

Package ‘hawkes’

July 22, 2025

Version 0.0-4

Date 2014-01-08

Title Hawkes process simulation and calibration toolkit

Author Riadh Zaatour <zaatour_riadh@yahoo.fr>

Maintainer Riadh Zaatour <zaatour_riadh@yahoo.fr>

Description The package allows to simulate Hawkes process both in univariate and multivariate settings. It gives functions to compute different moments of the number of jumps of the process on a given interval, such as mean, variance or autocorrelation of process jumps on time intervals separated by a lag.

License GPL (>= 2)

Depends R (>= 3.0.2)

Imports Rcpp (>= 0.11.1)

LinkingTo Rcpp, RcppArmadillo (>= 0.4.100.2.1)

Repository CRAN

Repository/R-Forge/Project hawkes

Repository/R-Forge/Revision 61

Repository/R-Forge/DateTimeStamp 2014-05-19 11:27:02

Date/Publication 2014-05-19 16:33:15

NeedsCompilation yes

Contents

jumpAutocorrelation	2
jumpMean	3
jumpVariance	4
likelihoodHawkes	5
simulateHawkes	6
Index	8

jumpAutocorrelation	<i>Autocorrelation of Hawkes process jumps on nonoverlapping time intervals with lag.</i>
---------------------	---

Description

The function returns the theoretical autocorrelation of the number of jumps of a Hawkes process on nonoverlapping time intervals with lag

Usage

```
jumpAutocorrelation(lambda0, alpha, beta, tau, lag)
```

Arguments

lambda0	Vector of initial intensity, a scalar in the monovariate case.
alpha	Matrix of excitation, a scalar in the monovariate case. Excitation values are all positive.
beta	Vector of betas, a scalar in the monovariate case.
tau	Time interval length.
lag	Time lag.

Details

Notice that in the scalar case, one must have $\beta > \alpha$ for the process to be stable, and in the multivariate case, the matrix $(\text{diag}(\beta) - \alpha)$ must have eigen values with strictly positive real parts for the process to be stable.

Value

Returns a matrix containing the autocorrelation of the number of jumps of process components.

References

Jose Da Fonseca and Riadh Zaatour Hawkes Process : Fast Calibration, Application to Trade Clustering and Diffusive Limit. *Journal of Futures Markets*, Volume 34, Issue 6, pages 497-606, June 2014.

Jose Da Fonseca and Riadh Zaatour Clustering and Mean Reversion in Hawkes Microstructure Models.

Examples

```
#One dimensional Hawkes process
lambda0<-0.02
alpha<-0.05
beta<-0.06
tau<-60#one minute
lag<-0#adjacent non overlapping intervals
h<-jumpAutocorrelation(lambda0,alpha,beta,tau,lag)

#Multivariate Hawkes process
lambda0<-c(0.02,0.02)
alpha<-matrix(c(0.05,0,0,0.05),byrow=TRUE,nrow=2)
beta<-c(0.06,0.06)
tau<-60#one minute
lag<-0#adjacent non overlapping intervals
h<-jumpAutocorrelation(lambda0,alpha,beta,tau,lag)
```

jumpMean	<i>Mean of Hawkes process jumps.</i>
----------	--------------------------------------

Description

The function returns the theoretical mean of the number of jumps of a Hawkes process on a time interval of length tau.

Usage

```
jumpMean(lambda0, alpha, beta, tau)
```

Arguments

lambda0	Vector of initial intensity, a scalar in the monovariate case.
alpha	Matrix of excitation, a scalar in the monovariate case. Excitation values are all positive.
beta	Vector of betas, a scalar in the monovariate case.
tau	Time interval length.

Details

Notice that in the scalar case, one must have $\beta > \alpha$ for the process to be stable, and in the multivariate case, the matrix $(\text{diag}(\beta) - \alpha)$ must have eigen values with strictly positive real parts for the process to be stable.

Value

Returns a vector containing the mean number of jumps of every process component.

References

Jose Da Fonseca and Riadh Zaatour Hawkes Process : Fast Calibration, Application to Trade Clustering and Diffusive Limit. *Journal of Futures Markets*, Volume 34, Issue 6, pages 497-606, June 2014.

Jose Da Fonseca and Riadh Zaatour Clustering and Mean Reversion in Hawkes Microstructure Models.

Examples

```
#One dimensional Hawkes process
lambda0<-0.02
alpha<-0.05
beta<-0.06
tau<-60#one minute
h<-jumpMean(lambda0,alpha,beta,tau)

#Multivariate Hawkes process
lambda0<-c(0.02,0.02)
alpha<-matrix(c(0.05,0,0,0.05),byrow=TRUE,nrow=2)
beta<-c(0.06,0.06)
tau<-60#one minute
h<-jumpMean(lambda0,alpha,beta,tau)
```

jumpVariance	<i>Variance of Hawkes process jumps.</i>
--------------	--

Description

The function returns the theoretical variance matrix of the number of jumps of a Hawkes process on a time interval of length tau.

Usage

```
jumpVariance(lambda0, alpha, beta, tau)
```

Arguments

lambda0	Vector of initial intensity, a scalar in the monovariate case.
alpha	Matrix of excitation, a scalar in the monovariate case. Excitation values are all positive.
beta	Vector of betas, a scalar in the monovariate case.
tau	Time interval length.

Details

Notice that in the scalar case, one must have $\beta > \alpha$ for the process to be stable, and in the multivariate case, the matrix $(\text{diag}(\beta) - \alpha)$ must have eigen values with strictly positive real parts for the process to be stable.

Value

Returns a matrix containing the variance of the number of jumps of every process component.

References

Jose Da Fonseca and Riadh Zaatour Hawkes Process : Fast Calibration, Application to Trade Clustering and Diffusive Limit. *Journal of Futures Markets*, Volume 34, Issue 6, pages 497-606, June 2014.

Jose Da Fonseca and Riadh Zaatour Clustering and Mean Reversion in Hawkes Microstructure Models.

Examples

```
#One dimensional Hawkes process
lambda0<-0.02
alpha<-0.05
beta<-0.06
tau<-60#one minute
h<-jumpVariance(lambda0,alpha,beta,tau)

#Multivariate Hawkes process
lambda0<-c(0.02,0.02)
alpha<-matrix(c(0.05,0,0,0.05),byrow=TRUE,nrow=2)
beta<-c(0.06,0.06)
tau<-60#one minute
h<-jumpVariance(lambda0,alpha,beta,tau)
```

likelihoodHawkes

Compute the likelihood function of a hawkes process

Description

Compute the likelihood function of a hawkes process for the given parameter and given the jump times vector (or list of vectors in the multivariate case), and until a time horizon.

Usage

```
likelihoodHawkes(lambda0, alpha, beta, history)
```

Arguments

lambda0	Vector of initial intensity, a scalar in the monovariate case.
alpha	Matrix of excitation, a scalar in the monovariate case. Excitation values are all positive.
beta	Vector of betas, a scalar in the monovariate case.
history	Jump times vector (or list of vectors in the multivariate case).

Value

Returns the opposite of the likelihood.

References

Y. Ogata. (1981) On Lewis simulation method for point processes. *IEEE Transactions on Information Theory*, **31**

Examples

```
#One dimensional Hawkes process
lambda0<-0.2
alpha<-0.5
beta<-0.7
history<-simulateHawkes(lambda0,alpha,beta,3600)
l<-likelihoodHawkes(lambda0,alpha,beta,history[[1]])

#Multivariate Hawkes process
lambda0<-c(0.2,0.2)
alpha<-matrix(c(0.5,0,0,0.5),byrow=TRUE,nrow=2)
beta<-c(0.7,0.7)
history<-simulateHawkes(lambda0,alpha,beta,3600)
l<-likelihoodHawkes(lambda0,alpha,beta,history)
```

simulateHawkes	<i>Hawkes process simulation Function</i>
----------------	---

Description

The function simulates a Hawkes process for the given parameter, and until a time horizon.

Usage

```
simulateHawkes(lambda0, alpha, beta, horizon)
```

Arguments

lambda0	Vector of initial intensity, a scalar in the monovariate case.
alpha	Matrix of excitation, a scalar in the monovariate case. Excitation values are all positive.
beta	Vector of betas, a scalar in the monovariate case.
horizon	Time horizon until which the simulation is to be conducted.

Details

Notice that in the scalar case, one must have $\beta > \alpha$ for the process to be stable, and in the multivariate case, the matrix $(\text{diag}(\beta) - \alpha)$ must have eigen values with strictly positive real parts for the process to be stable.

Value

Returns a vector of jump times in the monovariate case, and a list of such vectors for every component in the multivariate case.

References

Y. Ogata. (1981) On Lewis simulation method for point processes. *IEEE Transactions on Information Theory*, **31**

Examples

```
#One dimensional Hawkes process
lambda0<-0.2
alpha<-0.5
beta<-0.7
horizon<-3600#one hour
h<-simulateHawkes(lambda0,alpha,beta,horizon)

#Multivariate Hawkes process
lambda0<-c(0.2,0.2)
alpha<-matrix(c(0.5,0,0,0.5),byrow=TRUE,nrow=2)
beta<-c(0.7,0.7)
horizon<-3600#one hour
h<-simulateHawkes(lambda0,alpha,beta,horizon)
```

Index

jumpAutocorrelation, [2](#)

jumpMean, [3](#)

jumpVariance, [4](#)

likelihoodHawkes, [5](#)

simulateHawkes, [6](#)