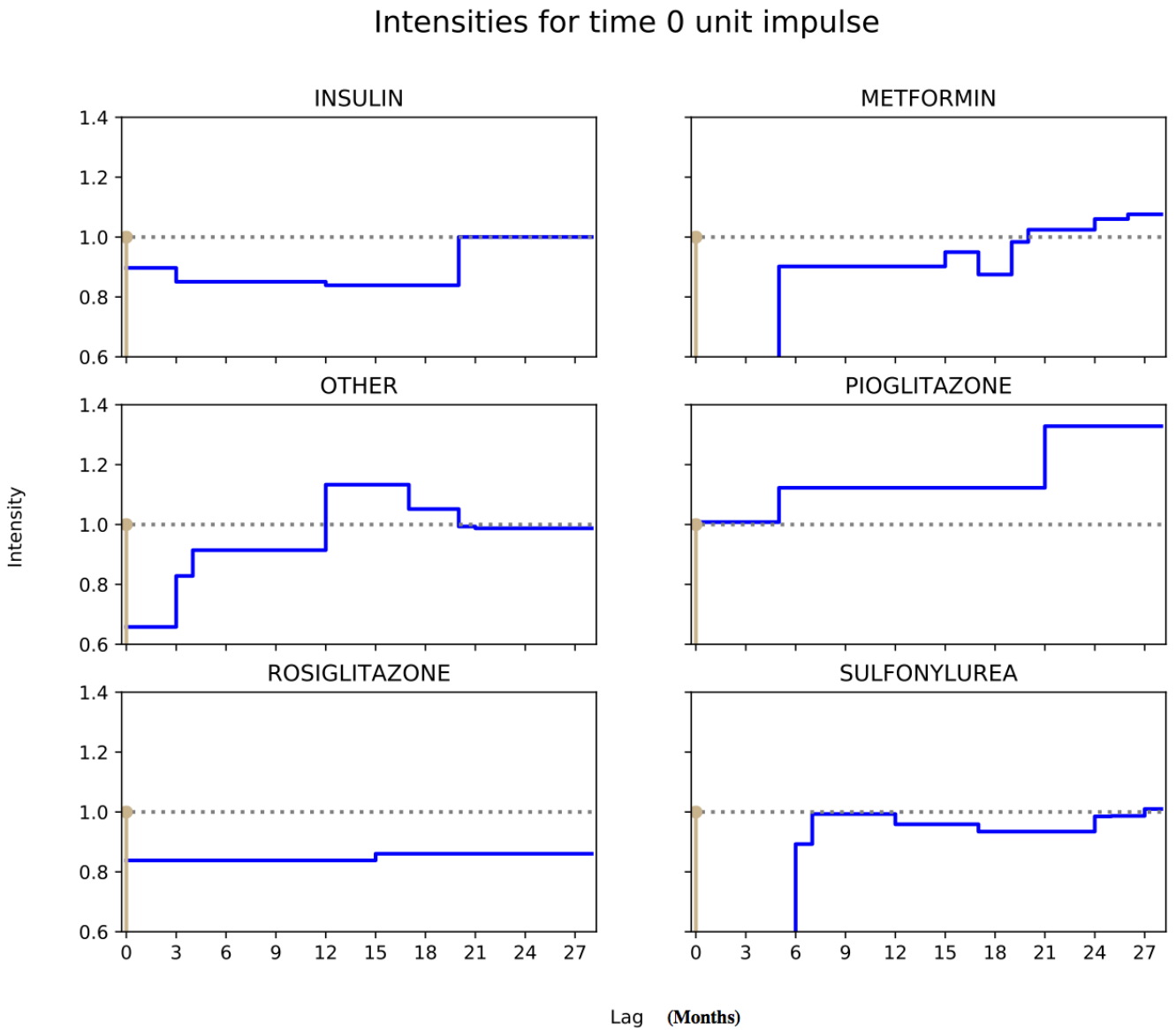
A new "self-controlled case-series" method

- Keep only citizens with the adverse effect (bladder cancer)
- Simplified data preparation (compared to what requires a Cox regression)
- Longitudinal model (features et labels)
- Estimate the impact (depending on time) of exposures to drugs on time of occurrence of the cancer
- "Weak signal" detection ability

A new methodology



A new methodology



Setting

- We have individuals $i = 1, \ldots, n$
- Time $[0, T]$ is partitioned in intervals $I_1, \ldots, I_B$ (length=month, week or day)
- We observe the number of adverse events $y_{i,b} \in \mathbb{N}$
- We put $n_i = \sum_{b=1}^{B} y_{i,b}$ = total number of adverse events of individual i
- We observe longitudinal features $x_{i,b} = (x_{i,b}^1, \ldots, x_{i,b}^d) \in \mathbb{R}^d$ over time intervals $b = 1, \ldots, B$ (drugs exposures, etc.)
- We observe "static" features $z_i = (z_i^1, \ldots, z_i^p) \in \mathbb{R}^p$ (gender, age if B is small, etc.)

Autoregressive features

The intensity of occurrence of adverse events at time b depends on feature j via:

$$\sum_{k=0}^{K-1} \theta_k^j x_{i,b}^{j,k}$$

where:

- θ_j^k = effect of feature j when exposure occurred k time intervals before the current one
- $x_{i,b}^{j,k}$ = exposure of individual i to drug j that occurred k intervals before interval b

Leads to a **translation-invariant parametrization** of the model

- **no "time-realignment"** between individuals is required
- strong improvement compared to SCCS literature, where only one type of exposure, i.e. a single molecule, can be used !

Notation

- b stands for the "current index"
- k stands for the "lag"
- $x_{i,b}^{j,k} = 0$ for any $k \geq b$

We define the $d \times B$ matrix $\boldsymbol{X}_{i,b}$ with entries

$$(\boldsymbol{X}_{i,b})_{j,k} = x_{i,b}^{j,k}$$

for $j = 1, \ldots, d$, $k = 0, \ldots, B-1$, $i = 1, \ldots, n$ and $b = 1, \ldots, B$.

We define also

$$\langle \boldsymbol{\theta}, \boldsymbol{X}_{i,b} \rangle = \sum_{j=1}^{d} \sum_{k=0}^{B-1} \theta_k^j x_{i,b}^{j,k}$$

Self-controlled case series or **conditional Poisson regression**

- Trick is to exploit the ordered statistic property of Poisson processes
- Use a model on the conditional distribution of $(y_{i,1}, \ldots, y_{i,B})|n_i$, where $n_i = \sum_{b=1}^{B} y_{i,b}$

Distribution of $(y_{i,1}, \ldots, y_{1,B})$ conditionally on n_i, x_i is

$$\text{Multinomial}\left(n_i, \frac{e^{\langle \boldsymbol{X}_{i,1}, \boldsymbol{\theta} \rangle}}{\sum_{b'=1}^{B} e^{\langle \boldsymbol{X}_{i,b'}, \boldsymbol{\theta} \rangle}}, \ldots, \frac{e^{\langle \boldsymbol{X}_{i,B}, \boldsymbol{\theta} \rangle}}{\sum_{b'=1}^{B} e^{\langle \boldsymbol{X}_{i,b'}, \boldsymbol{\theta} \rangle}}\right)$$

Namely, we have that

$$\mathbb{P}(y_{i,1},\dots,y_{1,B}|n_i,x_i) = \frac{n_i!}{\prod_{b=1}^{B} y_{ib}!}\prod_{b=1}^{B}\left(\frac{e^{\langle \boldsymbol{X}_{i,b},\boldsymbol{\theta}\rangle}}{\sum_{b'=1}^{B} e^{\langle \boldsymbol{X}_{i,b'},\boldsymbol{\theta}\rangle}}\right)^{y_{i,b}}$$

Important remark
Constant effects (independent on b, such as the z_i) are killed by the conditioning with respect to n_i, since whenever

$$\lambda_{i,b} = e^{\langle \boldsymbol{X}_{i,b'},\boldsymbol{\theta}\rangle+\beta^\top z+c_b},$$

we have

$$\frac{\lambda_{i,b}}{\sum_{b'=1}^{B}\lambda_{i,b'}} = \frac{e^{\langle \boldsymbol{X}_{i,b},\boldsymbol{\theta}\rangle}}{\sum_{b'=1}^{B} e^{\langle \boldsymbol{X}_{i,b'},\boldsymbol{\theta}\rangle}}$$

Penalization

- We want to consider a large number of lags K, but we want to "smooth" time-adjacent coefficients $\theta^j_1, \ldots, \theta^j_B$
- We use "group" total-variation penalization

Algorithm

We minimize the following over $\boldsymbol{\theta}$

$$-\frac{1}{n}\sum_{i=1}^{n}\sum_{b=1}^{B}\delta_i y_{i,b}\left(\langle \boldsymbol{X}_{i,b}, \boldsymbol{\theta}\rangle - \log\left(\sum_{b'=1}^{B} e^{\langle \boldsymbol{X}_{i,b'}, \boldsymbol{\theta}\rangle}\right)\right) + \lambda \sum_{i=1}^{d}\sum_{k=1}^{B-1}|\theta^j_k - \theta^j_{k-1}|$$

Tips and tricks

- Stratified V-fold cross-validation for λ
- Very fast solver: SGD with variance reduction
- Fast proximal operator for total-variation penalization
- Exploit sparsity of the matrix $\boldsymbol{X}_{i,b}$

Available generalizations

- Right-censoring
- Other types of featuring in $\boldsymbol{X}_{i,b}$
- Features product (joint exposures)
- Confidence intervals

Next generalizations

- Multi-task (many adverse events at the same time)

Thank you!