Package ‘etasFLP’

September 14, 2023

Type Package

Title Mixed FLP and ML Estimation of ETAS Space-Time Point Processes for Earthquake Description

Version 2.2.2

Date 2023-09-14

Description Estimation of the components of an ETAS (Epidemic Type Aftershock Sequence) model for earthquake description. Non-parametric background seismicity can be estimated through FLP (Forward Likelihood Predictive). New version 2.0.0: covariates have been introduced to explain the effects of external factors on the induced seismicity; the parametrization has been changed; Chiodi, Adelfio (2017)<doi:10.18637/jss.v076.i03>.

Imports fields,maps

License GPL (>= 2)

Depends R (>= 3.5.0), mapdata

Author Marcello Chiodi [aut, cre],
Giada Adelfio [aut]

Maintainer Marcello Chiodi <marcello.chiodi@unipa.it>

LazyData true

RoxygenNote 7.1.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2023-09-14 14:00:02 UTC

R topics documented:

`etasFLP-package` *Mixed FLP and ML Estimation of ETAS Space-Time Point Processes*

Description

New version 2.2.0. Covariates have been introduced to explain the effects of external factors on the induced seismicity. Since the parametrization is changed, the etasclass object created with the previous versions are not compatible with the one obtained with the current version. Estimation of the components of an ETAS (Epidemic Type Aftershock Sequence) model for earthquake description. Non-parametric background seismicity can be estimated through FLP (Forward Likelihood Predictive), while parametric components are estimated through maximum likelihood. The two estimation steps are alternated until convergence is obtained. For each event the probability of being a background event is estimated and used as a weight for declustering steps. Many options to control the estimation process are present, together with some diagnostic tools. Some descriptive functions for earthquakes catalogs are included; also `plot`, `print`, `summary`, `profile` methods are defined for main output (objects of class `etasclass`); update methods are now present.

Details

Package:	etasFLP
Type:	Package
Version:	2.2.2
Date:	2023-09-14
License:	GPL (>=2)
Imports: fields,maps Depends:	R (>= 3.5.0), mapdata
Suggests:	MASS

`etasclass` is the main function of the package `etasFLP`: strongly renewed from version 2.0. `update.etasclass` and `timeupdate.etasclass` two new different kind of updating existing `etasclass` objects. Very useful for large catalogues

Note

The package is intended for the estimation of the ETAS model for seismicity description (introduced by Ogata (1988), see reference), but theoretically it can be used for other fields of application.

Author(s)

Marcello Chiodi and Giada Adelfio

Maintainer: Marcello Chiodi<marcello.chiodi@unipa.it>

References

Adelfio G., Chiodi, M. (2009). Second-Order Diagnostics for Space-Time Point Processes with Application to Seismic Events. *Environmetrics*, **20**(8), 895-911. doi:10.1002/env.961.

Adelfio, G., Chiodi, M. (2013) Mixed estimation technique in semi-parametric space-time point processes for earthquake description. *Proceedings of the 28th International Workshop on Statistical Modelling 8-13 July, 2013, Palermo* (Muggeo VMR, Capursi V, Boscaino G, Lovison G, editors). Vol. **1**. pp.65-70.

Adelfio, G., Chiodi, M. (2015) Alternated estimation in semi-parametric space-time branching-type point processes with application to seismic catalogs. *Stochastic Environmental Research and Risk Assessment* **29**(2), pp. 443-450. DOI: 10.1007/s00477-014-0873-8

Adelfio G., Chiodi, M. (2015). FLP Estimation of Semi-Parametric Models for Space-Time Point Processes and Diagnostic Tools. *Spatial Statistics*, **14**(B), 119-132. doi:10.1016/j.spasta.2015.06.004.

Adelfio G., Chiodi, M. (2020). Including covariates in a space-time point process with application to seismicity. *Statistical Methods and Applications* , doi:10.1007/s10260-020-00543-5.

Adelfio G., Schoenberg, FP (2009). Point Process Diagnostics Based on Weighted Second- Order Statistics and Their Asymptotic Properties. *The Annals of the Institute of Statistical Mathematics*, **61**(4), 929-948. doi:10.1007/s10463-008-0177-1.

Chiodi, M., Adelfio, G., (2011) Forward Likelihood-based predictive approach for space-time processes. *Environmetrics*, vol. **22** (6), pp. 749-757. DOI:10.1002/env.1121.

Chiodi, M., Adelfio, G., (2017) Mixed Non-Parametric and Parametric Estimation Techniques in R Package etasFLP for Earthquakes' Description. *Journal of Statistical Software*, vol. **76** (3), pp. 1-29. DOI: 10.18637/jss.v076.i03

Console, R., Jackson, D. D. and Kagan, Y. Y. Using the ETAS model for Catalog Declustering and Seismic Background Assessment. *Pure Applied Geophysics* **167**, 819–830 (2010). DOI:10.1007/s00024-010-0065-5.

Nicolis, O., Chiodi, M. and Adelfio G. (2015) Windowed ETAS models with application to the Chilean seismic catalogs, *Spatial Statistics*, Volume **14**, Part B, November 2015, Pages 151-165, ISSN 2211-6753, http://dx.doi.org/10.1016/j.spasta.2015.05.006.

Ogata, Y. Statistical models for earthquake occurrences and residual analysis for point processes. *Journal of the American Statistical Association*, **83**, 9–27 (1988).

Veen, A. , Schoenberg, F.P. Estimation of space-time branching process models in seismology using an EM-type algorithm. *Journal of the American Statistical Association*, **103**(482), 614–624 (2008).

Zhuang, J., Ogata, Y. and Vere-Jones, D. Stochastic declustering of space-time earthquake occurrences. *Journal of the American Statistical Association*, **97**, 369–379 (2002). DOI:10.1198/016214502760046925.

`b.guten` *Estimates the parameter of the Gutenberg-Richter law.*

Description

Estimates the parameter of the Gutenberg-Richter law for the magnitude distribution of earthquakes, given a threshold magnitude; it uses moment estimator on transformed data.

Usage

```
b.guten(magn, m0=min(magn))
```

Arguments

`magn`	a vector of magnitudes coming from an earthquake catalog.
`m0`	A threshold value. Only values of `magn` not less than `m0` will be used.

Details

Maximum likelihodd estimation for the Gutenberg-Richter Law:

$$\log_{10} N(> m) = a - b\, M$$

where $N(> m)$ is the number of events exceeding a magnitude m and a, b are two parameters: a is related to the total seismicity rate of the region while b, to be estimated, should be usually near 1.

Catalog is assumed to be complete (in a certain space-time region) at least for a magnitude `m0`, that is, every earthquake of magnitude at least `m0` in that space-time region, is certainly present in the catalog.

Value

`b`	estimate of the parameter b of the Gutenberg-Richter Law.
`se`	estimate of the standard error of the estimate `b`.

Note

the plot produced by `magn.plot` can be used to have an idea, for a given catalog, of the magnitude threshold value.

Author(s)

Marcello Chiodi

References

Gutenberg, B. and Richter, C. F. (1944). Frequency of earthquakes in California. *Bulletin of the Seismological Society of America*, 34, 185-188.

See Also

magn.plot

Examples

```
data(italycatalog)
b.guten(italycatalog$magn1)
```

---

bwd.nrd — *Silverman's rule optimal for the estimation of a kernel bandwidth*

Description

Computes the optimal bandwidth with the Silverman's rule of thumb, to be used for a kernel estimator with given points and weights.

Usage

```
bwd.nrd(x, w=replicate(length(x),1), d = 2)
```

Arguments

x	numeric vector: sample points to be used for a normal kernel estimator.
w	numeric vector of the same length of x: weights to give to the elements of x. Default is a vector of ones
d	number of dimensions of the kernel estimator.

Details

Computes the optimal bandwidth with the Silverman rule, for a kernel estimator with points x and weights w. If a multivariate kernel is used, (i.e. d > 1), bwd.nrd must be called for each variable. It computes dispersion only with the weighted standard deviation, with no robust alternative. Called by kde2dnew.fortran.

Value

The value of the bandwidth for a sample x and weights w.

Note

It is used in connection with the the declustering method of `etasFLP`. Points with an higher probability of being part of the background seismicity will weight more in the estimation of the background seismicity.

Note

This is a slight modification of `bw.nrd`.

Author(s)

Marcello Chiodi

References

Silverman, B.W. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman and Hall: London.

Examples

```
#####
```

`californiacatalog` *Sample catalog of North California earthquakes*

Description

Sample catalog of North California earthquakes of magnitude at least 3.0 from year 1968 to year 2012.

Usage

```
californiacatalog
```

Format

a data matrix with 18,545 observations and 5 variables: `time`, `lat`, `long`, `z`, `magn1`.

Source

Northern California Earthquake Data Center.

References

Northern California Earthquake Catalog Search: http://www.ncedc.org/ncedc/catalog-search.html.

Examples

```
data(californiacatalog)
str(californiacatalog)
```

---

catalog.withcov *Small sample catalog of italian earthquakes with covariates*

Description

A small sample catalog of italian earthquakes of magnitude at least 2.5 from May 2012 to May 2016 with extra information.

Usage

```
catalog.withcov
```

Format

a data matrix with 2,226 observations and 11 variables: `Date`, `id_ev`, `time`, `lat`, `long`, `z`, `magn1`, `err_h_rev`, `min_distance_rev`, `rms_rev`, `nstaloc_rev` .

Details

A small sample catalog of italian earthquakes of magnitude at least 2.5 from May 2012 to May 2016 with extra information. `Date` is the date, `id_ev` is an identifier of the events, `time`, `lat`, `long`, `z`, `magn1` are time (days from 1900-01-01), latitude, longitude, depth, magnitude, `err_h_rev` is an estimate of hypocentral uncertainty, `min_distance_rev` is the distance from the nearest station, `rms_rev` is a measure of the quality of the location, `nstaloc_rev` is the number of stations that registered the event, `distmin` is the distance from the nearest fault

Source

INGV (Istituto Nazionale di Geofisica e Vulcanologia) ISIDE Data Base.

References

INGV home page: https://www.ingv.it/.

Examples

```
data(catalog.withcov)
str(catalog.withcov)
```

`compare.etasclass` *Compare two etasclass objects*

Description

Compare the results of two `etasclass` executions through the comparison of some elements of two `etasclass` objects.

Usage

```
compare.etasclass(etas1,etas2)
```

Arguments

`etas1`	an `etasclass` object.
`etas2`	an `etasclass` object.

Details

Compare the results of two `etasclass` executions through the comparison of some elements of two `etasclass` objects estimated on the same catalog

Value

`diffstd.params`	Standardized comparison of the estimated parameters, included covariates.
`AIC`	Difference of AIC values.
`weigths`	Comparison of the weights `rho.weights` of the two input objects.
`rho.weigths`	Standardized comparison of the weights `rho.weights` of the two input objects.
`cor.weights`	Correlation between the weights `rho.weights` of the two input objects
`cor.trig`	Correlation between the triggered intensities of the two input objects
`cor.back`	Correlation between the background intensities of the two input objects

Author(s)

Marcello Chiodi

See Also

`etasclass`

daily.etasclass — *Title daily.etasclass*

Description

A daily estimation on a space grid is made

Usage

```
daily.etasclass(x, ngrid = 201, nclass = 20, tfixed = 0, flag.log = FALSE, ...)
```

Arguments

`x`	an etasclass object
`ngrid`	subdivisions of x and y axis for grid computation of intensities
`nclass`	number of class for horizontal and vertical axes of the output grid
`tfixed`	day of computation
`flag.log`	if log intensity must be used
`...`	other optional parameters

Value

a grid with daily theoretical intensities

eqcat — *Check earthquake catalog*

Description

Preliminary check of the names of an earthquake catalog. `summary` and `plot` methods for earthquake catalogs are defined.

Usage

```
eqcat(x)
## S3 method for class 'eqcat'
plot(x,extended=TRUE,...)
## S3 method for class 'eqcat'
summary(object,extended=TRUE,...)
```

Arguments

`x`	an earthquake catalog.
`object`	an `eqcat` object.
`extended`	if `TRUE` some extra summary functions are computed.
`...`	other arguments.

Details

Minimal check of an earthquake catalog; checks only if it is suitable for the use as argument of the functions of `etasFLP` (mainly `etasclass`); checks only the presence of variables with the names `time`, `lat`, `long`, `z`, `magn1`. `summary` and `plot` methods are defined for earthquake catalogs. and the input object can be the `cat` output of `eqcat`

Value

If the catalog passes the check, then the catalog is returned in the object `cat` with the new class name `eqcat`; otherwise an error message is printed.

`cat`	the input catalog is returned. If the check is ok, this is an `eqcat` class object.
`ok`	A flag: `TRUE` if the check is ok; `FALSE` elsewhere.

Note

In this first version if you have a catalog without the depth (`z`), please insert however a constant column. The depth can be used only in some plot and not in the estimation routines of the package etasFLP; `etasclass` uses only `time`, `lat`, `long`, `magn1`. From version 2.0 you could use `z` as a covariate for the triggered component

Author(s)

Marcello Chiodi

See Also

`etasclass`

Examples

```
## Not run:
data(italycatalog)
f=eqcat(italycatalog)
print(f$ok)
summary(f$cat)
plot(f$cat)

## End(Not run)
```

`etas.starting` *Guess starting values of ETAS parameters (beta-version). Only from package version 1.2.0*

Description

`etas.starting` is a simple function to give starting values of the 7 ETAS parameters for the function `etasclass`.

It gives only rough approximations, based on some assumptions, intended to give only the order of magnitude of each parameter (but should be better than nothing). Returns a list with starting values. In the present version user can give manually the output of this function in the input of `etasclass`. Otherwise, the function is called by `etasclass` at first steps, to supply initial values to start estimation.

Usage

```
etas.starting(cat.orig,
magn.threshold=2.5,
p.start=1,
gamma.start=0.5,
q.start=2,
betacov.start=.7,
longlat.to.km=TRUE,
sectoday=FALSE,
onlytime=FALSE
)
```

Arguments

`cat.orig`	An earthquake catalog, possibly an object of class `eqcat`, or however a `data.frame` with variables of names `time`, `lat`, `long`, `z`, `magn1`. No missing values are allowed.
`magn.threshold`	Threshold magnitude (only events with a magnitude at least `magn.threshold` will be used). Default value = 2.5.
`p.start`	Parameter 4 of the ETAS model; the exponent of the Omori law for temporal decay rate of aftershocks; see details. Default value = 1.0.
`gamma.start`	Parameter 5 (γ) of the ETAS model; together with `a` is related to the efficiency of an event of given magnitude in generating aftershocks; see details. Default value = 0.5.
`q.start`	Parameter 7 of the ETAS model; parameter related to the spatial influence of the mainshock; see details. Default value = 2.
`betacov.start`	coefficient of the covariate (as default the magnitude). Default value = 0.7.
`sectoday`	if `TRUE`, then `time` variable of `cat.orig` is converted from seconds to days. Default value = `TRUE`.
`longlat.to.km`	if `TRUE`, then `long` and `lat` variables of `cat.orig` are treated as geographical coordinates and converted to kilometers. Default value = `TRUE`.
`onlytime`	if `TRUE` then a time process is fitted to data , regardless to space location (in this case `is.backconstant` is set to `TRUE` and `declustering`, `flp` are set to `FALSE`). Default value = `FALSE`.

Details

It is a beta-version of a very crude method to give starting values for the seven parameters of an ETAS (Epidemic type aftershock sequences) model for the description of the seismicity of a space-time region. These starting values can be used as input for the function `etasclass` `sectoday` and `longlat.to.km` flags must the same that will be used in `etasclass`.

In this first attempt to give starting values for the ETAS model, many approximations are used

It gives only rough approximation, based on some assumptions, intended to give only the order of magnitude of each parameter (but it should be better than nothing). It returns a list with 7 starting values. With this beta-version user must give manually the output of this function in the input of `etasclass`.

The values of `p.start`, `gamma.start` and `q.start` must be however given by the user (we did not find anything reasonable). Default choices for `p` and `q` (`p.start=1`, `q.start=2`) are strongly reccomended.

`c` and `d` are estimated from the emprical distributions of time differences and space distances, respectively. `mu` and `k0` are then estimated given the other starting values, solving the two ML equations, that is derivatives of the whole likelihood with respect to `mu` and `k0` equated to zero. In the computation of the likelihood an approximation for the integral of the intensity function is used (quoted also in Schoenberg (2013)).

Value

returns a list:

`mu.start`	guess value for `mu`
`k0.start`	guess value for `k0`
`c.start`	guess value for `c`
`p.start`	guess value for `p` (the same as input value)
`gamma.start`	guess value for `gamma` (the same as input value)
`d.start`	guess value for `d`
`q.start`	guess value for `q` (the same as input value)
`longlat.to.km`	`longlat.to.km` (the same as input value)
`sectoday`	`sectoday` (the same as input value)

Note

The optimization algorithm used in `etasclass` depends on the choice of initial values. Some default guess choice is performed in the present beta-version of the function `etas.starting`. If convergence problem are experienced, a useful strategy can be to start with an high magnitude threshold value m_0 (that is, with a smaller catalog with bigger earthquakes), and then using this first output as starting guess for a running with a lower magnitude threshold value m_0. In this trial executions avoid declustering (`declustering=FALSE`) or at least use a small value of `ndeclust`; small values of `iterlim` and `ntheta` can speed first executions.

Quicker executions are obtained using smaller values of `iterlim` and `ntheta` in the input.

Also a first execution with `is.backconstant = TRUE`, to fit a first approximation model with constant background, can be useful.

Some other useful information can be obtained estimating a pure time process, that can give a good guess at least for some parameters, like μ, κ_0, c, p.

Input times are expected in days, and so final intensities are expected number of events per day. If input values are in seconds, then set `sectoday=TRUE`

Author(s)

Marcello Chiodi, Giada Adelfio

References

Schoenberg, F. P. (2013).Facilitated Estimation of ETAS. *Bulletin of the Seismological Society of America*, Vol. 103, No. 1, pp. 601-605, February 2013, doi: 10.1785/0120120146

See Also

`etasclass`

`etasclass` *Mixed estimation of an ETAS model (renewed in version 2.0)*

Description

`etasclass` is the main function of the package `etasFLP`.

`etassclass` *objects of previous versions are not compatible with the current version*

Performs the estimation of the components of the ETAS (Epidemic Type Aftershock Sequence) model for the description of the seismicity in a space-time region. Background seismicity is estimated non-parametrically, while triggered seismicity is estimated by MLE. In particular also the bandwidth for a kernel smoothing can be estimated through the Forward Likelihood Predictive (FLP) approach. For each event the probability of being a background event or a triggered one is estimated.

New in version 2.0.0: *Covariates have been introduced to explain the effects of external factors on the induced seismicity. Since the parametrization is changed, the etasclass object created with the previous versions are not compatible with the one obtained with the current version.*

New in version 2.2.0: *New algoruthm for starting values. A new argument (n.iterweight) and an update method and a* `timeupdate` *option*

An ETAS with up to `7+ncov` parameters can be estimated, with several options and different methods.

Returns an `etasclass` object, for which `plot`, `summary`, `print` and `profile` methods are defined.

Usage

```
etasclass(cat.orig,
           time.update=FALSE,
           magn.threshold =2.5,
           magn.threshold.back=magn.threshold+2,
           tmax =max(cat.orig$time),
           long.range=range(cat.orig$long),
           lat.range=range(cat.orig$lat),
           ##### starting values for parameters
           mu =1,
           k0 =1,
           c =0.5,
           p =1.01,
           gamma =.5,
           d =1.,
           q =1.5,
           betacov         =0.7,
           ### indicators: if params.ind[i] i-th parameter will be estimated
           params.ind=replicate(7,TRUE),
#           params.lim=c(0,0,0,1.0,0,0,0),
           ### formula for covariates (magnitude should always be included):
           formula1            ="time~magnitude-1",
           offset              =0,
           hdef=c(1,1),
           w =replicate(nrow(cat.orig),1),
           hvarx  =replicate(nrow(cat.orig),1),
           hvary  =replicate(nrow(cat.orig),1),
           ### flags for the kind of declustering and smoothing:
           declustering    =TRUE,
           thinning        =FALSE,
           flp          =TRUE,
           m1            =NULL,
           ndeclust        =5,
           n.iterweight    =1,
           onlytime =FALSE,
           is.backconstant =FALSE,
           ##### end of  main input arguments.
           ##### Control and secondary arguments:
           description ="",
           cat.back    =NULL,
           back.smooth =1.0,
           sectoday =FALSE,
           longlat.to.km   =TRUE,
#           fastML=FALSE, #### not yet implemented
#           fast.eps=0.001, #### not yet implemented
           usenlm =TRUE,
           method ="BFGS",
           compsqm  =TRUE,
```

```
epsmax = 0.0001,
iterlim =50,
ntheta =36)
```

Arguments

`cat.orig` An earthquake catalog, possibly an object of class `eqcat`, or however a `data.frame` with variables of names `time`, `lat`, `long`, `z`, `magn1`. No missing values are allowed.

`time.update` Logical. It is `TRUE` if the excution is called by `time.update` to update to new time maximum. Default value = `FALSE`.

`magn.threshold` Threshold magnitude (only events with a magnitude at least `magn.threshold` will be used). Default value = 2.5.

`magn.threshold.back` Threshold magnitude used to build the catalog `cat.back` for the first estimation of the background seismicity. Default value = `magn.threshold+2`.

`tmax` Maximum value of time. Only observations before `tmax` will be used for estimation. Default value = `max(cat.orig$time)`.

`long.range` Longitude range. Only observations with `long` in the range `long.range` will be used for estimation. Default value = `range(cat.orig$long)`.

`lat.range` Latitude range. Only observations with `lat` in the range `lat.range` will be used for estimation. Default value = `range(cat.orig$lat)`.

Values for the 7 parameters of the ETAS model (starting values or fixed values according to `params.ind`):

`mu` Parameter 1 (μ) of the ETAS model: background general intensity; see details. Default value = 1.

`k0` Parameter 2 (κ_0) of the ETAS model: measures the strength of the aftershock activity; see details. Default value = 1.

`c` Parameter 3 of the ETAS model; a shift parameter of the Omori law for temporal decay rate of aftershocks; see details. Default value = 0.5.

`p` Parameter 4 of the ETAS model; the exponent of the Omori law for temporal decay rate of aftershocks; see details. Default value = 1.01.

`gamma` Parameter 5 (γ) of the ETAS model; together with `a` is related to the efficiency of an event of given magnitude in generating aftershocks; see details. Default value = 0.5.

`d` Parameter 6 of the ETAS model; parameter related to the spatial influence of the mainshock; see details. Default value = 1.

`q` Parameter 7 of the ETAS model; parameter related to the spatial influence of the mainshock; see details. Default value = 1.5.

`betacov` Numerical array. Parameters of the covariates ETAS model (the parameters β_j); see details. Default value = 0.7. Parameters in `betacov` are not limited

End of model pararameter input

params.ind
: vector of 7 logical values: `params.ind[i] = TRUE` means that the i-th parameter must be estimated. `params.ind[i] = FALSE` means that the i-th parameter is fixed to its input value (the order of parametrs is: `mu`, `k0`, `c`, `p`, `gamma`, `d`, `q`). Default value = `replicate(7,TRUE)`, that is, `etasclass` estimates all parameters.

params.lim
: vector of 7 numerical values: `params.lim[i] = theta0` means that the i-th parameter must be greater than `theta0` (the default limits of parametrs are: 0 for `mu`, `k0`, `c`, 1 for `p`, 0 for `gamma`, `d`, `q`). Default value = `replicate(7,TRUE)`, that is, `etasclass` estimates all parameters.

formula1
: a character variable: Formula which defines the covariates acting on the induced seismicity. In classical etas model the covariate is the magnitude. The left side (dummy) element must be the time, which is a variable certainly present in the data set. The right part of the formula determines `ncov` the number of covariates. Default value=`"time~magnitude-1"`; input *must* be a character value: it is converted in a formula inside the program

offset
: An offset, for which no parameter will be estimated. Default value=0

Flags for the kind of declustering and smoothing:

hdef
: Starting values for the `x,y` bandwidths used in the kernel estimator of background seismicity. Default value = `1,1`.

w
: Starting values for the weigths used in the kernel estimator of background seismicity. The length must be equal to the number of events of the catalog after event selection (can be less than `nrow(cat.orig)`).
: Default value = `replicate(nrow(cat.orig),1)`.

hvarx
: Longitude bandwidths adjustement used in the kernel estimator of background seismicity. The length must be equal to the number of events of the catalog after event selection (can be less than `nrow(cat.orig)`). Default value = NULL

hvary
: Longitude bandwidths adjustement used in the kernel estimator of background seismicity. The length must be equal to the number of events of the catalog after event selection (can be less than `nrow(cat.orig)`). Default value = NULL

declustering
: if `TRUE` the catalog is iteratively declustered to optimally estimate the background intensity (through thinning, if `thinning=TRUE`, or through weighting if `thinning=FALSE`). Default value = `TRUE`.

thinning
: if `thinning=TRUE` a background catalog is obtained sampling from the original catalog with probabilities estimated during the iterations. Default value =`FALSE`.

flp
: if `flp=TRUE` then background seismicity is estimated through Forward Likelihood Predictive (see details). Otherwise the Silverman rule is used. Default value =`TRUE`.

m1
: Used only if `flp=TRUE`. Indicates the range of points used for the FLP steps. See details. If missing it is set to `nrow(cat)/2`.

ndeclust
: maximum number of iterations for the general declustering procedure. Default=5.

n.iterweight
: New in version 2.2. The weighting and the density computations will be alternated `n.iterweight` times after each maximum likelihood step: in many situations this improves the general convergence procedure. Default=1.

`onlytime` if `TRUE` then a time process is fitted to data , regardless to space location (in this case `is.backconstant` is set to `TRUE` and `declustering`, `flp` are set to `FALSE`). Default value = `FALSE`.

`is.backconstant` if `TRUE` then background seismicity is assumed to be homogeneous in space (and `declustering, flp` are set to `FALSE`). Default value = `FALSE`.

Other control parameters:

`description` a description string used for the output. Default value = `""`.

`cat.back` external catalog used for the estimation of the background seismicity. Default value = `NULL`.

`back.smooth` Controls the level of smoothing for the background seismicity (meaningful only if `flp=FALSE`). Default value = 1.

`sectoday` if `TRUE`, then `time` variable of `cat.orig` is converted from seconds to days. Default value = `FALSE`.

`longlat.to.km` if `TRUE`, then `long` and `lat` variables of `cat.orig` are treated as geographical coordinates and converted to kilometers. Default value = `TRUE`.

`usenlm` if `TRUE`, then `nlm` function (gauss-newton method) is used in the maximum likelihood steps; if `FALSE`, then `optim` function is used (with `method =method`). Default value = `TRUE`.

`method` used if `usenlm=FALSE`: method used by `optim`. Default value = `"BFGS"`.

`compsqm` if `TRUE`, then standard errors are computed. Default value = `TRUE`.

`epsmax` maximum allowed difference between estimates in subsequent iterations (default = 0.0001).

`iterlim` maximum number of iterations in the maximum likelihood steps (used in `nlm` or `optim`). Default value = 100.

`ntheta` number of subdivisions of the round angle, used in the approximation of the integral involved in the likelihood computation of the ETAS model. Default value = 100.

Details

Estimates the components of an ETAS (Epidemic type aftershock sequence) model for the description of the seismicity of a space-time region. Background seismicity is estimated nonparametrically, while triggered seismicity is estimated by MLE.

From version 2.0 of package `etasFLP` covariates are allowed to improve the fitting of the triggered part, through the input `formula1`, which as a default values of `"time ~ magnitude - 1"`, which corresponds to the previous version of package `etasFLP`, that is, magnitude as the only covariate which influence the average number of aftershocks.

The bandwidth of the kernel density estimator is estimated through the Forward Likelihood Predictive approach (FLP), (theoretical reference on Adelfio and Chiodi, 2013) if `flp` is set to `TRUE`. Otherwise the bandwidth is estimated trough the Silverman's rule. FLP steps for the estimation of nonparametric background component is alternated with the Maximum Likelihood step for the estimation of parametric components (only if `declustering=TRUE`). For each event the probability of being a background event or a triggered one is estimated, according to a declustering procedure in a way similar to the proposal of Zhuang, Ogata, and Vere-Jones (2002).

The ETAS model for conditional space time intensity $\lambda(x, y, t)$ is given by:

$$\lambda(x,y,t) = \mu f(x,y) + \kappa_0 \sum_{t_j<t} \frac{e^{\eta_j}}{(t-t_j+c)^p} \left\{ \frac{(x-x_j)^2+(y-y_j)^2}{e^{\gamma\ (m_j-m_0)}} + d \right\}^{-q}$$

where $\eta_j = \sum_{j=1,ncov} \beta_j cov_{ij}$

parameters β_j are the elements of the array variable `betacov`

$f(x, y)$ is estimated through a weighted kernel gaussian estimator; if `flp` is set to `TRUE` then the bandwidth is estimated through a FLP step.

Weights (computed only if `declustering=TRUE`) are given by the estimated probabilities of being a background event; for the i-th event this is given by $\rho_i = \frac{\mu f(x_i,y_i)}{\lambda(x_i,y_i,t_i)}$. The weights ρ_i are updated after a whole iteration.

`mu` (μ) measures the background general intensity (which is assumed temporally homogeneous);

`k0` (κ_0) is a scale parameter related to the importance of the induced seismicity;

`c` and `p` are the characteristic parameters of the seismic activity of the given region; `c` is a shift parameter while `p`, which characterizes the pattern of seismicity, is the exponent parameter of the modified Omori law for temporal decay rate of aftershocks;

$\eta_j = \sum_{j=1,ncov} \beta_j cov_{ij}$ measures the efficiency of an event of a given magnitude in generating aftershock sequences;

`d` and `q` are two parameters related to the spatial influence of the mainshocks.

Many kinds of ETAS models can be estimated, managing some control input arguments. The eight ETAS parameters can be fixed to some input value, or can be estimated, according to `params.ind`: if `params.ind`[i]=FALSE the i-th parameter is kept fixed to its input value, otherwise, if `params.ind[i] = TRUE`, the i-th parameter is estimated and the input value is used as a starting value.

By default `params.ind=c(TRUE,TRUE,TRUE,TRUE,TRUE,TRUE,TRUE)`, and so a full 7+`ncov` parameters ETAS model will be estimated.

The seven parameters are internally ordered in this way: `params` = (`mu`, `k0`, `c`, `p`, `gamma`, `d`, `q`); for example a model with a fixed value `p=1` (and `params.ind`[4] = FALSE) can be estimated and compared with the model where `p` is estimated (`params.ind`[4]=TRUE);

for example a 6+`ncov` parameters model can be fitted with `gamma=0` and `params.ind[5]=FALSE`, so that input must be in this case: `params.ind=c(TRUE,TRUE,TRUE,TRUE,FALSE,TRUE,TRUE)`;

if `onlytime=TRUE` a time process is fitted to data (with a maximum of 5 parameters), regardless to space location (however the input catalog `cat.orig` must contain three columns named `long`, `lat`, `z`);

if `is.backconstant=TRUE` a process (space-time or time) with a constant background intensity μ is fitted;

if `mu` is fixed to a very low value a process with very low background intensity is fitted, that is with only clustered intensity (useful to fit a model to a single cluster of events).

If `flp=TRUE` the bandwidth for the kernel estimation of the background intensity is evaluated maximizing the Forward Likelihood Predictive (FLP) quantity, given by (Chiodi, Adelfio, 2011; Adelfio, Chiodi, 2013):

$$FLP_{k_1,k_2}(\hat{\boldsymbol{\psi}}) \equiv \sum_{k=k_1}^{n-1} \delta_{k,k+1}(\hat{\boldsymbol{\psi}}(H_{t_k}); H_{t_{k+1}})$$

with $k_1 = \frac{n}{2}, k_2 = n-1$ and where $\delta_{k,k+1}(\hat{\boldsymbol{\psi}}(H_{t_k}); H_{t_{k+1}})$ is the *predictive information* of the first k observations on the $k+1$-th observation, and is so defined:

$$\delta_{k,k+1}(\hat{\boldsymbol{\psi}}(H_{t_k}); H_{t_{k+1}}) \equiv \log L(\hat{\boldsymbol{\psi}}(H_{t_k}); H_{t_{k+1}}) - \log L(\hat{\boldsymbol{\psi}}(H_{t_k}); H_{t_k})$$

where H_k is the history of the process until time t_k and $\hat{\boldsymbol{\psi}}(H_{t_k})$ is an estimate based only on history until the $k-th$ observation.

In the ML step, the vector of parameter $\theta = (\mu, \kappa_0, c, p, \alpha, \gamma, d, q)$ is estimated maximizing the sample log-likelihood given by:

$$\log L(\boldsymbol{\theta}; H_{t_n}) = \sum_{i=1}^{n} \log \lambda(x_i, y_i, t_i; \boldsymbol{\theta}) - \int_{T_0}^{T_{max}} \int\int_{\Omega_{(x,y)}} \lambda(x, y, t; \boldsymbol{\theta})\, dx\, dy\, dt$$

Value

returns an object of class `etasclass`.

The main items of the output are:

- `this.call` — reports the exact call of the function
- `params.ind` — indicates which parameters have been estimated (see details)
- `params` — ML estimates of the ETAS parameters.
- `sqm` — Estimates of standard errors of the ML estimates of the ETAS parameters (`sqm[i]=0` if `params.ind[i]=FALSE` or where the hessian is not computed or near to singularity).
- `AIC.iter` — AIC values at each iteration.
- `hdef` — final bandwidth used for the kernel estimation of background spatial intensity (however estimated, with `flp=TRUE` or `flp=FALSE`).
- `rho.weights` — Estimated probability for each event to be a background event (ρ).
- `time.res` — rescaled time residuals (for time processes only).
- `params.iter` — A matrix with estimates values at each iteration.
- `sqm.iter` — A matrix with the estimates of the standard errors at each iteration.
- `rho.weights.iter` — A matrix with the values of `rho.weights` at each iteration.
- `l` — A vector with estimated intensities, corresponding to observed points

`summary`, `print` and `plot` methods are defined for an object of class `etasclass` to obtain main output.

A `profile` method (`profile.etasclass`) is also defined to make approximate inference on a single parameter

Note

In this version the x-y space region, where the point process is defined, is a rectangle embedding the catalog values.

The optimization algorithm depends on the choice of initial values. Some default guess choice is performed inside the function for parameters without input starting values; the function `etas.starting` gives rough first guess for initial values. If convergence problem are experienced, a useful strategy can be starting with an higher magnitude threshold value m_0 (that is, with a smaller catalog with bigger earthquakes), and then using this first output as starting guess for a running with a lower magnitude threshold value m_0. In this trial executions avoid declustering (`declustering=FALSE`) or at least use a small value of `ndeclust`; small values of `iterlim` and `ntheta` can speed first executions.

Quicker executions are obtained using smaller values of `iterlim` and `ntheta` in the input.

Also a first execution with `is.backconstant = TRUE`, to fit a first approximation model with constant background, can be useful.

Some other useful information can be obtained estimating a pure time process, that can give a good guess at least for some parameters, like $\mu, \kappa_0, \alpha, c, p$.

Input times are expected in days, and so final intensities are expected number of events per day. If input values are in seconds, then set `sectoday=TRUE`

Author(s)

Marcello Chiodi, Giada Adelfio

References

Adelfio, G. and Chiodi, M. (2013) Mixed estimation technique in semi-parametric space-time point processes for earthquake description. *Proceedings of the 28th International Workshop on Statistical Modelling 8-13 July, 2013, Palermo* (Muggeo V.M.R., Capursi V., Boscaino G., Lovison G., editors). Vol. **1** pp.65-70.

Adelfio G, Chiodi M (2015). Alternated Estimation in Semi-Parametric Space-Time Branching-Type Point Processes with Application to Seismic Catalogs. *Stochastic Environmental Research and Risk Assessment*, **29**(2), 443-450. doi:10.1007/s00477-014-0873-8.

Adelfio G, Chiodi M (2015). FLP Estimation of Semi-Parametric Models for Space-Time Point Processes and Diagnostic Tools. *Spatial Statistics*, **14**(B), 119-132. doi:10.1016/j.spasta.2015.06.004.

Adelfio G., Chiodi, M. (2020). Including covariates in a space-time point process with application to seismicity. *Statistical Methods and Applications*, doi:10.1007/s10260-020-00543-5.

Chiodi, M. and Adelfio, G., (2011) Forward Likelihood-based predictive approach for space-time processes. *Environmetrics*, vol. **22** (6), pp. 749-757. DOI:10.1002/env.1121.

Chiodi, M. and Adelfio, G., (2017) Mixed Non-Parametric and Parametric Estimation Techniques in R Package etasFLP for Earthquakes' Description. *Journal of Statistical Software*, vol. **76** (3), pp. 1-29. DOI: 10.18637/jss.v076.i03.

Zhuang, J., Ogata, Y. and Vere-Jones, D. Stochastic declustering of space-time earthquake occurrences. *Journal of the American Statistical Association*, **97**, 369–379 (2002). DOI:10.1198/016214502760046925.

See Also

eqcat, plot.etasclass,print.etasclass, summary.etasclass, profile.etasclass, etas.starting

Examples

```
## Not run:
data("italycatalog")
# load a sample catalog of the italian seismicity
esecov1<-etasclass(cat.orig = catalog.withcov, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 0.99, gamma = 0, d = 1,
q = 1.5, params.ind = c(TRUE, TRUE, TRUE, TRUE, FALSE, TRUE,
TRUE), formula1 = "time ~  magnitude- 1", declustering = TRUE,
thinning = FALSE, flp = TRUE, ndeclust = 15, onlytime = FALSE,
is.backconstant = FALSE, sectoday = FALSE, usenlm = TRUE,
compsqm = TRUE, epsmax = 1e-04, iterlim = 100, ntheta = 36)

# execution of etasclass for events with minimum magnitude of 3.0.
# The events with magnitude at least 3.5 are used to build a first approximation
# for the background intensity function
# (magn.threshold.back=3.5)
# The magnitude effect is given by the covariate magnitude
# in the formula "time ~  magnitude- 1"
# magnitude is the internal name for magn1-magn.threshold
# print method for the etasclass object

print(esecov1)

> print(esecov1)
Call:

etasclass(cat.orig = catalog.withcov, magn.threshold = 2.5, magn.threshold.back = 3.9,
    mu = 0.3, k0 = 0.02, c = 0.015, p = 0.99, gamma = 0, d = 1,
    q = 1.5, params.ind = c(TRUE, TRUE, TRUE, TRUE, FALSE, TRUE,
        TRUE), formula1 = "time ~  magnitude- 1", declustering = TRUE,
    thinning = FALSE, flp = TRUE, ndeclust = 15, onlytime = FALSE,
    is.backconstant = FALSE, sectoday = FALSE, usenlm = TRUE,
    compsqm = TRUE, epsmax = 1e-04, iterlim = 100, ntheta = 36)




Number of observations             2226
ETAS Parameters:
        mu          k0           c           p       gamma           d           q
  0.667509    0.022393    0.014769    1.110059    0.000000    1.905461    1.947223
magnitude
  0.740109
# summary method for more informative output etasclass object

summary(esecov1)
# plot results with maps of intensities and diagnostic tools
```

```
plot(esecov1)

## an application with 5 covariates
esecov5<-etasclass(cat.orig = catalog.withcov, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 0.99, gamma = 0, d = 1,
q = 1.5, params.ind = c(TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, TRUE),
formula1 = "time ~ z + magnitude +nstaloc_rev +min_distance_rev+distmin- 1",
declustering = TRUE, thinning = FALSE, flp = TRUE, ndeclust = 15,
onlytime = FALSE, is.backconstant = FALSE, sectoday = FALSE,
usenlm = TRUE, compsqm = TRUE, epsmax = 1e-04, iterlim = 100,
ntheta = 36)

## print results, more out put with summary
 print(esecov5)
Call:

etasclass(cat.orig = catalog.withcov, magn.threshold = 2.5, magn.threshold.back = 3.9,
    mu = 0.3, k0 = 0.02, c = 0.015, p = 0.99, gamma = 0, d = 1,
    q = 1.5, params.ind = c(TRUE, TRUE, TRUE, TRUE, FALSE, TRUE,
      TRUE), formula1 = "time ~ z + magnitude +nstaloc_rev +min_distance_rev+distmin- 1",
    declustering = TRUE, thinning = FALSE, flp = TRUE, ndeclust = 15,
    onlytime = FALSE, is.backconstant = FALSE, sectoday = FALSE,
    usenlm = TRUE, compsqm = TRUE, epsmax = 1e-04, iterlim = 100,
    ntheta = 36)




Number of observations             2226
ETAS Parameters:
              mu               k0                c                p
        0.705351         0.073070         0.019396         1.154186
           gamma                d                q                z
        0.000000         1.942929         2.004915        -0.041256
      magnitude      nstaloc_rev min_distance_rev          distmin
        1.157698        -0.009010        -0.011020        -1.826717


## End(Not run)
```

`italycatalog` *Small sample catalog of italian earthquakes*

Description

A small sample catalog of italian earthquakes of magnitude at least 3.0 from year 2005 to year 2013.

Usage

```
italycatalog
```

Format

a data matrix with 2,158 observations and 5 variables: `time`, `lat`, `long`, `z`, `magn1`.

Source

INGV (Istituto Nazionale di Geofisica e Vulcanologia) ISIDE Data Base.

References

INGV home page: https://www.ingv.it/.

Examples

```
data(italycatalog)
str(italycatalog)
```

---

kde2dnew.fortran *A 2-d normal kernel estimator*

Description

A simple and quick 2-d weighted normal kernel estimator, with fixed bandwidth and relative integral.

Usage

```
kde2dnew.fortran(
#         parallel=FALSE,
          xkern,ykern,gx,gy,h,
          factor.xy=1,eps=0,w=replicate(length(xkern),1),
hvarx=replicate(length(xkern),1),hvary=replicate(length(xkern),1)
)

kde2d.integral(xkern,ykern,gx=xkern,gy=ykern,eps=0,factor.xy=1,
        h  = c(  bwd.nrd(xkern,w),bwd.nrd(ykern,w)),w=replicate(length(xkern),1),
            hvarx=replicate(length(xkern),1),hvary=replicate(length(xkern),1)
            )
```

Arguments

`xkern`	x-values of kernel points of length `n` (`n=length(xkern)`).
`ykern`	y-values of kernel points of length `n`.
`gx`	x-values of the points where densities must be estimated.
`gy`	y-values of the points where densities must be estimated.
`h`	bandwidths: a length 2 numerical vector.

`eps` enlargment factor for the region of interest.

`factor.xy` expansion factor for bandwidths (density will be smoother if `factor.xy`>1).

`w` vector of weights to give to observed points (length `n`).

[]

`hvarx` Longitude bandwidths adjustement used in the kernel estimator of background seismicity. The length must be equal to the number of events of the catalog after event selection (can be less than `nrow(cat.orig)`). Default value = `replicate(length(xkern),1)`

`hvary` Longitude bandwidths adjustement used in the kernel estimator of background seismicity. The length must be equal to the number of events of the catalog after event selection (can be less than `nrow(cat.orig)`).Default value = `replicate(length(xkern),1)`

Details

A standard bivariate normal kernel estimator.

Value

grid values and estimated densities.

Author(s)

Marcello Chiodi.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.
Wand, M.P and Jones, M.C. (1995). *Kernel Smoothing*. London: Chapman & Hall/CRC.

`magn.plot` *Transformed plot of the magnitudes distribution of an earthquakes catalog*

Description

Plots the logarithm of the cumulative frequency of eccedence vs. magnitude in an earthquake catalog.

Usage

```
magn.plot(catalog, main = "Transformed plot of magnitude frequencies", ...)
```

Arguments

`catalog`	should be a `eqcat` object, or at least must contain a column with name `magn1`.
`main`	Title to give to the plot
`...`	other arguments to be passed to `plot()`

Details

For each magnitude m_j, if N_j is the number of values of `magn1` greater than m_j, the values of $\log(N_j)$ vs. m_j are plotted.

According to the Gutenberg-Richter law, this plot should be linear. If there is a linear behaviour only for values greater than a given m_0, then m_0 is probably the magnitude threshold of the catalog.

Value

A new plot is printed (see details).

Author(s)

Marcello Chiodi.

Examples

```
## Not run:
data(italycatalog)
magn.plot(italycatalog)

## End(Not run)
```

`MLA.freq` *Display a pretty frequency table*

Description

Display a pretty frequency table. It is only a wrapper to the function `table` but with a richer output, at least for numerical variables.

Usage

```
MLA.freq(x)
```

Arguments

`x`	a numeric vector.

Details

The output gives the different kinds of frequencies and cumulated frequencies: single frequencies, cumulated and back cumulated (absolute and relatives).

Value

return a matrix with 7 columns: the modaldistinct values of x, frequencies, relative frequencies, cumulated frequencies, cumulated relative frequencies, back cumulated frequencies and back cumulated relative frequencies.

Author(s)

Marcello Chiodi

Examples

```
x=trunc(runif(1000)*10)
MLA.freq(x)

data(italycatalog)
MLA.freq(italycatalog$magn1)
```

`plot.etasclass` *Plot method for etasclass objects*

Description

This is the main method to visualize graphically the output of an object of class `etasclass`.

By default the space-time region is the same used for the estimation of the ETAS model. Background, triggered and total space intensities are also plotted for a grid of values.

Usage

```
## S3 method for class 'etasclass'
plot(x,pdf=FALSE,file ="etasplot", ngrid=201,nclass=10,tfixed=0,flag.log=FALSE,...)
```

Arguments

`x`	an `etaclass` object.
`pdf`	If `TRUE`, then 2D plots are sent to a pdf file
`file`	name of the pdf file
`ngrid`	number of points for each direction (`x`, `y`) of a `ngrid*ngrid` grid where estimated intensities must be evaluated. Default value= 201.
`nclass`	number of class for each direction (`x`, `y`) of a grid of `nclass*nclass` cells where estimated intensities must be evaluated. Must divide `ngrid-1`. Default value= 10.

`tfixed`	If a positive value is given, then the triggered intensity at time `tfixed` is estimated and visualized.
`flag.log`	If `TRUE` then a `log` scale is used to plot intensities.
`...`	other arguments.

Details

Different plots of the output of an object of class `etasclass`.

By default the space-time region is the same used for the estimation of the ETAS model. Background, triggered and total space intensities are also computed and plotted for a grid of values.

If a positive value is given for `tfixed`, then the triggered intensity at time `tfixed` is estimated and visualized. A tipical use can be with `tfixed` a day after a big earthquake.

For space dimension, four plot are drawn with triggered, observed, total intensity, and total intensity with points.

Starting with the package version 1.2.0 different kind of residual analysis are computed and visualized, separately for the space and time dimensions. (8 plot on three windows for the space and 2 plots on one window for the time)

Then two plots are printed for space residuals for total and background intensities

Space residuals are computed dividing the observed rectangular space area in a equally spaced grid of `nclass` intervals for each dimension, so to divide the observed space area in `nclass` x `nclass` rectangular cells. We obtain the classical comparison between observed and theoretical frequencies. All frequencies are related to the whole time interval (and thus theoretical frequencies are obtained integrating estimated intensities with respect to time).

Fifth graph (image plot)

We define `nclass` x `nclass` standardized residuals:

$$z_{\ell j} = \frac{n_{\ell j} - \hat{\nu}_{\ell j}}{\sqrt{\hat{\nu}_{\ell j}}} \qquad (\ell = 1, 2, ..., nclass;\ j = 1, 2, ..., nclass)$$

For each cell ℓj we have observed ($n_{\ell j}$) and theoretical frequency ($\hat{\nu}_{\ell j}$).

Sixth graph (image plot)

We used a similar technique to compute residuals for the background seismicity only, to check if at least the estimation of the background component is appropriate. To this purpose the observed background frequencies (${}_b n_{\ell j}$) are now computed by the sum of the estimated weights `rho.weights` and the theoretical background frequency ${}_b\hat{\nu}_{\ell j}$ by the estimated marginal space background intensity in each cell.

From these quantities we obtain `nclass` x `nclass` standardized residuals for the background intensity only:

$${}_b z_{\ell j} = \frac{{}_b n_{\ell j} - {}_b\hat{\nu}_{\ell j}}{\sqrt{{}_b\hat{\nu}_{\ell j}}} \qquad (\ell = 1, 2, ..., nclass;\ j = 1, 2, ..., nclass)$$

seventh plot: (space intensities (integrated over time))

A 3x2 plot: first column for observed vs.theoretical , second column for standardized residuals vs theoretical values. First row for total intensity, second row for background intensity, and third row for their difference, the triggered intensities eight-th graph:

To check departure of the model for the time dimension, we first integrated the estimated intensity function with respect to the observed space region, so to obtain an estimated time process (a one dimensional ETAS model):

$$\hat{\lambda}(t) = \int\int_{\Omega_{(x,y)}} \hat{\lambda}(x, y, t)\, dx\, dy$$

As known, a non-homogeneous time process can be transformed to a homogeneous one through the integral transformation:

$$\tau_i = \int_{t_0}^{t_i} \hat{\lambda}(t)\, dt$$

Then, a plot of τ_i versus i can give information about the departures of the models in the time dimension. In particular, this plot, together with a plot of the estimated time intensities, drawn on the same graphic winodw, can inform on the time at which departures are more evident

If `pdf=TRUE` all graphs are printed on a pdf file, as spcified by `file`; otherwise default screen device is used.

Value

This `plot` method computes, among others, `back.grid`, `trig.grid`, with coordinates `x.grid` and `y.grid` used to obtain image plots of background, triggered and total spatial estimated intensities (see `etasclass` to see the details of the mixed estimation method used).

`x.grid`	x grid values.
`y.grid`	y grid values.
`back.grid`	background intensity estimated on a `ngrid` x `ngrid` grid.
`trig.grid`	triggered intensities estimated on a grid of `ngrid` x `ngrid` points.
`tot.grid`	total intensities estimated on a grid of `ngrid` x `ngrid` points.
`tfixed`	the fixed time for which intensity is estimated and visualized.
`totfixed.grid`	total intensities estimated on a grid of `ngrid` x `ngrid` points at time `tfixed`.
`back.grid`	background space intensity estimated for observed points.
`trig.grid`	triggered space intensities estimated for observed points.
`tot.grid`	total space intensities estimated for observed points.
`teo1`	matrix of `nclass*nclass` cells with theoretical total space intensities.
`teo2`	matrix of `nclass*nclass` cells with theoretical background space intensities.
`emp1`	matrix of `nclass*nclass` cells with empirical total space intensities.
`emp2`	matrix of `nclass*nclass` cells with empirical background space intensities.
`t.trasf`	vector of transformed times.

Note

In this first version the x-y space region, where the point process is defined, by default is a rectangle embedding the catalog values.

Author(s)

Marcello Chiodi, Giada Adelfio

References

Adelfio G, Chiodi M (2009).Second-Order Diagnostics for Space-Time Point Processes with Application to Seismic Events. *Environmetrics*, **20**(8), 895-911. doi:10.1002/env.961.

Adelfio G, Chiodi M (2015). FLP Estimation of Semi-Parametric Models for Space-Time Point Processes and Diagnostic Tools. *Spatial Statistics*, **14**(B), 119-132. doi:10.1016/j.spasta.2015.06.004.

Adelfio G, Schoenberg FP (2009). Point Process Diagnostics Based on Weighted Second- Order Statistics and Their Asymptotic Properties. *The Annals of the Institute of Statistical Mathematics*, **61**(4), 929-948. doi:10.1007/s10463-008-0177-1.

Chiodi, M. and Adelfio, G., (2017) Mixed Non-Parametric and Parametric Estimation Techniques in R Package etasFLP for Earthquakes' Description. *Journal of Statistical Software*, vol. **76** (3), pp. 1-28. DOI: 10.18637/jss.v076.i03.

See Also

etasclass, eqcat, profile.etasclass

Examples

```
## Not run:
data("italycatalog")
# load a sample catalog of the italian seismicity

class(italycatalog)<-"eqcat"


# plot method

plot(\code{an etasclass object})


## End(Not run)
```

---

plot.profile.etasclass   *plot method for profile.etasclass objects (profile likelihood of ETAS model)*

Description

`plot` method for `profile.etasclass` objects (profile likelihood of ETAS model). Plots a smooth interpolation of the profile likelihood of a parameter of an ETAS model, as output from `profile.etasclass`.

Usage

```
## S3 method for class 'profile.etasclass'
plot(x,prob=c(0.90,0.95,0.99), use.main = TRUE,...)
```

Arguments

x	An object of the class `profile.etasclass`.
prob	A vector of coverage probability for the asymptotic confidence interval computed using -2log(LR). Default value `prob=c`(0.90,0.95,0.99).
use.main	Logical. If `use.main=TRUE` then a title is printed in the profile plot
...	other arguments.

Details

Plots a spline interpolation of the profile likelihood for a parameter of the ETAS model for earthquake seismicity, computed with `profile.etasclass`;

the order of parametrs is: (`mu`, `k0`, `c`, `p`, `a`, `gamma`, `d`, `q`).

A `plot` method is defined for `profile.etasclass` objects. A number of grid points `nprofile` of 7 (the default) usually is enough to have a good interpolation of the profile likelihood.

Value

Plots a profile likelihood (in the scale -2log(LR)), and plots horizontal lines corresponding to the percentiles of a 1df chi-square variable of levels `prob`; the approximate confidence intervals corresponding to the levels `prob` are printed. Returns a list:

spline.profile	The spline interpolation of the profile likelihood.
conf	The approximate confidence intervals corresponding to the levels `prob`.
prob	The `prob` values used.

Note

A odd number of grid points `nprofile` is adviced, so that the central point is the unconstrained ML estimate for the profiled parameter, and the interpolation of the profile likelihood will have a better quality.

Author(s)

Marcello Chiodi, Giada Adelfio

See Also

`eqcat`, `etasclass`, `profile.etasclass`

Examples

```
## Not run:
## see example in profile.etasclass

## End(Not run)
```

---

print.etasclass — *Print method for etasclass objects*

Description

Print method for an object of class `etasclass`.

Gives some information on the execution and gives estimates of the ETAS parameters.

Usage

```
## S3 method for class 'etasclass'
print(x,...)
```

Arguments

`x`	an `etaclass` object.
`...`	other arguments.

Details

Print brief information about an object of class `etasclass`. More output is obtained with `summary`.

Value

Displays parameters estimates and information on the execution of the `etasclass` estimation process. Displays also the exact call of the function that generated `etasclass`

Author(s)

Marcello Chiodi, Giada Adelfio

See Also

`etasclass`,`eqcat`, `profile.etasclass` `profile.etasclass` *profile method for etasclass objects (ETAS model) (To be checked)*

Description

profile method for etasclass objects (ETAS model).

Usage

```
## S3 method for class 'etasclass'
profile(fitted,iprofile =1,
 nprofile =7,
 kprofile =3,
 profile.approx =FALSE,...)
```

Arguments

`fitted`	An object of the class `etasclass`
.	
`iprofile`	An integer in the range 1-7. Profile likelihood will be computed with respect to the parameter of index `iprofile`. The order of parametrs is: `mu`, `k0`, `c`, `p`, `a`, `gamma`, `d`, `q`. In this version 2.0.0 `profile` only for etas parameters, not for covariates parameters. It will added future versions
`nprofile`	Number of values of `params[iprofile]` for which profile likelihood must be computed. Default value= 7.
`kprofile`	Maximum absolute standardized value for `params[iprofile]`. Profile likelihood will be computed in the standardized range [-`kprofile`, `kprofile`]. Default value= 3.
`profile.approx`	if `TRUE`, then a conditional-likelihood approach is used as a first value for each maximization step in profile likelihood computation. Default value= `FALSE`.
`...`	other arguments.

Details

Profile likelihood for the `iprofile`-th parameter of the ETAS model for earthquake seismicity, estimated with `etasclass`; the order of parameters is: `mu`,`k0`,`c`,`p`,`gamma`,`d`,`q` and `betacov`.

A `plot` method is defined for `profile.etasclass` objects. A number of grid points `nprofile` of 7 (the default) usually is enough to have a good interpolation of the profile likelihood. The profile is computed using the final estimation of the background seismicity used to obtain the object `etas` of class `etasclass` and regardless to the method used. The computing time (for each of the `nprofile` values) is generally less than a single execution of `etasclass` without clustering, because only ML estimation is performed. Parameters not estimated in `etas` (with `params.ind[i]=FALSE`) will remain fixed do the value `params.fix[i]`.

To obtain profiles for different parameters, run `profile.etasclass` with different values of `iprofile`.

Value

Returns a list:

`params.vec` vector of values of the parameter `iprofile` used to evaluate the profile likelihood.

`logl.vec` vector of likelihoods corresponding to the values of `params.vec`

`plot` method is defined to represent profile likelihood (in scale -2log(LR)), using a spline interpolation through grid points, with superimposition of approximate confidence intervals.

Note

A odd number of grid points `nprofile` is adviced, so that the central point is the unconstrained ML estimate for the profiled parameter, and the interpolation of the profile likelihood will have a better quality.

Author(s)

Marcello Chiodi, Giada Adelfio

See Also

`eqcat`, `etasclass`, `plot.profile.etasclass`

Examples

```
## Not run: ##
data("italycatalog")
# load a sample catalog of italian seismicity

   etas.flp<-etasclass(italycatalog,
   magn.threshold = 3.0,  magn.threshold.back = 3.5,
   k0 = 0.005, c = 0.005, p = 1.01, gamma = 0.6, q = 1.52, d = 1.1,
   params.ind = c(TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE),
   formula1 = "time ~  magnitude- 1", declustering = TRUE,
   thinning = FALSE, flp = TRUE, ndeclust = 15,
   onlytime = FALSE, is.backconstant = FALSE,
   description = "etas flp",sectoday = TRUE,
   usenlm = TRUE, epsmax = 0.001)
    # execution of etasclass for events with minimum magnitude of 3.0.
    # The events with magnitude at least 3.5 are used to build a first approximation
    # for the background intensity function
    # (magn.threshold.back=3.5)

## compute profile likelihood for the first parameter (mu)
 system.time( prof.flp <- profile(etas.flp, nprofile = 7, iprofile = 1))
 plot(prof.flp)
#### output:
Asymptotic  confidence  intervals:
  Coverage Lower Upper
1     0.90 0.335 0.376
```

```
2     0.95 0.334 0.378
3     0.99 0.329 0.385


## End(Not run)
```

simpson.coeff *Computes Simpson integration rule coefficients*

Description

Computes Simpson integration rule coefficients.

Usage

```
simpson.coeff(n)
simpson.kD(n,k=2)
```

Arguments

n	number of points of the simpson formula a single dimension
k	number of dimensions

Details

`simpson.coeff` computes the coefficients of the standard Simpson rule (for unit spaced points), according to the sequence (1+4+2+4+...+2+4+1)/3 for each dimension. `simpson.kD` expand the formula over a grid of n^k points in k dimensions.

Value

a vector of `n` coefficients (for `simpson.coeff`), a k-dimensions array with a total of n^k elements for `simpson.kD`.

Author(s)

Marcello Chiodi

summary.etasclass — *Summary method for etasclass objects*

Description

This is the main method to summarize the output of an object of class `etasclass`.

It gives some information on the execution and gives estimates of the ETAS parameters together with the standard errors.

More detailed output is avaliable by inspecting `str(etasclass.object)`, and printing single objects.

Usage

```
## S3 method for class 'etasclass'
summary(object,full=FALSE,...)
```

Arguments

`object`	an `etaclass` object to pass to `summary`.
`full`	logical. New in version 2.2. If `TRUE` the full original call will be displayed together with some more output.
`...`	other arguments.

Details

Displays summary information about an object of class `etasclass`.

Value

Displays AIC values, parameters estimates and their standard errors, together with some information on the execution of the `etasclass` estimation process. Displays also the exact call of the function that generated `etasclass`

Author(s)

Marcello Chiodi, Giada Adelfio

See Also

`etasclass`,`eqcat`, `profile.etasclass`

Examples

```
## Not run:
# summary method for the etasclass object esecov1 and esecov5

(see examples in \code{\link{etasclass}})

## only with one covariate, the magnitude, classical ETAS model
>   summary(esecov1)
Call:

etasclass(cat.orig = catalog.withcov, magn.threshold = 2.5, magn.threshold.back = 3.9,
    mu = 0.3, k0 = 0.02, c = 0.015, p = 0.99, gamma = 0, d = 1,
    q = 1.5, params.ind = c(TRUE, TRUE, TRUE, TRUE, FALSE, TRUE,
        TRUE), formula1 = "time ~  magnitude- 1", declustering = TRUE,
    thinning = FALSE, flp = TRUE, ndeclust = 15, onlytime = FALSE,
    is.backconstant = FALSE, sectoday = FALSE, usenlm = TRUE,
    compsqm = TRUE, epsmax = 1e-04, iterlim = 100, ntheta = 36)



Execution started:                   2020-05-03 00:24:08
Elapsed time of execution (hours)    0.2294818
Number of observations              2226
Magnitude threshold                 2.5
declustering                         TRUE
Number of declustering iterations    6
Kind of declustering                 weighting
flp                                  TRUE
sequence of AIC values for each iteration
44887.75 43348.46 43250.77 43249.77 43249.27 43249.19

final AIC value
44887.75 43348.46 43250.77 43249.77 43249.27 43249.19

-------------------------------------------------------

formula for covariates of the triggered components:
time ~ magnitude - 1
<environment: 0x55968d6fd660>
ETAS Parameters:
                 Estimates       std.err.
mu                0.667509       0.022620
k0                0.022393       0.005781
c                 0.014769       0.002708
p                 1.110059       0.015709
gamma             0.000000       0.000000
d                 1.905461       0.260360
q                 1.947223       0.077627
magnitude        0.740109       0.092558
-------------------------------------------------------
```

```
#### using covariates
>   summary(esecov5)
Call:

etasclass(cat.orig = catalog.withcov, magn.threshold = 2.5, magn.threshold.back = 3.9,
    mu = 0.3, k0 = 0.02, c = 0.015, p = 0.99, gamma = 0, d = 1,
    q = 1.5, params.ind = c(TRUE, TRUE, TRUE, TRUE, FALSE, TRUE,
      TRUE), formula1 = "time ~ z + magnitude +nstaloc_rev +min_distance_rev+distmin- 1",
    declustering = TRUE, thinning = FALSE, flp = TRUE, ndeclust = 15,
    onlytime = FALSE, is.backconstant = FALSE, sectoday = FALSE,
    usenlm = TRUE, compsqm = TRUE, epsmax = 1e-04, iterlim = 100,
    ntheta = 36)




Execution started:                  2020-05-03 12:22:31
Elapsed time of execution (hours)   0.4827933
Number of observations             2226
Magnitude threshold                2.5
declustering                        TRUE
Number of declustering iterations   3
Kind of declustering                weighting
flp                                 TRUE
sequence of AIC values for each iteration
44693.04 42884.07 42706.16

final AIC value
44693.04 42884.07 42706.16

-------------------------------------------------------

formula for covariates of the triggered components:
time ~ z + magnitude + nstaloc_rev + min_distance_rev + distmin -
    1
<environment: 0x55968d5ed118>
ETAS Parameters:
                       Estimates       std.err.
mu                      0.705351       0.022740
k0                      0.073070       0.021194
c                       0.019396       0.003435
p                       1.154186       0.016874
gamma                   0.000000       0.000000
d                       1.942929       0.272434
q                       2.004915       0.084784
z                      -0.041256       0.005779
magnitude              1.157698       0.085360
nstaloc_rev            -0.009010       0.001817
min_distance_rev       -0.011020       0.002804
distmin                -1.826717       0.298649
-------------------------------------------------------
```

```
## End(Not run)
```

`time2date` *Date time conversion tools*

Description

Date time conversion tools, useful in connection with package etasFLP for earthquake description. Base date is Jan. 1st 1900.

Usage

```
time2date(t)

timecharunique2seq(timestring)
```

Arguments

`t`	seconds elapsed from 1900-1-1.
`timestring`	A time string.

Details

`time2date` converts sequential time in seconds into character string; `timecharunique2seq` converts character times of catalogs into sequential time (seconds elapsed from the base date): the input is a single string.

Value

`time2date` returns a character string; `timecharunique2seq` returns a list:

`char`	the input string.
`sec`	seconds elapsed from the base date.
`day`	days elapsed from the base date.

Author(s)

Marcello Chiodi

Examples

```
## Not run:
tchar="1960-11-06 11:09:35.000"
tsec =timecharunique2seq(tchar)[["sec"]]
time2date(tsec)

## End(Not run)
```

timeupdate.etasclass *timeupdate.etasclass*

Description

New in version 2.2. A time updating of an etasclass objects: a very experimental version that can be used only on etasclass objects obtained from etasflp versions 2.2 or newer.

Usage

```
timeupdate.etasclass(object, params.estimation = FALSE, ...)
```

Arguments

`object` an etasclass object obtained from `etasFLP` versions 2.2 or newer that will be updated for a new time window with new events.

`params.estimation` logical. if `TRUE` parameters will be estimated again even if quickly, with few optimizations steps. Elsewhere ETAS estimates of input object will be mantained

`...` optional arguments that will override the corresponding arguments in `object`, possibly including a new catalog input or a new `tmax`

Details

It is a beta version. A new ETAS model is fitted to a previous object of class etasclass with a new catalog which must be a catalog which extends the previous one on a wider time window, that is a catalog with new observations.

As a default a new quick execution is made, with one quick iteration for parameter updating and an iteration for background density estimation.

Value

a new `etasclass` object

See Also

`update.etasclass`

`update.etasclass` *update.etasclass*

Description

New in version 2.2. A method update for etasclass objects: a very experimental version that can be used only on etasclass objects obtained from etasflp versions 2.2 or newer.

Usage

```
## S3 method for class 'etasclass'
update(object, ...)
```

Arguments

`object`	an etasclass object obtained from etasflp versions 2.2 or newer that will be updated.
`...`	optional arguments that will override the corresponding arguments in `object`

Details

It is a beta version. The catalog must be the same, and options in "..." must leave unchanged the number of observations used for estimation. Arguments given in "..." will override arguments already present in `object`. Not all arguments are suitable for updating: among them `formula` and `params.ind` should not be included in "..." list (to update such parameters it is better to assign them to a variable and then pass the variable name) . A new etasclass execution will start, using as arguments values of input `object`, eventually integrated with the list in "...". Tipically a first execution can be given with low values of iterlim, ndeclust, ntheta and high values of epsmax (e.g. iterlim=5, ndeclust=1, ntheta=24, epsmax=0.01), to obtain good starting values for parameters and for weights. Then an update can be run with better values such as iterlim=50, ndeclust=10, ntheta=60, epsmax=0.0001.

Value

un updated `etasclass` object

See Also

`timeupdate.etasclass`

xy.grid *Creates a 2-d grid*

Description

Creates a 2-d grid.

Usage

```
xy.grid(rangex, rangey, nx, ny = nx)
```

Arguments

rangex	A length 2 numeric vector: the range of the x-variable.
rangey	A length 2 numeric vector: the range of the y-variable.
nx	The number of points of the grid in the x-direction.
ny	The number of points of the grid in the y-direction.

Value

A grid of the coordinates of nx*ny points on the x-y plane, expanded in a matrix of nx*ny rows and 2 columns: a row gives the (x,y) coordinates of a point.

Examples

```
xy.grid(c(3,7),c(11,17),nx=5,ny=4)
```

Index