

hawkesbow

Felix Cheyssou

This package implements a spectral approach to the parametric estimation of Hawkes processes from Binned Observations through Whittle likelihood (HawkesBOW). It is based on the results of the article (Cheyssou and Lang 2020).

Contents

1	Installation	2
2	Details of the package	2
2.1	The Hawkes process	2
2.2	Estimation procedure	2
3	Main usage	4
3.1	Simulation of Hawkes processes	4
3.2	Estimation of Hawkes processes	4
3.3	Other functions	5
4	Reproduction kernels	5
4.1	The exponential kernel	5
4.2	The symmetric exponential kernel	6
4.3	The gaussian kernel	6
4.4	The power law kernel	6
4.5	The Pareto kernels	7
5	To be implemented	7
	References	8

1 Installation

You can install the released version of hawkesbow from CRAN with:

```
install.packages("hawkesbow")
```

Alternatively, you can install the up-to-date version of hawkesbow from GitHub with:

```
devtools::install_github("fcheysson/hawkesbow")
```

2 Details of the package

2.1 The Hawkes process

Hawkes processes form a family of models for point processes for which the occurrence of an event temporarily increases the probability of future events occurring (Alan G. Hawkes 1971). Formally, a Hawkes process N on \mathbb{R} can be defined from its conditional intensity $\lambda(\cdot)$:

$$\begin{aligned}\lambda(t) &= \eta + \int_{-\infty}^t h(t-u)N(du) \\ &= \eta + \sum_{T_i < t} h(t-T_i),\end{aligned}$$

where the variables T_i denote the arrival times of the process, the *immigration intensity* η is a positive constant, and the *reproduction function* $h : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is a measurable function. The reproduction function can be further decomposed as $h = \mu h^*$, where $\mu = \int_{\mathbb{R}} h(t)dt < 1$ is called the *reproduction mean* and h^* is a true density function, $\int_{\mathbb{R}} h^*(t)dt = 1$, called the *reproduction kernel*.

Alternatively, the Hawkes process can be constructed as a poissonian cluster process (Alan G. Hawkes and Oakes 1974). The process consists of a flow of *immigrants*, the cluster centres, arriving according to a homogeneous Poisson process of intensity η . Then, an immigrant arriving at time T_i generates *children* according to an inhomogeneous Poisson process of intensity $h(\cdot - T_i)$. These in turn independently generate children according to the same process, and so on *ad infinitum*. The processes consisting of an immigrant and all its descendants are therefore branching processes, and are independent of each other. Finally, the process resulting from the superposition of all these branching processes is a Hawkes process of conditional intensity $\lambda(\cdot)$ (see Figure 1).

This package also supports non-causal Hawkes processes, for which the reproduction kernel h^* may take non-negative values on $\mathbb{R}_{\leq 0}$. Such processes are not defined through their conditional intensity functions, but through the poisson cluster representation, where each individual can potentially generate offsprings both in the future and in the past.

The count sequence of a Hawkes process is the time series generated by the event counts of the process, that is the series obtained by counting the events of the process on intervals of fixed length. Formally, the count sequence with bin size Δ associated to the point process N is the sequence $(X_k)_{k \in \mathbb{Z}}$.

2.2 Estimation procedure

This package fully supports the parametric estimation of Hawkes processes from their count sequences via minimisation of the Whittle likelihood, and partially supports the estimation from the arrival times via maximisation of the likelihood.

Maximum likelihood estimation The parameters of a Hawkes process with arrival times $(T_i)_{1 \leq i \leq n}$ on the interval $[0, T]$ can be estimated by maximising the following pseudo-likelihood

$$\mathcal{L}_n(\theta) = \left(\prod_{i=1}^n \lambda(T_i) \right) \exp \left(- \int_0^T \lambda(s) ds \right).$$

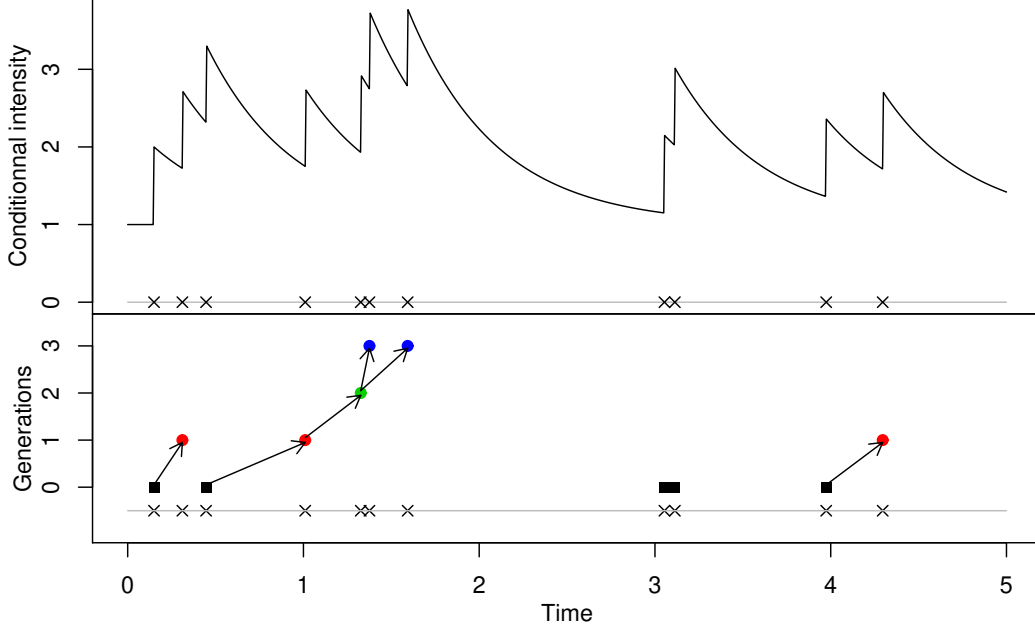


Figure 1: Realisation of an exponential Hawkes process, with $\eta = 1$, $\mu = 0.5$ and $h^*(t) = 2e^{-2t}$. The crosses represent the arrival times of the process. (Top) Representation of the conditional intensity of the process: each event increases the probability of occurrence of future events, according to the reproduction function $h = \mu h^*$. (Bottom) Representation of the process as a Poisson cluster process: each immigrant (black squares, of generation 0) can generate children (red dots, of generation 1), which can in turn generate children, and so on.

Note that each step of the optimisation is usually of complexity $\mathcal{O}(n^2)$.

Minimisation of the Whittle likelihood Alternatively, the parameters of a Hawkes process with count sequence $(X_k)_{1 \leq k \leq N}$ can be estimated by minimising the log-spectral (Whittle) likelihood

$$\mathcal{L}'_N(\theta) = \frac{1}{4\pi} \int_{-\pi}^{\pi} \left(\log f_{\theta}(\omega) + \frac{I_N(\omega)}{f_{\theta}(\omega)} \right) d\omega,$$

where $I_N(\omega) = (2\pi N)^{-1} \left| \sum_{k=1}^N X_k \exp(-ik\omega) \right|^2$ denotes the periodogram of (X_k) and $f_{\theta}(\cdot)$ its spectral density function. Note that each step of the optimisation is of complexity $\mathcal{O}(n)$ and the periodogram can be calculated in complexity $\mathcal{O}(n \log n)$ using a Fast Fourier Transform algorithm, making this method usually faster than maximum likelihood estimation.

Spectral density function of the count sequence Note that the spectral density function f_{θ} of (X_k) can be related to the spectral density f'_{θ} of the time-continuous count sequence $(X_t)_{t \in \mathbb{R}}$ by taking into account spectral aliasing:

$$f_{\theta}(\omega) = \sum_{k \in \mathbb{Z}} f'_{\theta}(\omega + 2k\pi).$$

In turn, f'_{θ} can be derived from the Bartlett spectrum of the Hawkes process (Daley and Vere-Jones 2003, example 8.2(e)) and is given by

$$f'_{\theta}(\omega) = \frac{4\Delta}{\omega^2} \frac{\eta}{1-\mu} \sin^2\left(\frac{\omega}{2}\right) \left| 1 - \tilde{h}\left(\frac{\omega}{\Delta}\right) \right|^{-2},$$

where \tilde{h} denotes the Fourier transform of h :

$$\tilde{h}(\omega) = \int_{\mathbb{R}} h(t) \exp(-i\omega t) dt.$$

3 Main usage

This package supports both Hawkes processes and their count sequences.

3.1 Simulation of Hawkes processes

Hawkes processes can be simulated by the function `hawkes`:

```
hawkes(end, fun, repr, family, M=null, ...)
```

where `end` denotes the end of the interval $[0, T]$. This function uses the cluster representation:

- First, the immigrants are drawn according to a (potentially inhomogeneous) Poisson process with intensity measure `fun`.
- Second, the number of offsprings of an immigrant are drawn from a Poisson distribution with intensity `repr`.
- Third, these offsprings are distributed according to the `family` distribution.
- Then, further offsprings are generated according to the last two steps.

The argument `fun` can take a numeric value or be specified as a function, in which case the argument `M` must be specified as an upper bound on `fun` (to allow simulation of the immigration process by thinning). The argument `family` can either be specified as a string `name` corresponding to a distribution with random generation function `rname` (for example `exp` for `rexp`), or directly as a random generation function. The optional arguments `...` are passed to the random generation function specified by `family`.

This returns a list of class `hawkes`, whose realisations are stored in the member `p`. Other members of the list are mainly used for support functions.

Examples

- Simulate a Hawkes process with immigration intensity $\eta = 1$, reproduction mean $\mu = 0.5$ and reproduction kernel $h^*(t) = 2e^{-2t}1_{\{t \geq 0\}}$ on $[0, 10]$:

```
x <- hawkes(10, fun = 1, repr = 0.5, family = "exp", rate = 2)
```

- Simulate a Hawkes process with inhomogeneous immigration process with intensity $\eta(t) = 1 + \sin(t)$, reproduction mean $\mu = 0.25$ and $[0, 1]$ -triangular reproduction kernel $h^*(t) = (1 - t)1_{\{0 \leq t \leq 1\}}$:

```
x <- hawkes(10, fun=function(y) {1+sin(y)}, M=2, repr=0.25,
           family=function(n) {1 - sqrt(1 - runif(n))})
```

Plot function

- Hawkes processes can be plotted with the function `plot.hawkes`:

```
plot.hawkes(x, intensity = FALSE, precision = 1e3, fun = NULL, repr = NULL, family = NULL,
           M = NULL, ...)
```

If `x` is of class `hawkes`, as for objects returned by the function `hawkes`, arguments `fun` through `M` can be ignored. If `intensity` is set to `FALSE`, this plots the genealogy of the simulated Hawkes process (as in Figure 1, bottom). If it is set to `TRUE`, this plots the conditional intensity of the process (as in Figure 1, top).

3.2 Estimation of Hawkes processes

Two functions implement the estimation of Hawkes processes: `mle` from arrival times (T_i) and `whittle` from count sequences (X_k) . While the optimisation procedure rely on existing functions (see below), calculations of both the usual and Whittle likelihood functions are done in C++ via Rcpp (Eddelbuettel and François 2011) and RcppArmadillo (Eddelbuettel and Sanderson 2014).

By maximum likelihood The maximum likelihood method is implemented by the function

```
mle(events, kern, end, init = NULL, opts = NULL, ...)
```

`events` holds the arrival times (T_i) in ascending order, `kern` must be a string (partially) matching one of the reproduction kernels (see below) and `end` denotes the endpoint T of observation of the process.

The optimisation of the maximum likelihood function is done by the function `nloptr` from the package `nloptr` (Johnson, n.d.) with algorithm L-BFGS-B, where the derivatives of the likelihood are calculated explicitly. By default, parameters are constrained to be positive, and additionally μ is constrained to be below 1. However, both the arguments `opts` and `...` are passed on to `nloptr`, so the algorithm, the constraints, or any other parameter of the optimisation procedure can be changed.

Example of use:

```
x = hawkes(100, fun = 1, repr = .5, family = "exp", rate = 1)
mle(x$p, "Exponential", x$end)
```

By minimisation of the Whittle likelihood The Whittle likelihood method is implemented by the function

```
whittle(counts, kern, binsize = NULL, trunc = 5L, init = NULL, ...)
```

`counts` holds the count sequence (X_k), `kern` must be a string (partially) matching one of the reproduction kernels (see below), `binsize` denotes the bin size Δ and `trunc` is the number of foldings due to aliasing taken into account.

The optimisation of the Whittle likelihood function is done by the function `optim`, with algorithm L-BFGS-B where the derivatives of the likelihood are approximated by finite differences. By default, parameters are constrained to be positive, and additionally μ is constrained to be below 1. However, the argument `...` is passed to `optim`, so any optimisation parameter can be changed.

Example of use:

```
x = hawkes(1000, fun = 1, repr = .5, family = "exp", rate = 1)
y = discrete(x, binsize = 1)
whittle(y, "Exponential", 1)
```

Note that `discrete` is a useful function to create the count sequence (X_k) associated with an object `x` of class `hawkes`.

3.3 Other functions

Other functions are all well documented and with examples. This subsection will be expanded in the future.

4 Reproduction kernels

We introduce the reproduction kernels that are currently implemented in this package. Recall that the Fourier transform of a reproduction kernel is given by

$$\widetilde{h^*}(\omega) = \int_{\mathbb{R}} \exp(-i\omega t) h(t) dt,$$

and that it is a Hermitian function $\widetilde{h^*}(-\omega) = \overline{\widetilde{h^*}(\omega)}$.

4.1 The exponential kernel

This is the exponential density function with parameter $\beta > 0$:

$$h^*(t) = \beta e^{-\beta t} \mathbf{1}_{\{t>0\}}.$$

Its Fourier transform is

$$\widetilde{h}^*(\omega) = \beta \frac{1}{\beta + i\omega} = \beta \frac{\beta - i\omega}{\beta^2 + \omega^2}.$$

The exponential kernel can be specified with the string `Exponential` and the parameter β with the usual argument `rate`. Both maximum and Whittle likelihood methods are fully implemented for exponential kernels. Moreover, the likelihood function is implemented in complexity $\mathcal{O}(n)$, using the relations in (Ozaki and Ogata 1979).

4.2 The symmetric exponential kernel

This is a symmetrised version of the exponential density function with parameter $\beta > 0$:

$$h^*(t) = \frac{1}{2} \beta e^{-\beta|t|}.$$

Its Fourier transform is

$$\widetilde{h}^*(\omega) = \frac{\beta^2}{\beta^2 + \omega^2}.$$

The symmetric exponential kernel can be specified with the string `SymmetricExponential` and the parameter β with the argument `rate`. Only the Whittle likelihood method is implemented for symmetric exponential kernels. Note that it is a non-causal kernel, as $h^*(t) \neq 0$ for $t < 0$.

4.3 The gaussian kernel

This is the gaussian density function with mean $\nu \in \mathbb{R}$ and variance $\sigma^2 > 0$:

$$h^*(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(t-\nu)^2}{2\sigma^2}\right).$$

Its Fourier transform is

$$\widetilde{h}^*(\omega) = \exp\left(-\frac{\sigma^2\omega^2}{2} - i\nu\omega\right).$$

The gaussian kernel can be specified with the string `Gaussian` and its parameters ν and σ with the usual arguments `mean` and `sd` respectively. Only the Whittle likelihood method is implemented for gaussian kernels. Note that it is a non-causal kernel, as $h^*(t) \neq 0$ for $t < 0$.

4.4 The power law kernel

This is a normalised and shifted power law function, with shape $\theta > 0$ and scale $a > 0$:

$$h^*(t) = \theta a^\theta (t+a)^{-\theta-1} \mathbf{1}_{\{t>0\}}.$$

For positive ω , its Fourier transform is given by

$$\widetilde{h}_\theta^*(\omega) = \theta \exp(i\omega a) E_{\theta+1}(i\omega a),$$

where $E_\theta(ix)$ denotes the integral

$$E_\theta(ix) = \int_1^\infty t^{-\theta} \exp(-ixt) dt.$$

With successive integration by parts, this integral can be related to $E_{\theta'}(ix)$, with $0 < \theta' \leq 1$.

If $\theta' = 1$ or equivalently $\theta \in \mathbb{N}_{\neq 0}$, the integral $E_1(ix)$, called the exponential integral with imaginary argument, can be related the trigonometric integrals and calculated using Padé approximants (Rowe et al. 2015, Appendix B), accurate to better than 10^{-16} . The function `E1_imaginary` implements this approximation.

If $\theta' < 1$ or equivalently $\theta \in \mathbb{R} \setminus \mathbb{N}$, the integral $E_\theta(ix)$ can be related to the incomplete gamma function with imaginary argument

$$\Gamma_i(x, \alpha) = \int_x^\infty t^{\alpha-1} e^{-it} dt,$$

where $0 < \alpha < 1$. We implemented Taylor approximations of this integral, accurate to better than 10^{-7} , in the function `inc_gamma_imag`.

The power law kernel can be specified with the string `PowerLaw` and its parameters θ and a with the arguments `shape` and `scale` respectively. Both maximum and Whittle likelihood methods are implemented for power law kernels.

4.5 The Pareto kernels

This is the Pareto density function with shape $\theta > 0$ and scale $a > 0$:

$$h_\theta^*(t) = \theta a^\theta t^{-\theta-1} 1_{\{t>a\}}.$$

For positive ω , its Fourier transform is given by

$$\widetilde{h}_\theta^*(\omega) = \theta E_{\theta+1}(i\omega a),$$

see above for the definition of $E_\theta(ix)$.

Only Pareto kernels with fixed $\theta = 1, 2,$ and 3 have been implemented and can be specified with the strings `Pareto1`, `Pareto2` and `Pareto3` respectively, with parameter a specified with the argument `scale`. Only the Whittle method is available for Pareto kernels.

5 To be implemented

- Improve this vignette: it is currently too sparse and functions of the package could need some better description.
- Add some real datasets to the package: real life case-studies with good datasets help understand the functionalities of a package.
- Variance and confidence intervals for the estimation with function `whittle`: note that currently, the variance-covariance matrix returned by the optimisation method in function `whittle` is not accurate, as it ignores the dependence within the count sequence (X_k) .
- Diagnostics for the estimated model: spectral density based goodness-of-fit tests are to be implemented for the estimated Hawkes processes, based on the work of (Paparoditis 2000).
- Custom built-kernels: allow the user to input reproduction kernels that are not already implemented.

References

- Cheysson, Felix, and Gabriel Lang. 2020. “Strong mixing condition for Hawkes processes and application to Whittle estimation from count data.” *arXiv*, March. <https://arxiv.org/abs/2003.04314>.
- Daley, D. J., and David Vere-Jones. 2003. *An Introduction to the Theory of Point Processes*. Probability and Its Applications. New York: Springer-Verlag. <https://doi.org/10.1007/b97277>.
- Eddelbuettel, Dirk, and Romain François. 2011. “Rcpp : Seamless R and C++ Integration.” *J. Stat. Softw.* 40 (8): 1–18. <https://doi.org/10.18637/jss.v040.i08>.
- Eddelbuettel, Dirk, and Conrad Sanderson. 2014. “RcppArmadillo: Accelerating r with High-Performance c++ Linear Algebra.” *Computational Statistics and Data Analysis* 71: 1054–63. <http://dx.doi.org/10.1016/j.csda.2013.02.005>.
- Hawkes, Alan G. 1971. “Spectra of Some Self-Exciting and Mutually Exciting Point Processes.” *Biometrika* 58 (1): 83–90. <https://doi.org/10.2307/2334319>.
- Hawkes, Alan G., and David Oakes. 1974. “A cluster process representation of a self-exciting process.” *J. Appl. Probab.* 11 (03): 493–503. <https://doi.org/10.2307/3212693>.
- Johnson, Steven G. n.d. “The NLOpt nonlinear-optimization package.”
- Ozaki, T., and Yosihiko Ogata. 1979. “Maximum likelihood estimation of Hawkes’ self-exciting point processes.” *Ann. Inst. Stat. Math.* 31 (1): 145–55. <https://doi.org/10.1007/BF02480272>.
- Paparoditis, Efsthios. 2000. “Spectral density based goodness-of-fit tests for time series models.” *Scand. J. Stat.* 27 (1): 143–76. <https://doi.org/10.1111/1467-9469.00184>.
- Rowe, B. T. P., M. Jarvis, R. Mandelbaum, G. M. Bernstein, J. Bosch, M. Simet, J. E. Meyers, et al. 2015. “GalSim: The modular galaxy image simulation toolkit.” *Astron. Comput.* 10: 121–50. <https://doi.org/10.1016/j.ascom.2015.02.002>.