Package ‘stopp’

February 24, 2023

Title Spatio-Temporal Point Pattern Methods, Model Fitting, Diagnostics, Simulation, Local Tests

Version 0.1.0

Date 2023-02-24

Type Package

Description Toolbox for different kinds of spatio-temporal analyses to be performed on observed point patterns, following the growing stream of literature on point process theory. This R package implements functions to perform different kinds of analyses on point processes, proposed in the papers (Siino, Adelfio, and Mateu 2018<doi:10.1007/s00477-018-1579-0>; Siino et al. 2018<doi:10.1002/env.2463>; Adelfio et al. 2020<doi:10.1007/s00477-019-01748-1>; D’Angelo, Adelfio, and Mateu 2021<doi:10.1016/j.spasta.2021.100534>; D’Angelo, Adelfio, and Mateu 2022<doi:10.1007/s00362-022-01338-4>; D’Angelo, Adelfio, and Mateu 2023<doi:10.1016/j.csda.2022.107679>). The main topics include modeling, statistical inference, and simulation issues on spatio-temporal point processes on Euclidean space and linear networks.

License GPL (>= 2)

Encoding UTF-8

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Depends R (>= 3.5.0)

Imports stats, graphics, KernSmooth, MASS, fields, gmp, latex2exp, optimx, plot3D, sparr, spatstat.explore, spatstat.geom, spatstat.linnet, spatstat.random, splancs, spatstat.model, stlnpp, stpp, etasFLP

LazyData true

Author Nicoletta D'Angelo [aut, cre] (<https://orcid.org/0000-0002-8878-5986>), Giada Adelfio [aut] (<https://orcid.org/0000-0002-3194-4296>)

Maintainer Nicoletta D'Angelo <nicoletta.dangelo@unipa.it>

RoxygenNote 7.2.3

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2023-02-24 19:30:02 UTC

R topics documented:

`stopp-package` *Spatio-Temporal Point Pattern Methods, Model Fitting, Diagnostics, Simulation, Local Tests*

Description

Toolbox for different kinds of spatio-temporal analyses to be performed on observed point patterns, following the growing stream of literature on point process theory. This R package implements functions to perform different kinds of analyses on point processes, proposed in the papers: Siino, Adelfio, and Mateu (2018), Siino et al. (2018), Adelfio et al. (2020), D'Angelo, Adelfio, and Mateu (2021), D'Angelo, Adelfio, and Mateu (2022), and D'Angelo, Adelfio, and Mateu (2023). The main topics include modeling, statistical inference, and simulation issues on spatio-temporal point processes on Euclidean space and linear networks.

Author(s)

Nicoletta D'Angelo [aut,cre] nicoletta.dangelo@unipa.it, Giada Adelfio [aut]

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022) Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Siino, M., Adelfio, G., and Mateu, J. (2018). Joint second-order parameter estimation for spatio-temporal log-Gaussian Cox processes. Stochastic environmental research and risk assessment, 32(12), 3525-3539.

Siino, M., Rodríguez-Cortés, F. J., Mateu, J. ,and Adelfio, G. (2018). Testing for local structure in spatiotemporal point pattern data. Environmetrics, 29(5-6), e2463.

`AIC.stppm` *Akaike's Information Criterion for spatio-temporal Poisson Point Process Model*

Description

This function returns the $AIC = 2k - 2\log(\hat{L})$ of a point process model fitted through the function stppm applied to an observed spatio-temporal point pattern of class `stp`.

As the model returned by stppm is fitted through a quadrature scheme, the log-likelihood is computed trough the quantity: $-\log L(\hat{\theta}; \boldsymbol{x}) = \frac{D}{2} + \sum_{j=1}^{n} I_j \log w_j + n(\boldsymbol{x})$

Usage

```
## S3 method for class 'stppm'
AIC(object, ..., k = 2)
```

Arguments

`object`	An abject of class `stppm`
`...`	additional unused argument
`k`	numeric, the penalty per parameter to be used; the default is k = 2

Value

AIC value

Author(s)

Nicoletta D'Angelo

References

Baddeley, A., Rubak, E., and Turner, R. (2015). Spatial point patterns: methodology and applications with R. CRC press.

See Also

stppm, BIC.stppm

Examples

```
## Homogeneous
set.seed(2)
ph <- rstpp(lambda = 200, nsim = 1, seed = 2, verbose = TRUE)
hom1 <- stppm(ph, formula = ~ 1)

## Inhomogeneous
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)

AIC(hom1)
AIC(inh1)
```

---

as.stlp — *Convert a `stlpp` object to a `stlp` object*

Description

This function converts an object of class `stlpp` to an object of class `stlp`

Usage

```
as.stlp(x)
```

Arguments

`x`	An object of class `stlpp`

Value

An object of class `stlp`

Author(s)

Nicoletta D'Angelo

References

Moradi M, Cronie O, Mateu J (2020). stlnpp: Spatio-temporal analysis of point patterns on linear networks.

See Also

stp, summary.stlp, print.stlp, as.stlpp, plot.stlp

Examples

```
set.seed(12345)
stlpp1 <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)

class(as.stlp(stlpp1))
```

---

as.stlpp *Convert a* `stlp` *object to a* `stlpp` *object*

Description

This function converts an object of class `stlp` to an object of class `stlpp`

Usage

```
as.stlpp(x)
```

Arguments

`x`	An object of class `stlp`

Value

An object of class `stlpp`

Author(s)

Nicoletta D'Angelo

References

Moradi M, Cronie O, Mateu J (2020). stlnpp: Spatio-temporal analysis of point patterns on linear networks.

See Also

stp, summary.stlp, print.stlp, as.stlp, plot.stlp

Examples

```
set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)
df0 <- cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], rpp1$xyt[, 3])
stlp1 <- stp(df0, chicagonet)

class(as.stlpp(stlp1))
```

---

as.stp — *Convert a* `stpp` *object to a* `stp` *object*

Description

This function converts an object of class `stpp` to an object of class `stp`

Usage

```
as.stp(x)
```

Arguments

`x`	An object of class `stpp`

Value

An object of class `stp`

Author(s)

Nicoletta D'Angelo

References

Gabriel, E., Rowlingson, B. S., & Diggle, P. J. (2013). stpp: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns. Journal of Statistical Software, 53(2), 1–29. https://doi.org/10.18637/jss.v053.i02

See Also

stp, print.stp, summary.stp, plot.stp, as.stpp

Examples

```
set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)

class(as.stp(rpp1$xyt))
```

---

as.stpp *Convert a* `stp` *object to a* `stpp` *object*

Description

This function converts an object of class `stp` to an object of class `stpp`

Usage

```
as.stpp(x)
```

Arguments

`x`	An object of class `stp`

Value

An object of class `stpp`

Author(s)

Nicoletta D'Angelo

References

Gabriel, E., Rowlingson, B. S., & Diggle, P. J. (2013). stpp: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns. Journal of Statistical Software, 53(2), 1–29. https://doi.org/10.18637/jss.v053.i02

See Also

stp, print.stp, summary.stp, plot.stp, as.stp

Examples

```
set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)
df0 <- cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], rpp1$xyt[, 3])
stp1 <- stp(df0)

class(as.stpp(stp1))
```

`BIC.stppm` *Bayesian Information Criterion for spatio-temporal Poisson Point Process Model*

Description

This function returns the $BIC = k \log n - 2\log(\hat{L})$ of a point process model fitted through the function stppm applied to an observed spatio-temporal point pattern of class `stp`.

As the model returned by stppm is fitted through a quadrature scheme, the log-likelihood is computed trough the quantity: $-\log L(\hat{\theta}; \boldsymbol{x}) = \frac{D}{2} + \sum_{j=1}^{n} I_j \log w_j + n(\boldsymbol{x})$

Usage

```
## S3 method for class 'stppm'
BIC(object, ...)
```

Arguments

`object`	An abject of class `stppm`
`...`	additional unused argument

Value

BIC value

Author(s)

Nicoletta D'Angelo

References

Baddeley, A., Rubak, E., and Turner, R. (2015). Spatial point patterns: methodology and applications with R. CRC press.

See Also

stppm, AIC.stppm

Examples

```
## Homogeneous
set.seed(2)
ph <- rstpp(lambda = 200, nsim = 1, seed = 2, verbose = TRUE)
hom1 <- stppm(ph, formula = ~ 1)

## Inhomogeneous
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)


BIC(hom1)
BIC(inh1)
```

---

chicagonet — *Roads of Chicago (Illinois, USA) close to the University of Chicago*

Description

A linear network of class `linnet` of the roads of Chicago (Illinois, USA) close to the University of Chicago. It represents the linear network of the Chicago dataset published and analysed in Ang, Baddeley and Nair (2012). Note that the network adjacency matrix is stored as a sparse matrix.

Usage

```
data(chicagonet)
```

Format

A linear network of class `linnet`

Author(s)

Nicoletta D'Angelo

References

Ang, Q.W., Baddeley, A. and Nair, G. (2012) Geometrically corrected second-order analysis of events on a linear network, with applications to ecology and criminology. Scandinavian Journal of Statistics 39, 591–617.

Examples

```
data(chicagonet)
```

`coef.locstppm` *Extract the fitted coefficients of a local spatio-temporal Poisson process model*

Description

Extract the fitted coefficients of a local spatio-temporal Poisson process model

Usage

```
## S3 method for class 'locstppm'
coef(object, ...)
```

Arguments

`object`	An object of class `locstppm`
`...`	additional unused argument

Value

A list containing the global and local fitted coefficients

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

locstppm, print.locstppm, summary.locstppm, plot.locstppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)

coef(inh00_local)
coef(inh01_local)
```

---

coef.stppm — *Extract the fitted coefficients of a spatio-temporal Poisson process model*

Description

Extract the fitted coefficients of a spatio-temporal Poisson process model

Usage

```
## S3 method for class 'stppm'
coef(object, ...)
```

Arguments

`object`	An object of class `stppm`
`...`	additional unused argument

Value

A list containing the fitted coefficients

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

stppm, print.stppm, summary.stppm, plot.stppm

Examples

```
## Homogeneous
set.seed(2)
ph <- rstpp(lambda = 200, nsim = 1, seed = 2, verbose = TRUE)
hom1 <- stppm(ph, formula = ~ 1)

coef(hom1)

## Inhomogeneous
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)

coef(inh1)
```

`globaldiag` *Global diagnostics of a spatio-temporal point process first-order intensity*

Description

This function performs global diagnostics of a model fitted for the first-order intensity of a spatio-temporal point pattern, by returning the plots of the inhomogeneous K-function weighted by the provided intensity to diagnose, its theoretical value, and their difference.

If applied to a `stp` object, it resorts to the spatio-temporal inhomogeneous K-function (Gabriel and Diggle, 2009) documented by the function STIKhat of the `stpp` package (Gabriel et al, 2013).

If applied to a `stlp` object, it uses the spatio-temporal inhomogeneous K-function on a linear network (Moradi and Mateu, 2020) documented by the function STLKinhom of the `stlnpp` package (Moradi et al., 2020).

Usage

```
globaldiag(X, intensity, samescale = TRUE)
```

Arguments

`X`	A `stp` object
`intensity`	A vector of intensity values, of the same length as the number of point in `X`
`samescale`	Logical value. It indicates whether to plot the observed and the theoretical K-function in the same or different scale. Default to `TRUE`.

Value

Character returning the sum of squared differences. The function plots three panels: the observed K-function, as returned by STLKinhom; the theoretical one; their difference. The function also prints the sum of squared differences between the observed and theoretical K-function on the console.

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022) Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

Gabriel, E., and Diggle, P. J. (2009). Second-order analysis of inhomogeneous spatio-temporal point process data. Statistica Neerlandica, 63(1), 43-51.

Gabriel, E., Rowlingson, B. S., & Diggle, P. J. (2013). stpp: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns. Journal of Statistical Software, 53(2), 1–29. https://doi.org/10.18637/jss.v053.i02

Moradi M, Cronie O, and Mateu J (2020). stlnpp: Spatio-temporal analysis of point patterns on linear networks.

Moradi, M. M., and Mateu, J. (2020). First-and second-order characteristics of spatio-temporal point processes on linear networks. Journal of Computational and Graphical Statistics, 29(3), 432-443.

See Also

infl, plot.localdiag, print.localdiag, summary.localdiag, localdiag,

Examples

```
#load data
set.seed(12345)
id <- sample(1:nrow(etasFLP::catalog.withcov), 200)
cat <- etasFLP::catalog.withcov[id, ]
stp1 <- stp(cat[, 5:3])

#fit two competitor models
# and extract the fitted spatio-temporal intensity

lETAS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, 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)$l

lPOIS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, gamma = 0, d = 1,
q = 1.5, params.ind = c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
                        FALSE), 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)$l

globaldiag(stp1, lETAS)


globaldiag(stp1, lPOIS)


globaldiag(stp1, lPOIS, samescale = FALSE)

## Network case

set.seed(12345)
stlp1 <- rETASlp(cat = NULL, params = c(0.078915 / 2, 0.003696,  0.013362,  1.2,
                                             0.424466,  1.164793),
                      betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
                      xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300,
                      iprint = TRUE, covdiag = FALSE, covsim = FALSE, L = chicagonet)

globaldiag(stlp1, intensity = density(as.stlpp(stlp1),
at = "points"))

globaldiag(stlp1, intensity = density(as.stlpp(stlp1),
                                      at = "points"), samescale = FALSE)
```

greececatalog *Catalog of Greek earthquakes*

Description

A dataset in `stp` format containing the catalog of Greek earthquakes of magnitude at least 4.0 from year 2005 to year 2014, analysed by mean of local log-Gaussian Cox processes in D'Angelo et al. (2022) and D'Angelo et al. (2023).

Data come from the Hellenic Unified Seismic Network (H.U.S.N.).

The same data have been analysed in Siino et al. (2017) by hybrids of Gibbs models, and more recently by Gabriel et al. (2022).

Usage

```
data(greececatalog)
```

Format

A `stp` object for a spatio-temoral point pattern with 1111 points

Details

The variables are as follows:

- x. longitude, ranging from 20.02 to 27.98
- y. latitude, ranging from 33.75 to 40.45
- t. time, rangin from 38354, 42000

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Siino, M., D'Alessandro, A., and Adelfio, G. (2022). Local spatial log-Gaussian Cox processes for seismic data. AStA Advances in Statistical Analysis, 1-39.

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Gabriel, E., Rodriguez-Cortes, F., Coville, J., Mateu, J., and Chadoeuf, J. (2022). Mapping the intensity function of a non-stationary point process in unobserved areas. Stochastic Environmental Research and Risk Assessment, 1-17.

Siino, M., Adelfio, G., Mateu, J., Chiodi, M., and D'alessandro, A. (2017). Spatial pattern analysis using hybrid models: an application to the Hellenic seismicity. Stochastic Environmental Research and Risk Assessment, 31(7), 1633-1648.

Examples

```
data(greececatalog)

# Spatio-temporal point pattern
# 1111 points
# Enclosing window: rectangle = [20.02, 27.98] x [33.75, 40.45] units
# Time period: [38353.906, 42000.338]

plot(greececatalog, tcum = TRUE)
```

`infl` *Display outlying LISTA functions*

Description

This function works on the objects of class `localdiag`, as returned by localdiag, plotting the identified 'outlying' LISTA functions. These correspond to the influential points in the fitting of the model provided by localdiag

Usage

```
infl(x, id = NULL)
```

Arguments

`x`	An object of class `localdiag`
`id`	The id of the LISTA to display. Default is set to the ids identified and stored in the `localdiag` object

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022) Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

See Also

localdiag, plot.localdiag, print.localdiag, summary.localdiag

Examples

```
#load data
set.seed(12345)
id <- sample(1:nrow(etasFLP::catalog.withcov), 200)
cat <- etasFLP::catalog.withcov[id, ]
stp1 <- stp(cat[, 5:3])

#fit two competitor models
# and extract the fitted spatio-temporal intensity

lETAS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, 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)$l

lPOIS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, gamma = 0, d = 1,
q = 1.5, params.ind = c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
                        FALSE), 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)$l

# let's identify the outlying points at a .9 percentile

resETAS <- localdiag(stp1, lETAS, p = .9)
resPOIS <- localdiag(stp1, lPOIS, p = .9)

# let's look at the outlying LISTA functions selected by localdiag() ...

infl(resETAS)
infl(resPOIS)

#... and at a some specific LISTA

infl(resETAS, id = c(75))
infl(resETAS, id = c(2, 4, 6))
infl(resPOIS, id = 1:6)
```

is.stlp — *Test whether an object is a spatio-temporal point pattern on a linear network*

Description

Test whether an object is a spatio-temporal point pattern on a linear network

Usage

```
is.stlp(x)
```

Arguments

`x`	Any object

Details

This function tests whether the object `x` is a spatio-temporal point pattern object of class `stlp`. See stp for details of this class

Value

`TRUE` is `x` is a spatio-temporal point pattern on a linear network, otherwise `FALSE`

Author(s)

Nicoletta D'Angelo

See Also

stp, summary.stlp, print.stlp, plot.stlp

Examples

```
set.seed(12345)
stlpp1 <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)
is.stlp(stlpp1)

df0 <- cbind(stlpp1$data$x, stlpp1$data$y, stlpp1$data$t)
L0 <- stlpp1$domain
stlp1 <- stp(df0, L0)

is.stlp(stlp1)
```

`is.stp` *Test whether an object is a spatio-temporal point pattern*

Description

Test whether an object is a spatio-temporal point pattern

Usage

```
is.stp(x)
```

Arguments

x	Any object

Details

This function tests whether the object `x` is a spatio-temporal point pattern object of class `stp`. See stp for details of this class

Value

`TRUE` is `x` is a spatio-temporal point pattern, otherwise `FALSE`

Author(s)

Nicoletta D'Angelo

See Also

stp, summary.stp, print.stp, plot.stp

Examples

```
set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)
is.stp(rpp1)

df0 <- cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], rpp1$xyt[, 3])
stp1 <- stp(df0)

is.stp(stp1)
```

localdiag — *Local diagnostics of spatio-temporal point process models*

Description

This function performs local diagnostics of a model fitted for the first-order intensity of a spatio-temporal point pattern, by means of the local spatio-temporal inhomogeneous K-function (Adelfio et al, 2020) documented by the function KLISTAhat of the stpp package (Gabriel et al, 2013).

The function can also perform local diagnostics of a model fitted for the first-order intensity of an spatio-temporal point pattern on a linear network, by means of the local spatio-temporal inhomogeneous K-function on linear networks (D'Angelo et al, 2021) documented by the function localSTLKinhom.

In both cases, it returns the points identified as outlying following the diagnostics procedure on individual points of an observed point pattern, as introduced in Adelfio et al. (2020), and applied in D'Angelo et al. (2022) for the linear network case

See the section 'Details'.

The points resulting from the local diagnostic procedure provided by this function can be inspected via the plot, print, summary, and infl functions.

Usage

```
localdiag(X, intensity, p = 0.95)
```

Arguments

`X`	Either a `stp` or a `stlp` object
`intensity`	A vector of intensity values, of the same length as the number of point in `X`
`p`	The percentile to consider as threshold for the outlying points. Default to 0.95.

Details

Adelfio et al. (2020) derived the expectation of the local inhomogeneous spatio-temporal K-function, under the Poisson case: $\mathbb{E}[\hat{K}^i(r,h)] = \pi r^2 h$.

Moreover, they found that when the local estimator is weighted by the true intensity function, its expectation, $\mathbb{E}[\hat{K}^i_I(r,h)]$, is the same as the expectation of $\hat{K}^i(r,h)$.

These results motivate the usage of such local estimator $\hat{K}^i_I(r,h)$ as a diagnostic tool for general spatio-temporal point processes for assessing the goodness-of-fit of spatio-temporal point processes of any generic first-order intensity function λ.

Indeed, if the estimated intensity function used for weighting in our proposed LISTA functions is the true one, then the LISTA functions should behave as the corresponding ones of a homogeneous Poisson process, resulting in small discrepancies between the two.

Therefore, this function computes such discrepancies by means of the χ_i^2 values, obtained following the expression

$$\chi_i^2 = \int_L \int_T \left(\frac{\left(\hat{K}_I^i(r,h) - \mathbb{E}[\hat{K}^i(r,h)]\right)^2}{\mathbb{E}[\hat{K}^i(r,h)]} \right) \mathrm{d}h\mathrm{d}r,$$

one for each point in the point pattern.

Basically, departures of the LISTA functions $\hat{K}_I^i(r,h)$ from the Poisson expected value rh directly suggest the unsuitability of the intensity function $\lambda(\cdot)$ used in the weighting of the LISTA functions for that specific point. This can be referred to as an *outlying point*.

Given that D'Angelo et al. (2022) proved the same results for the network case, that is, $\mathbb{E}[\hat{K}_L^i(r,h)] = rh$ and $\mathbb{E}[\hat{K}_{L,I}^i(r,h)] = \mathbb{E}[\hat{K}_L^i(r,h)]$ when $\hat{K}_{L,I}^i(r,h)$ is weighted by the true intensity function, we implemented the same above-mentioned diagnostics procedure to work on intensity functions fitted on spatio-temporal point patterns occurring on linear networks.

Note that the Euclidean procedure is implemented by the local K-functions of Adelfio et al. (2020), documented in KLISTAhat of the `stpp` package (Gabriel et al, 2013). The network case uses the local K-functions on networks (D'Angelo et al., 2021), documented in localSTLKinhom.

Value

A list object of class `localdiag`, containing

`X` The `stp` object provided as input

`listas` The LISTA functions, in a list object

`ids` The ids of the points identified as outlying

`x2` A vector with the individual contributions to the Chi-squared statistics, normalized

`p` The percentile considered

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022) Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

Gabriel, E., Rowlingson, B. S., and Diggle, P. J. (2013). stpp: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns. Journal of Statistical Software, 53(2), 1–29. https://doi.org/10.18637/jss.v053.i02

See Also

infl, plot.localdiag, print.localdiag, summary.localdiag, globaldiag

Examples

```
#load data
set.seed(12345)
id <- sample(1:nrow(etasFLP::catalog.withcov), 200)
cat <- etasFLP::catalog.withcov[id, ]
stp1 <- stp(cat[, 5:3])

#fit two competitor models
# and extract the fitted spatio-temporal intensity

lETAS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
  mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, 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)$l

lPOIS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
  mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, gamma = 0, d = 1,
  q = 1.5, params.ind = c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
  FALSE), 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)$l

# let's identify the outlying points at a .9 percentile

resETAS <- localdiag(stp1, lETAS, p = .9)
resPOIS <- localdiag(stp1, lPOIS, p = .9)


#### Network case

set.seed(12345)
stlp1 <- rETASlp(cat = NULL, params = c(0.078915 / 2, 0.003696,  0.013362,  1.2,
                                             0.424466,  1.164793),
                      betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
                      xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300,
                      iprint = TRUE, covdiag = FALSE, covsim = FALSE, L = chicagonet)

res <- localdiag(stlp1, intensity = density(as.stlpp(stlp1),
                                                at = "points"), p = .65)
```

`localplot.locstppm` *Plot the coefficients of a fitted local spatio-temporal Poisson process model*

Description

The function plots the local estimates of a fitted local spatio-temporal Poisson process model.

Usage

```
localplot.locstppm(x, par = TRUE)
```

Arguments

`x`	An object of class `locstppm`
`par`	Default to `TRUE`.

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

locstppm, print.locstppm, summary.locstppm, localsummary.locstppm, plot.locstppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)

localplot.locstppm(inh00_local)
localplot.locstppm(inh01_local)
```

localplot.stlgcppm — *Plot the coefficients of a fitted local LGCP model*

Description

The function plots the local estimates. In the case of local covariance parameters, the function displays the local estimates of the chosen covariance function.

Usage

```
localplot.stlgcppm(x, par = TRUE)
```

Arguments

`x`	An object of class `stlgcppm`
`par`	Default to `TRUE`.

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

stlgcppm, print.stlgcppm, summary.stlgcppm, localsummary.stlgcppm, plot.stlgcppm

Examples

```
# Example with complex seismic point pattern
data("greececatalog")

lgcp2 <- stlgcppm(greececatalog, formula = ~ x, first = "local", second = "global")
localplot.stlgcppm(lgcp2)
```

`localSTLginhom` *Local inhomogeneous Spatio-temporal pair correlation functions on a linear network*

Description

The functions `localSTLKinhom` and `localSTLginhom` implement the inhomogeneous LISTA functions proposed in D'Angelo et al. (2022).

The homogeneous K-function and pair correlation functions, in D'Angelo et al. (2021), can be obtained easily with `localSTLKinhom` and `localSTLginhom`, by imputing a lambda vector of constant intensity values, the same for each point.

These local functions are the local counterparts of STLKinhom and STLginhom by Moradi and Mateu (2020), available in the `stlnpp` package (Moradi et al., 2020). Basically, we avoid summing up all the points as in the global statistics counterparts, and we denote the individual contribution to the global statistics with the index i.

Usage

```
localSTLginhom(x, lambda, normalize = FALSE, r = NULL, t = NULL, nxy = 10)
```

Arguments

`x`	A realisation of a spatio-temporal point processes on a linear network in `stlp` format
`lambda`	values of estimated intensity.
`normalize`	normalization factor to be considered.
`r`	values of argument r where pair correlation function will be evaluated. optional.
`t`	values of argument t where pair correlation function will be evaluated. optional.
`nxy`	pixel array dimensions. optional.

Details

The *local spatio-temporal inhomogeneous* K-function for the i-th event $(\boldsymbol{u}_i, t_i)$ on a linear network is

$$\hat{K}^i_{L,I}(r,h) = \frac{1}{|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{I\{d_L(\boldsymbol{u}_i,\boldsymbol{v}) < r, |t_i - s| < h\}}{\hat{\lambda}(\boldsymbol{u}_i,t_i)\hat{\lambda}(\boldsymbol{v},s)M((\boldsymbol{u}_i,t_i),d_L(\boldsymbol{u}_i,\boldsymbol{v}),|t_i-s|)},$$

and the corresponding *local pair correlation function* (pcf)

$$\hat{g}^i_{L,I}(r,h) = \frac{1}{|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{\kappa(d_L(\boldsymbol{u}_i,\boldsymbol{v}) - r)\kappa(|t_i - s| - h)}{\hat{\lambda}(\boldsymbol{u}_i,t_i)\hat{\lambda}(\boldsymbol{v},s)M((\boldsymbol{u}_i,t_i),d_L(\boldsymbol{u}_i,\boldsymbol{v}),|t_i-s|)},$$

with

$$D(X) = \frac{n-1}{|L||T|} \sum_{i=1}^{n} \sum_{i \neq j}$$

normalization factor. This leads to the unbieased estimators $\frac{1}{D(X)}\hat{K}^i_{L,I}(r,h)$ and $\frac{1}{D(X)}\hat{g}^i_{L,I}(r,h)$ (D'Angelo et al., 2022) .

The homogeneous versions (D'Angelo et al., 2021) can be obtained by weighting the second-order summary statistics (either K or pcf) by a constant intensity $\hat{\lambda} = n/(|L||T|)$, giving

$$\hat{K}^i_{L}(r,h) = \frac{1}{\hat{\lambda}^2|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{I\{d_L(\boldsymbol{u}_i,\boldsymbol{v}) < r, |t_i - s| < h\}}{M((\boldsymbol{u}_i,t_i),d_L(\boldsymbol{u}_i,\boldsymbol{v}),|t_i-s|)},$$

and

$$\hat{g}^i_{L}(r,h) = \frac{1}{\hat{\lambda}^2|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{\kappa(d_L(\boldsymbol{u}_i,\boldsymbol{v}) - r)\kappa(|t_i - s| - h)}{M((\boldsymbol{u}_i,t_i),d_L(\boldsymbol{u}_i,\boldsymbol{v}),|t_i-s|)}.$$

Value

A list of objects of class `sumstlpp` (Moradi and Mateu, 2020).

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022). Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

Moradi M, Cronie O, and Mateu J (2020). stlnpp: Spatio-temporal analysis of point patterns on linear networks.

Moradi, M. M., and Mateu, J. (2020). First-and second-order characteristics of spatio-temporal point processes on linear networks. Journal of Computational and Graphical Statistics, 29(3), 432-443.

See Also

localSTLginhom, STLKinhom, STLginhom

Examples

```
set.seed(10)
X <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)
X
lambda <- density(X, at = "points")
x <- as.stlp(X)
g <- localSTLginhom(x, lambda = lambda, normalize = TRUE)
#select an individual point
j = 1
g[[j]]
#plot the lista function
inhom <- list(x = g[[j]]$r, y = g[[j]]$t, z = g[[j]]$ginhom)

fields::image.plot(inhom, main = "ginhom", col = hcl.colors(12, "YlOrRd", rev = FALSE),
xlab = "Spatial distance", ylab = "Temporal distance")
```

`localSTLKinhom` *Local inhomogeneous Spatio-temporal K-functions on a linear network*

Description

The functions `localSTLKinhom` and `localSTLginhom` implement the inhomogeneous LISTA functions proposed in D'Angelo et al. (2022).

The homogeneous K-function and pair correlation functions, in D'Angelo et al. (2021), can be obtained easily with `localSTLKinhom` and `localSTLginhom`, by imputing a lambda vector of constant intensity values, the same for each point.

These local functions are the local counterparts of STLKinhom and STLginhom by Moradi and Mateu (2020), available in the `stlnpp` package (Moradi et al., 2020). Basically, we avoid summing up all the points as in the global statistics counterparts, and we denote the individual contribution to the global statistics with the index i.

Usage

```
localSTLKinhom(
  x,
  lambda = lambda,
```

```
    normalize = FALSE,
    r = NULL,
    t = NULL,
    nxy = 10
)
```

Arguments

`x`	A realisation of a spatio-temporal point processes on a linear network in `stlp` format
`lambda`	values of estimated intensity.
`normalize`	normalization factor to be considered.
`r`	values of argument r where K-function will be evaluated. optional.
`t`	values of argument t where K-function will be evaluated. optional.
`nxy`	pixel array dimensions. optional.

Details

The *local spatio-temporal inhomogeneous* K-function for the i-th event $(\boldsymbol{u}_i, t_i)$ on a linear network is

$$\hat{K}^i_{L,I}(r,h) = \frac{1}{|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{I\{d_L(\boldsymbol{u}_i,\boldsymbol{v}) < r, |t_i - s| < h\}}{\hat{\lambda}(\boldsymbol{u}_i,t_i)\hat{\lambda}(\boldsymbol{v},s)M((\boldsymbol{u}_i,t_i), d_L(\boldsymbol{u}_i,\boldsymbol{v}), |t_i - s|)},$$

and the corresponding *local pair correlation function* (pcf)

$$\hat{g}^i_{L,I}(r,h) = \frac{1}{|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{\kappa(d_L(\boldsymbol{u}_i,\boldsymbol{v}) - r)\kappa(|t_i - s| - h)}{\hat{\lambda}(\boldsymbol{u}_i,t_i)\hat{\lambda}(\boldsymbol{v},s)M((\boldsymbol{u}_i,t_i), d_L(\boldsymbol{u}_i,\boldsymbol{v}), |t_i - s|)},$$

with

$$D(X) = \frac{n-1}{|L||T|} \sum_{i=1}^{n} \sum_{i \neq j}$$

normalization factor. This leads to the unbieased estimators $\frac{1}{D(X)}\hat{K}^i_{L,I}(r,h)$ and $\frac{1}{D(X)}\hat{g}^i_{L,I}(r,h)$ (D'Angelo et al., 2022) .

The homogeneous versions (D'Angelo et al., 2021) can be obtained by weighting the second-order summary statistics (either K or pcf) by a constant intensity $\hat{\lambda} = n/(|L||T|)$, giving

$$\hat{K}^i_{L}(r,h) = \frac{1}{\hat{\lambda}^2|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{I\{d_L(\boldsymbol{u}_i,\boldsymbol{v}) < r, |t_i - s| < h\}}{M((\boldsymbol{u}_i,t_i), d_L(\boldsymbol{u}_i,\boldsymbol{v}), |t_i - s|)},$$

and

$$\hat{g}^i_{L}(r,h) = \frac{1}{\hat{\lambda}^2|L||T|} \sum_{(\boldsymbol{u}_i,t_i)\neq(\boldsymbol{v},s)} \frac{\kappa(d_L(\boldsymbol{u}_i,\boldsymbol{v}) - r)\kappa(|t_i - s| - h)}{M((\boldsymbol{u}_i,t_i), d_L(\boldsymbol{u}_i,\boldsymbol{v}), |t_i - s|)}.$$

Value

A list of objects of class `sumstlpp` (Moradi and Mateu, 2020).

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

D'Angelo, N., Adelfio, G., and Mateu, J. (2022). Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

Moradi M, Cronie O, and Mateu J (2020). stlnpp: Spatio-temporal analysis of point patterns on linear networks.

Moradi, M. M., and Mateu, J. (2020). First-and second-order characteristics of spatio-temporal point processes on linear networks. Journal of Computational and Graphical Statistics, 29(3), 432-443.

See Also

localSTLginhom, STLKinhom, STLginhom

Examples

```
set.seed(10)
X <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)
X
lambda <- density(X, at = "points")
x <- as.stlp(X)
k <- localSTLKinhom(x, lambda = lambda, normalize = TRUE)
#select an individual point
j = 1
k[[j]]
#plot the lista function and compare it with its theoretical value
inhom <- list(x = k[[j]]$r, y = k[[j]]$t, z = k[[j]]$Kinhom)
theo <- list(x = k[[j]]$r, y = k[[j]]$t, z = k[[j]]$Ktheo)
diff <- list(x = k[[j]]$r, y = k[[j]]$t, z = k[[j]]$Kinhom - k[[j]]$Ktheo)
oldpar <- par(no.readonly = TRUE)
par(mfrow = c(1, 3))
fields::image.plot(inhom, main= "Kinhom", col = hcl.colors(12, "YlOrRd", rev = FALSE),
xlab = "Spatial distance", ylab = "Temporal distance")
fields::image.plot(theo, main = "Ktheo", col = hcl.colors(12, "YlOrRd", rev = FALSE),
xlab = "Spatial distance", ylab = "Temporal distance")
fields::image.plot(diff, main = "Kinhom - Ktheo", col = hcl.colors(12, "YlOrRd", rev = FALSE),
xlab = "Spatial distance", ylab = "Temporal distance")
par(oldpar)
```

localsummary.locstppm *Summary plots of the fitted coefficient of a local spatio-temporal Poisson process model*

Description

The function breaks up the contribution of the local estimates to the fitted intensity, by plotting the overall intensity and the density kernel smoothing of some artificial intensities, obtained by imputing the quartiles of the local parameters' distributions.

Usage

```
localsummary.locstppm(
  x,
  scaler = c("silverman", "IQR", "sd", "var"),
  do.points = TRUE,
  print.bw = FALSE,
  zap = 1e-05,
  par = TRUE
)
```

Arguments

`x`	An object of class `locstppm`
`scaler`	Optional. Controls the value for a scalar representation of the spatial scale of the data. Either a character string, `"silverman"` (default), `"IQR"`, `"sd"`, or `"var"`; or positive numeric value(s). See OS.
`do.points`	Add points to plot
`print.bw`	It prints the estimated oversmoothing (OS) bandwidth selector
`zap`	Noise threshold factor (default to 0.00001). A numerical value greater than or equal to 1. If the range of pixel values is less than `zap * .Machine$double.eps`, the image will be treated as constant. This avoids displaying images which should be constant but contain small numerical errors.
`par`	Default to `TRUE`.

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Davies, T.M. and Hazelton, M.L. (2010). Adaptive kernel estimation of spatial relative risk, Statistics in Medicine, 29(23) 2423-2437.

Terrell, G.R. (1990). The maximal smoothing principle in density estimation, Journal of the American Statistical Association, 85, 470-477.

See Also

locstppm, print.locstppm, summary.locstppm, plot.locstppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)

localsummary.locstppm(inh00_local)
localsummary.locstppm(inh01_local)
```

`localsummary.stlgcppm` *Summary plots of the fitted coefficient of a local LGCP*

Description

The function breaks up the contribution of the local estimates to the fitted intensity, by plotting the overall intensity and the density kernel smoothing of some artificial intensities, obtained by imputing the quartiles of the local parameters' distributions.

Usage

```
localsummary.stlgcppm(
  x,
  scaler = c("silverman", "IQR", "sd", "var"),
  do.points = TRUE,
  print.bw = FALSE,
  zap = 1e-05,
  par = TRUE
)
```

Arguments

`x`	An object of class `stlgcppm`
`scaler`	Optional. Controls the value for a scalar representation of the spatial scale of the data. Either a character string, `"silverman"` (default), `"IQR"`, `"sd"`, or `"var"`; or positive numeric value(s). See OS.
`do.points`	Add points to plot
`print.bw`	It prints the estimated oversmoothing (OS) bandwidth selector
`zap`	Noise threshold factor (default to 0.00001). A numerical value greater than or equal to 1. If the range of pixel values is less than `zap * .Machine$double.eps`, the image will be treated as constant. This avoids displaying images which should be constant but contain small numerical errors.
`par`	Default to `TRUE`.

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Davies, T.M. and Hazelton, M.L. (2010). Adaptive kernel estimation of spatial relative risk, Statistics in Medicine, 29(23) 2423-2437.

Terrell, G.R. (1990). The maximal smoothing principle in density estimation, Journal of the American Statistical Association, 85, 470-477.

See Also

stlgcppm, print.stlgcppm, summary.stlgcppm, plot.stlgcppm, localplot.stlgcppm

Examples

```
# Example with complex seismic point pattern
data("greececatalog")

lgcp2 <- stlgcppm(greececatalog, formula = ~ x, first = "local", second = "global")
localsummary.stlgcppm(lgcp2)
```

`localtest` *Test of local structure for spatio-temporal point processes*

Description

This function performs the permutation test of the local structure for spatio-temporal point pattern data, proposed in Siino et al. (2018), as well as for spatio-temporal point pattern data occurring on the same linear network, following D'Angelo et al. (2021).

Usage

```
localtest(X, Z, method = "K", k, alpha = 0.05, verbose = TRUE)
```

Arguments

`X`	Background spatio-temporal point pattern. Usually, the most clustered between `X` and `Z`. Must be either a `stp` or `stlp` object.
`Z`	Other spatio-temporal point pattern. Must also be of the same class as `X`.
`method`	Character string indicating which version of LISTA function to use: either `"K"` or `"g"`. If `"K"`, the local spatio-temporal K-function is used to run the test. If `"g"`, the local spatio-temporal pair correlation function is used.
`k`	Number of permutations
`alpha`	Significance level
`verbose`	If TRUE (default) the progress of the test is printed

Details

The test detects local differences between $\mathbf{x}$ and $\mathbf{z}$ occurring on the same space-time region. This procedure was firstly introduced in Moraga and Montes (2011), extended in the spatio-temporal context by Siino at al, (2018). Finally, test has been made suitable also for spatio-temporal point patterns with spatial domain coinciding with a linear network by D'Angelo et al. (2021).

In general, for each point $(\mathbf{u}, t)$ in the spatio-temporal observed point pattern $\mathbf{x}$, we test

$$\begin{cases} \mathcal{H}_0: & \text{no difference in the second-order local structure of } (\mathbf{u},t) \quad \text{w.r.t} \quad \{\{\mathbf{x} \setminus (\mathbf{u},t)\} \cup \mathbf{z}\} \\ \mathcal{H}_1: & \text{significant difference in the second-order local structure of} (\mathbf{u},t) \quad \text{w.r.t} \quad \{\{\mathbf{x} \setminus (\mathbf{u},t)\} \cup \mathbf{z}\} \end{cases}$$

The sketch of the test is as follows:

1. Set k as the number of permutations

2. For each point $(\mathbf{u}_i, t_i) \in \mathbf{x}, i = 1, \ldots, n$:

2.1. Estimate the LISTA function $\hat{L}^{(i)}(r,h)$ and Compute the local deviation test $T^i = \int_0^{t_0} \int_0^{r_0} \Big(\hat{L}^{(i)}(r,h) - \hat{L}^{-(i)}_{H_0}(r,h) \Big)^2 \mathrm{d}r\mathrm{d}h$, where $\hat{L}^{-(i)}_{H_0}(r,h)$ is the LISTA function for the i^{th} point, averaged over the $j = 1, \ldots, k$ permutations

2.2 Compute a p-value as $p^i = \sum_{j=1}^{k} \mathbf{1}(T_{H_0}^{i,j} \geq T^i)/k$

The test ends providing a vector p of p- values, one for each point in **x**.

If the test is performed for spatio-temporal point patterns as in Siino et al. (2018), that is, on an object of class `stp`, the LISTA functions $\hat{L}^{(i)}$ employed are the local functions of Adelfio et al. (2020), documented in KLISTAhat and LISTAhat of the `stpp` package (Gabriel et al, 2013).

If the function is applied to a `stlp` object, that is, on two spatio-temporal point patterns observed on the same linear network `L`, the LISTA function $\hat{L}^{(i)}$ used are the ones proposed in D'Angelo et al. (2021), documented in localSTLKinhom and localSTLginhom.

Details on the performance of the test are found in Siino et al. (2018) and D'Angelo et al. (2021), for Euclidean and network spaces, respectively.

Value

A list of class `localtest`, containing

`p` A vector of p-values, one for each of the points in `X`

`X` The background spatio-temporal point pattern given in input

`Z` The alternative spatio-temporal point pattern given in input

`alpha` The threshold given in input

`Xsig` A `stp` object storing the resulting significant points

`Xnosig` A `stp` object storing the resulting non-significant points

`id` The ids of the resulting significan points

Author(s)

Nicoletta D'Angelo and Marianna Siino

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

Gabriel, E., Rowlingson, B. S., and Diggle, P. J. (2013). stpp: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns. Journal of Statistical Software, 53(2), 1–29. https://doi.org/10.18637/jss.v053.i02

Moraga, P. and Montes, F. (2011). Detection of spatial disease clusters with lisa functions. Statistics in Medicine, 30(10):1057–1071

Siino, M., Rodríguez-Cortés, F. J., Mateu, J. ,and Adelfio, G. (2018). Testing for local structure in spatiotemporal point pattern data. Environmetrics, 29(5-6), e2463.

See Also

print.localtest, summary.localtest, plot.localtest

Examples

```
# background pattern
set.seed(12345)
X <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(.05, 4),
           nsim = 1, seed = 2, verbose = TRUE)

# alternative pattern
set.seed(12345)
Z <- rstpp(lambda = 25, nsim = 1, seed = 2, verbose = TRUE)


# run the local test
test <- localtest(X, Z, method = "K", k = 9)

## Example on a linear network

# background pattern
set.seed(12345)
X <- rETASlp(cat = NULL, params = c(0.078915 / 1.95, 0.003696,  0.013362,  1.2,
                                    0.424466,  1.164793),
                 betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
                 xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300,
                 iprint = TRUE, covdiag = FALSE, covsim = FALSE, L = chicagonet)

# alternative pattern, on the same linear network
l <- 20 / (spatstat.geom::volume(chicagonet) * (200 - 25))
set.seed(12345)
stlppPOIS <- stlnpp::rpoistlpp(lambda = l, a = 25, b = 200, L = chicagonet)
Z <- as.stlp(stlppPOIS)

# run the local test
test <- localtest(X, Z, method = "K", k = 9)
```

`locstppm` *Fit a local Poisson process model to a spatio-temporal point pattern*

Description

This function fits a Poisson process model to an observed spatio-temporal point pattern stored in a `stp` object, that is, a Poisson model with a set of parameters θ_i for each point i.

We assume that the template model is a Poisson process, with a parametric intensity or rate function $\lambda(\mathbf{u}, t; \theta_i)$ with space and time locations $\mathbf{u} \in W, t \in T$ and parameters $\theta_i \in \Theta$.

Estimation is performed through the fitting of a `glm` using a localized version of the quadrature scheme by Berman and Turner (1992), firstly introduced in the purely spatial context by Baddeley (2017), and in the spatio-temporal framework by D'Angelo et al. (2023).

See the 'Details' section.

Usage

```
locstppm(
  X,
  formula,
  verbose = TRUE,
  mult = 4,
  hs = "global",
  npx0 = 10,
  npt0 = 10
)
```

Arguments

`X`	A `stp` object
`formula`	An object of class `"formula"`: a symbolic description of the model to be fitted. The current version only supports formulas depending on the spatial and temporal coordinates: `x`, `y`, `t`.
`verbose`	Default to `TRUE`
`mult`	The multiplicand of the number of data points, for setting the number of dummy points to generate for the quadrature scheme
`hs`	Character string indicating whether to select fixed or variable bandwidths for the kernel weights to be used in the log-likelihood. In any of those cases, the well-supported rule-of-thumb for choosing the bandwidth of a Gaussian kernel density estimator is employed. If `hs = "global"` (default), a fixed bandwidth is selected. If `hs = "local"`, an individual bandwidth is selected for each point in the pattern `X`.
`npx0`	Number of lags for the space grid period for variable bandwidths kernel
`npt0`	Number of lags for the time period for variable bandwidths kernel

Details

The local log-likelihood associated with the spatio-temporal location $(\mathbf{v}, s)$ is given by

$$\log L((\mathbf{v}, s); \theta) = \sum_i w_{\sigma_s}(\mathbf{u}_i - \mathbf{v}) w_{\sigma_t}(t_i - s) \lambda(\mathbf{u}_i, t_i; \theta) - \int_W \int_T \lambda(\mathbf{u}, t; \theta) w_{\sigma_s}(\mathbf{u}_i - \mathbf{v}) w_{\sigma_t}(t_i - s) \mathrm{d}t \mathrm{d}u$$

where w_{σ_s} and w_{σ_t} are weight functions, and $\sigma_s, \sigma_t > 0$ are the smoothing bandwidths. It is not necessary to assume that w_{σ_s} and w_{σ_t} are probability densities. For simplicity, we shall consider only kernels of fixed bandwidth, even though spatially adaptive kernels could also be used.

Note that if the template model is the homogeneous Poisson process with intensity λ, then the local likelihood estimate $\hat{\lambda}(\mathbf{v}, s)$ reduces to the kernel estimator of the point process intensity with kernel proportional to $w_{\sigma_s} w_{\sigma_t}$.

We now use an approximation similar to $\log L(\theta) \approx \sum_j a_j (y_j \log \lambda(\mathbf{u}_j, t_j; \theta) - \lambda(\mathbf{u}_j, t_j; \theta)) + \sum_j a_j$, but for the local log-likelihood associated with each desired location $(\mathbf{v}, s) \in W \times T$, that is:

$$\log L((\mathbf{v}, s); \theta) \approx \sum_j w_j(\mathbf{v}, s) a_j (y_j \log \lambda(\mathbf{u}_j, t_j; \theta) - \lambda(\mathbf{u}_j, t_j; \theta)) + \sum_j w_j(\mathbf{v}, s) a_j,$$

where $w_j(\mathbf{v}, s) = w_{\sigma_s}(\mathbf{v} - \mathbf{u}_j) w_{\sigma_t}(s - t_j)$.

Basically, for each desired location $(\mathbf{v}, s)$, we replace the vector of quadrature weights a_j by $a_j(\mathbf{v}, s) = w_j(\mathbf{v}, s) a_j$ where $w_j(\mathbf{v}, s) = w_{\sigma_s}(\mathbf{v} - \mathbf{u}_j) w_{\sigma_t}(s - t_j)$, and use the GLM software to fit the Poisson regression.

The local likelihood is defined at any location $(\mathbf{v}, s)$ in continuous space. In practice it is sufficient to consider a grid of points $(\mathbf{v}, s)$.

We refer to D'Angelo et al. (2023) for further discussion on bandwidth selection and on computational costs.

Value

An object of class `locstppm`. A list of

`IntCoefs` The fitted global coefficients

`IntCoefs_local` The fitted local coefficients

`X` The `stp` objectt provided as input

`nX` The number of points in `X`

`I` Vector indicating which points are dummy or data

`y_resp` The response variable of the model fitted to the quadrature scheme

`formula` The formula provided as input

`l` Fitted intensity through the global parameters

`l_local` Fitted intensity through the local parameters

`mod_global` The `glm` object of the model fitted to the quadrature scheme

`newdata` The data used to fit the model, without the dummy points

`time` Time elapsed to fit the model, in minutes

Author(s)

Nicoletta D'Angelo

References

Baddeley, A. (2017). Local composite likelihood for spatial point processes. Spatial Statistics, 22, 261-295.

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

stppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)
```

`plot.localdiag` *Plot of the local diagnostics' result on a spatio-temporal point process model*

Description

This function plots the result of the local diagnostics performed with localdiag on either a `stp` or `stlp` object. It highlights the points of the analysed spatio-temporal point pattern `X` which are identified as outlying by the previously performed local diagnostics; the remaining points of `X` are also represented.

It also shows the underlying linear network, if the local diagnostics has been applied to point patterns occurring on the same linear network, that is, if localdiag has been applied to a `stlp` object.

Usage

```
## S3 method for class 'localdiag'
plot(x, marg = TRUE, col = "grey", col2 = "red", cols = "lightgrey", ...)
```

Arguments

`x`	A `localdiag` object
`marg`	Default to `TRUE`. If `marg = F`, only the spatio-temporal point pattern is plotted
`col`	Color of the outlying points
`col2`	Color of the network (if applicable)
`cols`	Color of the non-outlying points
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022) Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

See Also

infl, print.localdiag, summary.localdiag

Examples

```
#load data
set.seed(12345)
id <- sample(1:nrow(etasFLP::catalog.withcov), 200)
cat <- etasFLP::catalog.withcov[id, ]
stp1 <- stp(cat[, 5:3])

#fit two competitor models
# and extract the fitted spatio-temporal intensity

lETAS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, 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)$l

lPOIS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, gamma = 0, d = 1,
q = 1.5, params.ind = c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
                        FALSE), 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)$l

# let's identify the outlying points at a .9 percentile

resETAS <- localdiag(stp1, lETAS, p = .9)
```

```
resPOIS <- localdiag(stp1, lPOIS, p = .9)

plot(resETAS)
plot(resPOIS)

plot(resETAS, marg = FALSE)
plot(resPOIS, marg = FALSE)
```

`plot.localtest` *Plot of the result of the local permutation test*

Description

This function plots the result of the local permutation test performed with localtest on either a `stp` or `stlp` object. It highlights the points of the background pattern `X`, which exhibit local differences in the second-order structure with respect to `Z`, according to the previously performed test. The remaining points of `X` are also represented.

It also shows the underlying linear network, if the local test has been applied to point patterns occurring on the same linear network, that is, if localtest has been applied to a `stlp` object.

Usage

```
## S3 method for class 'localtest'
plot(x, col = "grey", cols = "lightgrey", col2 = "red", ...)
```

Arguments

`x`	An object of class `localtest`
`col`	Color of the significant points
`cols`	Color of the linear network. If applicable.
`col2`	Color of the non-significant points
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

Siino, M., Rodríguez-Cortés, F. J., Mateu, J. ,and Adelfio, G. (2018). Testing for local structure in spatiotemporal point pattern data. Environmetrics, 29(5-6), e2463.

See Also

localtest, print.localtest, summary.localtest

Examples

```
## Not run:

# Euclidean
# background pattern
set.seed(12345)
X <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(.05, 4),
           nsim = 1, seed = 2, verbose = TRUE)

# alternative pattern
set.seed(12345)
Z <- rstpp(lambda = 25, nsim = 1, seed = 2, verbose = TRUE)

# run the local test
test <- localtest(X, Z, method = "K", k = 9)


plot(test)

# Linear networks

# background pattern
set.seed(12345)
X <- rETASlp(cat = NULL, params = c(0.078915 / 1.95, 0.003696,  0.013362,  1.2,
                                    0.424466,  1.164793),
                 betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
                 xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300,
                 iprint = TRUE, covdiag = FALSE, covsim = FALSE, L = chicagonet)

# alternative pattern, on the same linear network
l <- 20 / (spatstat.geom::volume(spatstat.geom::domain(spatstat.data::chicago)) * (200 - 25))
set.seed(12345)
stlppPOIS <- stlnpp::rpoistlpp(lambda = l, a = 25, b = 200, L = chicagonet)
Z <- as.stlp(stlppPOIS)

# run the local test
test <- localtest(X, Z, method = "K", k = 9)
```

```
plot(test)

## End(Not run)
```

---

plot.locstppm   *Plot of the fitted intensity of a local spatio-temporal Poisson process model*

Description

The function plots the local fitted intensity, displayed both in space and in space and time.

Usage

```
## S3 method for class 'locstppm'
plot(
  x,
  scaler = c("silverman", "IQR", "sd", "var"),
  do.points = TRUE,
  print.bw = FALSE,
  zap = 1e-05,
  par = TRUE,
  ...
)
```

Arguments

x	An object of class `locstppm`
scaler	Optional. Controls the value for a scalar representation of the spatial scale of the data. Either a character string, `"silverman"` (default), `"IQR"`, `"sd"`, or `"var"`; or positive numeric value(s). See OS.
do.points	Add points to plot
print.bw	It prints the estimated oversmoothing (OS) bandwidth selector
zap	Noise threshold factor (default to 0.00001). A numerical value greater than or equal to 1. If the range of pixel values is less than `zap * .Machine$double.eps`, the image will be treated as constant. This avoids displaying images which should be constant but contain small numerical errors.
par	Default to `TRUE`.
...	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Davies, T.M. and Hazelton, M.L. (2010), Adaptive kernel estimation of spatial relative risk, Statistics in Medicine, 29(23) 2423-2437.

Terrell, G.R. (1990). The maximal smoothing principle in density estimation, Journal of the American Statistical Association, 85, 470-477.

See Also

locstppm, print.locstppm, summary.locstppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)

plot(inh00_local)
plot(inh01_local)
```

---

plot.sepstlppm *Plot of the fitted intensity of a separable spatio-temporal Poisson model*

Description

The function plots the fitted intensity, displayed both in space and in space and time.

Usage

```
## S3 method for class 'sepstlppm'
plot(x,
do.points = TRUE,
par = TRUE,
...)
```

Arguments

x	An object of class `sepstlppm`
do.points	Add points to plot
par	Default to `TRUE`. If `par=FALSE`, the user is asked for input, before a new figure is drawn.
...	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

Examples

```
mod1 <- sepstlppm(valenciacrimes[1:2500, ], spaceformula = ~x,
timeformula = ~ crime_hour + week_day, L = valencianet)

plot.sepstlppm(mod1)
```

---

plot.sepstppm — *Plot of the fitted intensity of a separable spatio-temporal Poisson model*

Description

The function plots the fitted intensity, displayed both in space and in space and time.

Usage

```
## S3 method for class 'sepstppm'
plot(
  x,
  scaler = c("silverman", "IQR", "sd", "var"),
  do.points = TRUE,
  print.bw = FALSE,
  zap = 1e-05,
  par = TRUE,
  sig = NULL,
  ...
)
```

Arguments

`x`	An object of class `sepstppm`
`scaler`	Optional. Controls the value for a scalar representation of the spatial scale of the data. Either a character string, `"silverman"` (default), `"IQR"`, `"sd"`, or `"var"`; or positive numeric value(s). See OS.
`do.points`	Add points to plot
`print.bw`	It prints the estimated oversmoothing (OS) bandwidth selector
`zap`	Noise threshold factor (default to 0.00001). A numerical value greater than or equal to 1. If the range of pixel values is less than `zap * .Machine$double.eps`, the image will be treated as constant. This avoids displaying images which should be constant but contain small numerical errors.
`par`	Default to `TRUE`. If `par=FALSE`, the user is asked for input, before a new figure is drawn.
`sig`	Smoothing bandwidth for spatial representation
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

Examples

```
df1 <- valenciacrimes[valenciacrimes$x < 210000 & valenciacrimes$x > 206000
& valenciacrimes$y < 4377000 & valenciacrimes$y > 4373000, ]

mod1 <- sepstppm(df1, spaceformula = ~x * y,
                 timeformula = ~ crime_hour + week_day)

plot.sepstppm(mod1)
```

`plot.stlgcppm` *Plot of the fitted intensity of a LGCP model*

Description

The function plots the fitted intensity, displayed both in space and in space and time. In the case of local covariance parameters, the function returns the mean of the random intensity, displayed both in space and in space and time.

Usage

```
## S3 method for class 'stlgcppm'
plot(
  x,
  scaler = c("silverman", "IQR", "sd", "var"),
  do.points = TRUE,
  print.bw = FALSE,
  zap = 1e-05,
  par = TRUE,
  ...
)
```

Arguments

`x`	An object of class `stlgcppm`
`scaler`	Optional. Controls the value for a scalar representation of the spatial scale of the data. Either a character string, `"silverman"` (default), `"IQR"`, `"sd"`, or `"var"`; or positive numeric value(s). See OS.
`do.points`	Add points to plot
`print.bw`	It prints the estimated oversmoothing (OS) bandwidth selector
`zap`	Noise threshold factor (default to 0.00001). A numerical value greater than or equal to 1. If the range of pixel values is less than `zap * .Machine$double.eps`, the image will be treated as constant. This avoids displaying images which should be constant but contain small numerical errors.
`par`	Default to `TRUE`.
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Davies, T.M. and Hazelton, M.L. (2010), Adaptive kernel estimation of spatial relative risk, Statistics in Medicine, 29(23) 2423-2437.

Siino, M., Adelfio, G., and Mateu, J. (2018). Joint second-order parameter estimation for spatio-temporal log-Gaussian Cox processes. Stochastic environmental research and risk assessment, 32(12), 3525-3539.

Terrell, G.R. (1990). The maximal smoothing principle in density estimation, Journal of the American Statistical Association, 85, 470-477.

See Also

stlgcppm, print.stlgcppm, summary.stlgcppm, localsummary.stlgcppm, localplot.stlgcppm

Examples

```
# Example with complex seismic point pattern
data("greececatalog")

lgcp2 <- stlgcppm(greececatalog, formula = ~ x, first = "local", second = "global")
plot(lgcp2)
```

---

plot.stlp *Plot a* stlp *object*

Description

This function plots the point pattern on a linear network stored in the `stlp` object given in input, in a three panel plot representing the plot3D of the coordinates, and the marginal spatial and temporal coordinates.

Usage

```
## S3 method for class 'stlp'
plot(x, tcum = FALSE, marg = TRUE, col = 1, cols = "grey", ...)
```

Arguments

`x`	An object of class `stp`
`tcum`	Default to `FALSE`. If `TRUE`, the temporal point pattern is displayed cumulatively. A barplot is automatically plotted if there are repeated counts (typically with discrete times).
`marg`	Default to `TRUE`. If `FALSE`, only the spatio-temporal point pattern is plotted.
`col`	The color of the points. Default to `"black"`
`cols`	The color of the linear network. Default to `"grey"`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

See Also

stp, summary.stlp, print.stlp, as.stlpp, as.stlp

Examples

```
set.seed(12345)
stlpp1 <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)
df0 <- cbind(stlpp1$data$x, stlpp1$data$y, stlpp1$data$t)
L0 <- stlpp1$domain
stlp1 <- stp(df0, L0)

#plot
plot(stlp1)

#cumulative time occurrances
plot(stlp1, tcum = TRUE)

#change color of points
plot(stlp1, col = "blue")

#change color of network
plot(stlp1, cols = "magenta")

#display only in space-time
plot(stlp1, marg = FALSE)

#discrete times
stp2 <- stp(cbind(stlpp1$data$x, stlpp1$data$y, round(stlpp1$data$t)))
plot(stp2)
```

`plot.stp` *Plot a* `stp` *object*

Description

This function plots the point pattern stored in the stp object given in input, in a three panel plot representing the 3Dplot of the coordinates, and the marginal spatial and temporal coordinates.

Usage

```
## S3 method for class 'stp'
plot(x, tcum = FALSE, marg = TRUE, col = 1, ...)
```

Arguments

x	An object of class stp
tcum	Default to FALSE. If TRUE, the temporal point pattern is displayed cumulatively. A barplot is automatically plotted if there are repeated counts (typically with discrete times).
marg	Default to TRUE. If FALSE, only the spatio-temporal point pattern is plotted.
col	The color of the points. Default to "black"
...	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

See Also

stp, print.stp, summary.stp, as.stp, as.stpp

Examples

```
set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)
df0 <- cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], rpp1$xyt[, 3])
stp1 <- stp(df0)

#plot
plot(stp1)

#cumulative time occurrances
plot(stp1, tcum = TRUE)

#change color of points
plot(stp1, col = "blue")

#display only in space-time
plot(stp1, marg = FALSE)

#discrete times
stp2 <- stp(cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], round(rpp1$xyt[, 3] * 10)))
plot(stp2)
```

plot.stppm *Plot of the fitted intensity of a spatio-temporal Poisson process model*

Description

The function plots the fitted intensity, displayed both in space and in space and time.

Usage

```
## S3 method for class 'stppm'
plot(
  x,
  scaler = c("silverman", "IQR", "sd", "var"),
  do.points = TRUE,
  print.bw = FALSE,
  zap = 1e-05,
  par = TRUE,
  ...
)
```

Arguments

x	An object of class `stppm`
scaler	Optional. Controls the value for a scalar representation of the spatial scale of the data. Either a character string, `"silverman"` (default), `"IQR"`, `"sd"`, or `"var"`; or positive numeric value(s). See OS.
do.points	Add points to plot
print.bw	It prints the estimated oversmoothing (OS) bandwidth selector
zap	Noise threshold factor (default to 0.00001). A numerical value greater than or equal to 1. If the range of pixel values is less than `zap * .Machine$double.eps`, the image will be treated as constant. This avoids displaying images which should be constant but contain small numerical errors.
par	Default to `TRUE`.
...	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Davies, T.M. and Hazelton, M.L. (2010), Adaptive kernel estimation of spatial relative risk, Statistics in Medicine, 29(23) 2423-2437.

Terrell, G.R. (1990). The maximal smoothing principle in density estimation, Journal of the American Statistical Association, 85, 470-477.

See Also

stppm, print.stppm, summary.stppm, coef.stppm

Examples

```
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)

plot(inh1)
```

`predict.locstppm` *Extract the fitted intensity of a local spatio-temporal Poisson process model*

Description

Extract the fitted intensity of a local spatio-temporal Poisson process model

Usage

```
## S3 method for class 'locstppm'
predict(object, ...)
```

Arguments

`object`	An object of class `locstppm`
`...`	additional unused argument

Value

A vector containing the fitted intensity

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

locstppm, print.locstppm, summary.locstppm, plot.locstppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)

predict.locstppm(inh00_local)
predict.locstppm(inh01_local)
```

---

predict.stppm — *Extract the the fitted intensity of a spatio-temporal Poisson process model*

Description

Extract the the fitted intensity of a spatio-temporal Poisson process model

Usage

```
## S3 method for class 'stppm'
predict(object, ...)
```

Arguments

`object`	An object of class `stppm`
`...`	additional unused argument

Value

A vector containing the fitted intensity values

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

stppm, print.stppm, summary.stppm, plot.stppm

Examples

```
## Homogeneous
set.seed(2)
ph <- rstpp(lambda = 200, nsim = 1, seed = 2, verbose = TRUE)
hom1 <- stppm(ph, formula = ~ 1)

predict.stppm(hom1)

## Inhomogeneous
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)

predict.stppm(inh1)
```

`print.localdiag` *Print of the diagnostics' result on a spatio-temporal point process model*

Description

It prints the main information on the result of the local diagnostics performed with localdiag on either a `stp` or `stlp` object: whether the local test was run on point patterns lying on a linear network or not; the number of points in the analysed spatio-temporal point pattern `X`; the number of points of `X` which are identified as outlying by the previously performed local diagnostics.

Usage

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

Arguments

`x`	A `localdiag` object
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022) Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

See Also

infl, plot.localdiag, summary.localdiag

Examples

```
#load data
set.seed(12345)
id <- sample(1:nrow(etasFLP::catalog.withcov), 200)
cat <- etasFLP::catalog.withcov[id, ]
stp1 <- stp(cat[, 5:3])

#fit two competitor models
# and extract the fitted spatio-temporal intensity

lETAS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, 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)$l

lPOIS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, gamma = 0, d = 1,
q = 1.5, params.ind = c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
                        FALSE), 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)$l

# let's identify the outlying points at a .9 percentile

resETAS <- localdiag(stp1, lETAS, p = .9)
resPOIS <- localdiag(stp1, lPOIS, p = .9)

resETAS

# Points outlying from the 0.9 percentile
# of the anaysed spatio-temporal point pattern
# --------------------------------------
#   Analysed pattern X: 200 points
# 20 outlying points

resPOIS

# Points outlying from the 0.9 percentile
# of the anaysed spatio-temporal point pattern
# --------------------------------------
#   Analysed pattern X: 200 points
# 20 outlying points
```

`print.localtest` *Print of the result of the permutation local test*

Description

It prints the main information on the result of the local permutation test performed with localtest on either a `stp` or `stlp` object: whether the local test was run on point patterns lying on a linear network or not; the number of points in the background `X` and alternative `Z` patterns; the number of points in `X` which exhibit local differences in the second-order structure with respect to `Z`, according to the performed test.

Usage

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

Arguments

`x`	An object of class `localtest`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

Siino, M., Rodríguez-Cortés, F. J., Mateu, J. ,and Adelfio, G. (2018). Testing for local structure in spatiotemporal point pattern data. Environmetrics, 29(5-6), e2463.

See Also

localtest, summary.localtest, plot.localtest

Examples

```
## Not run:

# background pattern
set.seed(12345)
X <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(.05, 4),
           nsim = 1, seed = 2, verbose = TRUE)

# alternative pattern
set.seed(12345)
Z <- rstpp(lambda = 25, nsim = 1, seed = 2, verbose = TRUE)

# run the local test
test <- localtest(X, Z, method = "K", k = 9)

test

# Test for local differences between two
# spatio-temporal point patterns
# --------------------------------------
#   Backgound pattern X: 17
# Alternative pattern Z: 20
#
# 1 significant points at alpha = 0.05


# Linear networks

# background pattern
set.seed(12345)
X <- rETASlp(cat = NULL, params = c(0.078915 / 1.95, 0.003696,  0.013362,  1.2,
                                    0.424466,  1.164793),
                 betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
                 xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300,
                 iprint = TRUE, covdiag = FALSE, covsim = FALSE, L = chicagonet)

# alternative pattern, on the same linear network
l <- 20 / (spatstat.geom::volume(chicagonet) * (200 - 25))
set.seed(12345)
stlppPOIS <- stlnpp::rpoistlpp(lambda = l, a = 25, b = 200, L = chicagonet)
Z <- as.stlp(stlppPOIS)

# run the local test
test <- localtest(X, Z, method = "K", k = 9)

test

# Test for local differences between two
# spatio-temporal point patterns on a linear network
# --------------------------------------------------
#   Backgound pattern X: 31
```

```
# Alternative pattern Z: 22
#
# 19 significant points at alpha = 0.05


## End(Not run)
```

`print.locstppm` *Print of a fitted local spatio-temporal Poisson process model*

Description

The function prints the main information of the distribution of the parameters of a fitted local spatio-temporal Poisson process model.

Usage

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

Arguments

`x`	An object of class `locstppm`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

locstppm, summary.locstppm, plot.locstppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)

inh00_local
inh01_local
```

---

print.stlgcppm *Print of a fitted LGCP model*

Description

The function prints the main information on the fitted model. In this case of local parameters (both first- and second-order), the summary function contains information on their distributions.

Usage

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

Arguments

x	An object of class stlgcppm
...	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Siino, M., Adelfio, G., and Mateu, J. (2018). Joint second-order parameter estimation for spatio-temporal log-Gaussian Cox processes. Stochastic environmental research and risk assessment, 32(12), 3525-3539.

See Also

stlgcppm, print.stlgcppm, localsummary.stlgcppm, plot.stlgcppm, localplot.stlgcppm

Examples

```
# Example with complex seismic point pattern
data("greececatalog")

lgcp1 <- stlgcppm(greececatalog, formula = ~ 1, first = "global", second = "global")
lgcp2 <- stlgcppm(greececatalog, formula = ~ x, first = "local", second = "global")

lgcp1


# Joint minimum contrast fit
# for a log-Gaussian Cox process with
# global first-order intensity and
# global second-order intensity
# --------------------------------------------------
#   Homogeneous Poisson process
# with Intensity: 0.0064
#
# Estimated coefficients of the first-order intensity:
#   (Intercept)
# -5.052
# --------------------------------------------------
#   Covariance function: separable
#
# Estimated coefficients of the second-order intensity:
#   sigma  alpha   beta
# 10.984  0.224 47.076
# --------------------------------------------------
#   Model fitted in 0.873 minutes


lgcp2


# Joint minimum contrast fit
# for a log-Gaussian Cox process with
# local first-order intensity and
# global second-order intensity
# --------------------------------------------------
#   Inhomogeneous Poisson process
# with Trend: ~x
#
# Summary of estimated coefficients of the first-order intensity
# (Intercept)           x
# Min.   :-6.282   Min.   :-0.96831
# 1st Qu.:-2.387   1st Qu.:-0.36685
# Median : 2.122   Median :-0.25871
```

```
# Mean   : 2.052   Mean   :-0.26309
# 3rd Qu.: 4.569   3rd Qu.:-0.07325
# Max.   :17.638   Max.   : 0.10269
# --------------------------------------------------
#   Covariance function: separable
#
# Estimated coefficients of the second-order intensity:
#   sigma  alpha   beta
# 2.612  0.001 36.415
# --------------------------------------------------
#   Model fitted in 3.503 minutes
```

`print.stlp` *Print a* `stlp` *object*

Description

It prints the main information on the spatio-temporal point pattern on a linear network stored in the `stlp` object: the number of points; vertices and lines of the linear network; the enclosing spatial window; the temporal time period.

Usage

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

Arguments

`x`	An object of class `stlp`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

See Also

stp, plot.stlp, summary.stlp, as.stlpp, as.stlp

Examples

```
set.seed(12345)
stlpp1 <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)
df0 <- cbind(stlpp1$data$x, stlpp1$data$y, stlpp1$data$t)
L0 <- stlpp1$domain
stlp1 <- stp(df0, L0)

stlp1

# Spatio-temporal point pattern on a linear network
# 43 points
# Linear network with 19 vertices and 26 lines
# Enclosing window: rectangle = [-0.01, 5.1] x [-0.01, 5.1] units
# Time period: [0.043, 4.93]
```

---

print.stp — *Print a* `stp` *object*

Description

It prints the main information on the spatio-temporal point pattern stored in the `stp` object: the number of points; the enclosing spatial window; the temporal time period.

Usage

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

Arguments

`x`	An object of class `stp`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

See Also

stp, summary.stp, plot.stp, as.stp, as.stpp

Examples

```
set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)
df0 <- cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], rpp1$xyt[, 3])
stp1 <- stp(df0)

stp1

# Spatio-temporal point pattern
# 208 points
# Enclosing window: rectangle = [0.0011366, 0.9933775] x [0.0155277,
#                                                          0.9960438] units
# Time period: [0.004, 0.997]
```

`print.stppm` *Print of a fitted spatio-temporal Poisson process model*

Description

The function prints the main information of the fitted model.

Usage

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

Arguments

`x`	An object of class `stppm`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

stppm, print.stppm, plot.stppm, coef.stppm

Examples

```
## Homogeneous
set.seed(2)
ph <- rstpp(lambda = 200, nsim = 1, seed = 2, verbose = TRUE)
hom1 <- stppm(ph, formula = ~ 1)

hom1

# Homogeneous Poisson process
# with Intensity: 202.093
#
# Estimated coefficients:
#   (Intercept)
# 5.309

## Inhomogeneous
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)

inh1

# Inhomogeneous Poisson process
# with Trend: ~x
#
# Estimated coefficients:
#   (Intercept)          x
# 2.180       5.783
```

---

rETASlp — *Simulation of a spatio-temporal ETAS (Epidemic Type Aftershock Sequence) model on a linear network*

Description

This function simulates a spatio-temporal ETAS (Epidemic Type Aftershock Sequence) process on a linear network. It is firstly introduced ans employed for simulation studies in D'Angelo et al. (2021).

It follows the generating scheme for simulating a pattern from an Epidemic Type Aftershocks-Sequences (ETAS) process (Ogata and Katsura 1988) with conditional intensity function (CIF) as in Adelfio and Chiodi (2020), adapted for the space location of events to be constrained on a linear network.

The simulation on the network is guaranteed by the homogeneous spatial Poisson processes being generated on the network.

See the 'Details' section.

Usage

```
rETASlp(
  cat = NULL,
  params = c(0.078915, 0.003696, 0.013362, 1.2, 0.424466, 1.164793),
  betacov = 0.39,
  m0 = 2.5,
  b = 1.0789,
  tmin = 0,
  t.lag = 200,
  xmin = 600,
  xmax = 2200,
  ymin = 4000,
  ymax = 5300,
  iprint = TRUE,
  covdiag = FALSE,
  covsim = FALSE,
  L
)
```

Arguments

`cat`	NULL
`params`	A vector of parameters of the ETAS model to be simulated. See the 'Details' section.
`betacov`	Numerical array. Parameters of the covariates ETAS model
`m0`	Parameter for the background general intensity of the ETAS model. In the common seismic analyses it represents the threshold magnitude.
`b`	1.0789
`tmin`	Minimum value of time.
`t.lag`	200
`xmin`	Minimum of x coordinate range
`xmax`	Maximum of x coordinate range
`ymin`	Minimum of y coordinate range
`ymax`	Maximum of y coordinate range
`iprint`	Default `TRUE`
`covdiag`	Default `FALSE`
`covsim`	Default `FALSE`
`L`	linear network

Details

The CIF of an ETAS process as in Adelfio and Chiodi (2020) can be written as

$$\lambda_\theta(t, \mathbf{u}|\mathcal{H}_t) = \mu f(\mathbf{u}) + \sum_{t_j<t} \frac{\kappa_0 \exp(\eta_j)}{(t - t_j + c)^p} \{(\mathbf{u} - \mathbf{u}_j)^2 + d\}^{-q},$$

where

$\mathcal{H}_t$ is the past history of the process up to time t

μ is the large-scale general intensity

$f(\mathbf{u})$ is the spatial density

$\eta_j = \boldsymbol{\beta}'\mathbf{Z}_j$ is a linear predictor

$\mathbf{Z}_j$ the external known covariate vector, including the magnitude

$\boldsymbol{\theta} = (\mu, \kappa_0, c, p, d, q, \boldsymbol{\beta})$ are the parameters to be estimated

κ_0 is a normalising constant

c and p are characteristic parameters of the seismic activity of the given region,

and d and q are two parameters related to the spatial influence of the mainshock

In the usual ETAS model for seismic analyses, the only external covariate represents the magnitude, $\boldsymbol{\beta} = \alpha$, as $\eta_j = \boldsymbol{\beta}'\mathbf{Z}_j = \alpha(m_j - m_0)$, where m_j is the magnitude of the j^{th} event and m_0 the threshold magnitude, that is, the lower bound for which earthquakes with higher values of magnitude are surely recorded in the catalogue.

Value

A `stlp` object

Author(s)

Nicoletta D'Angelo and Marcello Chiodi

References

Adelfio, G., and Chiodi, M. (2021). Including covariates in a space-time point process with application to seismicity. Statistical Methods & Applications, 30(3), 947-971.

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

Ogata, Y., and Katsura, K. (1988). Likelihood analysis of spatial inhomogeneity for marked point patterns. Annals of the Institute of Statistical Mathematics, 40(1), 29-39.

Examples

```
set.seed(5)
X <- rETASlp(cat = NULL, params = c(0.1293688525, 0.003696, 0.013362, 1.2,0.424466,  1.164793),
```

```
betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300, iprint = TRUE,
covdiag = FALSE, covsim = FALSE, L = chicagonet)
```

---

rETASp *Simulation of a spatio-temporal ETAS (Epidemic Type Aftershock Sequence) model*

Description

This function simulates a spatio-temporal ETAS (Epidemic Type Aftershock Sequence) process.

It follows the generating scheme for simulating a pattern from an Epidemic Type Aftershocks-Sequences (ETAS) process (Ogata and Katsura 1988) with conditional intensity function (CIF) as in Adelfio and Chiodi (2020), adapted for the space location of events to be constrained.

See the 'Details' section.

Usage

```
rETASp(
  cat = NULL,
  params = c(0.078915, 0.003696, 0.013362, 1.2, 0.424466, 1.164793),
  betacov = 0.39,
  m0 = 2.5,
  b = 1.0789,
  tmin = 0,
  t.lag = 200,
  xmin = 600,
  xmax = 2200,
  ymin = 4000,
  ymax = 5300,
  iprint = TRUE,
  covdiag = FALSE,
  covsim = FALSE
)
```

Arguments

cat	NULL
params	A vector of parameters of the ETAS model to be simulated. See the 'Details' section.
betacov	Numerical array. Parameters of the ETAS model covariates.
m0	Parameter for the background general intensity of the ETAS model. In the common seismic analyses it represents the threshold magnitude.

`b`	1.0789
`tmin`	Minimum value of time.
`t.lag`	200
`xmin`	Minimum of x coordinate range
`xmax`	Maximum of x coordinate range
`ymin`	Minimum of y coordinate range
`ymax`	Maximum of y coordinate range
`iprint`	Default `TRUE`
`covdiag`	Default `FALSE`
`covsim`	Default `FALSE`

Details

The CIF of an ETAS process as in Adelfio and Chiodi (2020) can be written as

$$\lambda_\theta(t, \mathbf{u}|\mathcal{H}_t) = \mu f(\mathbf{u}) + \sum_{t_j<t} \frac{\kappa_0 \exp(\eta_j)}{(t - t_j + c)^p} \{(\mathbf{u} - \mathbf{u}_j)^2 + d\}^{-q},$$

where

$\mathcal{H}_t$ is the past history of the process up to time t

μ is the large-scale general intensity

$f(\mathbf{u})$ is the spatial density

$\eta_j = \boldsymbol{\beta}'\mathbf{Z}_j$ is a linear predictor

$\mathbf{Z}_j$ the external known covariate vector, including the magnitude

$\boldsymbol{\theta} = (\mu, \kappa_0, c, p, d, q, \boldsymbol{\beta})$ are the parameters to be estimated

κ_0 is a normalising constant

c and p are characteristic parameters of the seismic activity of the given region,

and d and q are two parameters related to the spatial influence of the mainshock

In the usual ETAS model for seismic analyses, the only external covariate represents the magnitude, $\boldsymbol{\beta} = \alpha$, as $\eta_j = \boldsymbol{\beta}'\mathbf{Z}_j = \alpha(m_j - m_0)$, where m_j is the magnitude of the j^{th} event and m_0 the threshold magnitude, that is, the lower bound for which earthquakes with higher values of magnitude are surely recorded in the catalogue.

Value

A `stp` object

Author(s)

Nicoletta D'Angelo and Marcello Chiodi

References

Adelfio, G., and Chiodi, M. (2021). Including covariates in a space-time point process with application to seismicity. Statistical Methods & Applications, 30(3), 947-971.

Ogata, Y., and Katsura, K. (1988). Likelihood analysis of spatial inhomogeneity for marked point patterns. Annals of the Institute of Statistical Mathematics, 40(1), 29-39.

Examples

```
set.seed(95)
X <- rETASp(cat = NULL, params = c(0.1293688525, 0.003696, 0.013362, 1.2,0.424466,  1.164793),
                    betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
                    xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300, iprint = TRUE,
                    covdiag = FALSE, covsim = FALSE)
```

---

rstlpp — *Simulate homogeneous and inhomogeneous spatio-temporal Poisson point patterns on linear networks*

Description

This function creates a `stlp` object, simulating a spatio-temporal point pattern on a linear network following either an homogeneous or inhomogeneous intensity

Usage

```
rstlpp(
  lambda = 500,
  nsim = 1,
  seed = 2,
  verbose = TRUE,
  par = NULL,
  minX = 0,
  maxX = 1,
  minY = 0,
  maxY = 1,
  minT = 0,
  maxT = 1,
  L
)
```

Arguments

`lambda`	Expected number of points to simulate
`nsim`	Number of patterns to simulate. Default to 1.
`seed`	Seed to set, if ones wished to reproduce the analyses

verbose	Default to TRUE
par	Parameters of the reference intensity
minX	Minimum of x coordinate range
maxX	Maximum of x coordinate range
minY	Minimum of y coordinate range
maxY	Maximum of y coordinate range
minT	Minimum of t coordinate range
maxT	Maximum of t coordinate range
L	linear network

Value

A stp object

Author(s)

Nicoletta D'Angelo

Examples

```
# homogeneous Poisson processes

h1 <- rstlpp(lambda = 500, nsim = 1, seed = 2, verbose = TRUE,
L = chicagonet)

h2 <- rstlpp(lambda = 50, nsim = 1, seed = 2, verbose = TRUE, L = stlnpp::easynet)

# inhomogeneous Poisson process

inh <- rstlpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(4, 1.5),
nsim = 1, seed = 2, verbose = TRUE, L = stlnpp::easynet)
```

---

rstpp *Simulate homogeneous and inhomogeneous spatio-temporal Poisson point patterns*

Description

This function creates a stp object, simulating a spatio-temporal point pattern following either an homogeneous or inhomogeneous intensity

Usage

```
rstpp(
  lambda = 500,
  nsim = 1,
  seed = 2,
  verbose = TRUE,
  par = NULL,
  minX = 0,
  maxX = 1,
  minY = 0,
  maxY = 1,
  minT = 0,
  maxT = 1
)
```

Arguments

lambda	Expected number of points to simulate
nsim	Number of patterns to simulate. Default to 1.
seed	Seed to set, if ones wished to reproduce the analyses
verbose	Default to TRUE
par	Parameters of the reference intensity
minX	Minimum of x coordinate range
maxX	Maximum of x coordinate range
minY	Minimum of y coordinate range
maxY	Maximum of y coordinate range
minT	Minimum of t coordinate range
maxT	Maximum of t coordinate range

Value

A `stp` object

Author(s)

Nicoletta D'Angelo

See Also

stppm, AIC.stppm, BIC.stppm

Examples

```
# homogeneous Poisson processes
h1 <- rstpp(lambda = 500, nsim = 1, seed = 2, verbose = TRUE)

h2 <- rstpp(lambda = 500, nsim = 1, seed = 2, verbose = TRUE, minX = 0,
             maxX = 2, minY = 3, maxY = 5, minT = 1, maxT = 9)

h3 <- rstpp(lambda = 900, nsim = 3, seed = 2, verbose = TRUE)

# inhomogeneous Poisson process
inh <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
```

`sepstlppm` *Fit a separable spatio-temporal Poisson process model on a linear network*

Description

Fit a separable spatio-temporal Poisson process model on a linear network

Usage

```
sepstlppm(x, spaceformula, timeformula, L)
```

Arguments

`x`	A dataframe
`spaceformula`	A formula for the spatial component. See lppm for details
`timeformula`	A formula for the temporal component. It fits a log-linear model with the glm function
`L`	A linear network of class `linnet`

Value

An object of class `sepstlppm`

Examples

```
mod1 <- sepstlppm(valenciacrimes[1:2500, ], spaceformula = ~x,
timeformula = ~ crime_hour + week_day, L = valencianet)
```

sepstppm *Fit a separable spatio-temporal Poisson process model*

Description

Fit a separable spatio-temporal Poisson process model

Usage

```
sepstppm(x, spaceformula, timeformula)
```

Arguments

`x`	A dataframe
`spaceformula`	A formula for the spatial component. See ppm for details
`timeformula`	A formula for the temporal component. It fits a log-linear model with the glm function

Value

An object of class `sepstppm`

Examples

```
df1 <- valenciacrimes[valenciacrimes$x < 210000 & valenciacrimes$x > 206000
& valenciacrimes$y < 4377000 & valenciacrimes$y > 4373000, ]

mod1 <- sepstppm(df1, spaceformula = ~x * y,
                timeformula = ~ crime_hour + week_day)
```

stlgcppm *Fit a log-Gaussian Cox process model to a spatio-temporal point pattern*

Description

This function estimates a local log-Gaussian Cox process (LGCP), following the *locally weighted minimum contrast* procedure introduced in D'Angelo et al. (2023).

Three covariances are available: separable exponential, Gneiting, and De Iaco-Cesare.

If both `first` and `second` arguments are set to `global`, a log-Gaussian Cox process is fitted by means of the *joint minimum contrast* procedure proposed in Siino et al. (2018).

Usage

```
stlgcppm(
  X,
  formula,
  verbose = TRUE,
  cov = "separable",
  first = "local",
  second = "local",
  mult = 4,
  hs = "global",
  npx0 = 10,
  npt0 = 10,
  itnmax = 100
)
```

Arguments

`X`	A stp object
`formula`	An object of class `formula`: a symbolic description of the first-order intensity to be fitted. The current version only supports formulas depending on the spatial and temporal coordinates: `x`, `y`, `t`.
`verbose`	Default to TRUE
`cov`	Covariance function to be fitted for the second-order intensity function. Default to `separable`. Other options are `gneiting` and `iaco-cesare`".
`first`	Character string indicating whether to fit a first-order intensity function with global or local parameters: either `local` (default) or `global`.
`second`	Character string indicating whether to fit a second-order intensity function with global or local parameters: either `local` (default) or `global`.
`mult`	The multiplicand of the number of data points, for setting the number of dummy points to generate for the quadrature scheme
`hs`	Character string indicating whether to select fixed or variable bandwidths for the kernel weights to be used in the log-likelihood. In any of those cases, the well-supported rule-of-thumb for choosing the bandwidth of a Gaussian kernel density estimator is employed. If `hs = "global"` (default), a fixed bandwidth is selected. If `hs = "local"`, an individual badwidth is selected for each point in the pattern `X`.
`npx0`	A positive integer representing the spatial distance to np-th closest event. Used in the computation of the local bandwidth. Suitable values are in the range from 10 (default) to 100.
`npt0`	A positive integer representing the temporal distance to np-th closest event. Used in the computation of the local bandwidth. Suitable values are in the range from 10 (default) to 100.
`itnmax`	Maximum number of iterations to run in the optimization procedure for the estiamtion of the second-order intensity parameters.

Details

Following the inhomogeneous specification in Diggle et al. (2013), we consider LGCPs with intensity

$$\Lambda(\mathbf{u},t) = \lambda(\mathbf{u},t)\exp(S(\mathbf{u},t)).$$

Following Siino et al. (2018), the second-order parameters $\boldsymbol{\psi}$ are found by minimising

$$M_J\{\boldsymbol{\psi}\} = \int_{h_0}^{h_{max}} \int_{r_0}^{r_{max}} \phi(r,h)\{\nu[\hat{J}(r,h)] - \nu[J(r,h;\boldsymbol{\psi})]\}^2 \mathrm{d}r\mathrm{d}h,$$

where $\phi(r,h)$ is a weight that depends on the space-time distance and ν is a transformation function.

They suggest $\phi(r,h) = 1$ and ν as the identity function, while r_{max} and h_{max} are selected as 1/4 of the maximum observable spatial and temporal distances.

Following D'Angelo et al. (2023), we can fit a localised version of the LGCP, that is, obtain a vector of parameters $\boldsymbol{\psi}_i$ for each point i, by minimising

$$M_{J,i}\{\boldsymbol{\psi}_i\} = \int_{h_0}^{h_{max}} \int_{r_0}^{r_{max}} \phi(r,h)\{\nu[\bar{J}_i(r,h)] - \nu[J(r,h;\boldsymbol{\psi})]\}^2 \mathrm{d}r\mathrm{d}h,$$

where $\bar{J}_i(r,h) = \frac{\sum_{i=1}^n \hat{J}_i(r,h) w_i}{\sum_{i=1}^n w_i}$ is the average of the local functions $\hat{J}_i(r,h)$, weighted by some point-wise kernel estimates.

In particular, we consider $\hat{J}_i(\cdot)$ as the local spatio-temporal pair correlation function (Gabriel et al, 2013) documented in LISTAhat.

Value

A list of the class `stlgcppm`, containing

`IntCoefs` The fitted coefficients of the first-order intensity function

`CovCoefs` The fitted coefficients of the second-order intensity function

`X` The stp objectt provided as input

`formula` The formula provided as input

`cov` A string with the chosen covariance type

`l` Fitted first-order intensity

`mu` Mean function of the random intensity

`mod_global` The glm object of the model fitted to the quadrature scheme for the first-order intensity parameters estimation

`newdata` The data used to fit the model, without the dummy points

`time` Time elapsed to fit the model, in minutes

Author(s)

Nicoletta D'Angelo, Giada Adelfio, and Marianna Siino

References

Baddeley, A. (2017). Local composite likelihood for spatial point processes. Spatial Statistics, 22, 261-295.

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Diggle, P. J., Moraga, P., Rowlingson, B., and Taylor, B. M. (2013). Spatial and spatio-temporal log-gaussian cox processes: extending the geostatistical paradigm. Statistical Science, 28(4):542–563.

Gabriel, E., Rowlingson, B. S., and Diggle, P. J. (2013). stpp: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns. Journal of Statistical Software, 53(2), 1–29. https://doi.org/10.18637/jss.v053.i02

Siino, M., Adelfio, G., and Mateu, J. (2018). Joint second-order parameter estimation for spatio-temporal log-Gaussian Cox processes. Stochastic environmental research and risk assessment, 32(12), 3525-3539.

See Also

print.stlgcppm, summary.stlgcppm, localsummary.stlgcppm, plot.stlgcppm, localplot.stlgcppm

Examples

```
# Example with complex seismic point pattern
data("greececatalog")

##

# If both first and second arguments are set to "global", a log-Gaussian
# Cox process is fitted by means of the joint minimum contrast.

lgcp1 <- stlgcppm(greececatalog, formula = ~ 1, first = "global", second = "global")

 ##

 # If first = "local", local parameters for the first-order intensity are
 # provided. In this case, the summary function contains information on
 # their distributions.

 lgcp2 <- stlgcppm(greececatalog, formula = ~ x, first = "local", second = "global")

 ##

# Finally, if second = "local" (default option), the model with local
 #parameters for the covariance is fitted.

 lgcp3 <- stlgcppm(greececatalog, formula = ~ 1, first = "global", second = "local")

 ##
```

```
#  It is also possible to fit local parameters for both the first and
#  second-order intensity, but we abstain from addressing this case here.
```

---

stp *Create* `stp` *and* `stlp` *objects for point patterns storage*

Description

This function creates a `stp` object as a dataframe with three columns: `x`, `y`, and `t`. If also the linear network `L`, of class linnet, is provided, a `stlp` object is created instead.

Usage

```
stp(df, L)
```

Arguments

`df`	A matrix with three columns, containing to two space and the temporal coordinates
`L`	Optional. The linear network of class `linnet`. If provided, the function returns a `stlp` object.

Value

An `stp` or `stlpp` object, depending on whether or not an object of class linnet is provided for the `L` argument.

Author(s)

Nicoletta D'Angelo

See Also

summary.stp, print.stp, plot.stp

stppm, AIC.stppm, BIC.stppm, print.stp, summary.stp, plot.stp, as.stp, as.stpp, print.stlp, summary.stlp, plot.stlp, as.stlp, as.stlpp

Examples

```
# stp

set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)
df0 <- cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], rpp1$xyt[, 3])
stp1 <- stp(df0)



# stlp

set.seed(12345)
stlpp1 <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)
df0 <- cbind(stlpp1$data$x, stlpp1$data$y, stlpp1$data$t)
L0 <- stlpp1$domain
stlp1 <- stp(df0, L0)
```

---

stppm *Fit a Poisson process model to a spatio-temporal point pattern*

Description

This function fits a Poisson process model to an observed spatio-temporal point pattern stored in a `stp` object.

We assume that the template model is a Poisson process, with a parametric intensity or rate function $\lambda(\mathbf{u}, t; \theta)$ with space and time locations $\mathbf{u} \in W, t \in T$ and parameters $\theta \in \Theta$.

Estimation is performed through the fitting of a `glm` using a spatio-temporal version of the quadrature scheme by Berman and Turner (1992).

See the 'Details' section.

Usage

```
stppm(X, formula, verbose = TRUE, mult = 4)
```

Arguments

`X`	A `stp` object
`formula`	An object of class `"formula"`: a symbolic description of the model to be fitted. The current version only supports formulas depending on the spatial and temporal coordinates: `x`, `y`, `t`.
`verbose`	Default to `TRUE`
`mult`	The multiplicand of the number of data points, for setting the number of dummy points to generate for the quadrature scheme

Details

The log-likelihood of the template model is

$$\log L(\theta) = \sum_i \lambda(\mathbf{u}_i, t_i; \theta) - \int_W \int_T \lambda(\mathbf{u}, t; \theta) \mathrm{d}t \mathrm{d}u$$

up to an additive constant, where the sum is over all points $\mathbf{u}_i$ in the spatio-temporal point process X.

Following Berman and Turner (1992), we use a finite quadrature approximation to the log-likelihood. Renaming the data points as $\mathbf{x}_1, \dots, \mathbf{x}_n$ with $(\mathbf{u}_i, t_i) = \mathbf{x}_i$ for $i = 1, \dots, n$, then generate m additional 'dummy points' $(\mathbf{u}_{n+1}, t_{n+1}) \dots, (\mathbf{u}_{m+n}, t_{m+n})$ to form a set of $n + m$ quadrature points (where $m > n$).

Then we determine quadrature weights $a_1, \dots, a_m$ so that integrals in the log-likelihood can be approximated by a Riemann sum

$$\int_W \int_T \lambda(\mathbf{u}, t; \theta) \mathrm{d}t \mathrm{d}u \approx \sum_{k=1}^{n+m} a_k \lambda(\mathbf{u}_k, t_k; \theta)$$

where a_k are the quadrature weights such that $\sum_{k=1}^{n+m} a_k = l(W \times T)$ where l is the Lebesgue measure.

Then the log-likelihood of the template model can be approximated by

$$\log L(\theta) \approx \sum_i \log \lambda(\mathbf{x}_i; \theta) + \sum_j (1 - \lambda(\mathbf{u}_j, t_j; \theta)) a_j = \sum_j e_j \log \lambda(\mathbf{u}_j, t_j; \theta) + (1 - \lambda(\mathbf{u}_j, t_j; \theta)) a_j$$

where $e_j = 1\{j \leq n\}$ is the indicator that equals 1 if u_j is a data point. Writing $y_j = e_j / a_j$ this becomes

$$\log L(\theta) \approx \sum_j a_j (y_j \log \lambda(\mathbf{u}_j, t_j; \theta) - \lambda(\mathbf{u}_j, t_j; \theta)) + \sum_j a_j.$$

Apart from the constant $\sum_j a_j$, this expression is formally equivalent to the weighted log-likelihood of a Poisson regression model with responses y_j and means $\lambda(\mathbf{u}_j, t_j; \theta) = \exp(\theta Z(\mathbf{u}_j, t_j) + B(\mathbf{u}_j, t_j))$.

This is maximised by this function by using standard GLM software.

In detail, we define the spatio-temporal quadrature scheme by considering a spatio-temporal partition of $W \times T$ into cubes C_k of equal volume ν, assigning the weight $a_k = \nu / n_k$ to each quadrature point (dummy or data) where n_k is the number of points that lie in the same cube as the point u_k (Raeisi et al, 2021).

The number of dummy points should be sufficient for an accurate estimate of the likelihood. Following Baddeley et al. (2000) and Raeisi et al. (2021), we start with a number of dummy points $m \approx 4n$, increasing it until $\sum_k a_k = l(W \times T)$.

Value

An object of class `stppm`. A list of

`IntCoefs` The fitted coefficients

`X` The `stp` object provided as input

`nX` The number of points in `X`

`I` Vector indicating which points are dummy or data

`y_resp` The response variable of the model fitted to the quadrature scheme

`formula` The formula provided as input

`l` Fitted intensity

`mod_global` The `glm` object of the model fitted to the quadrature scheme

`newdata` The data used to fit the model, without the dummy points

`time` Time elapsed to fit the model, in minutes

Author(s)

Nicoletta D'Angelo

References

Baddeley, A. J., Møller, J., and Waagepetersen, R. (2000). Non-and semi-parametric estimation of interaction in inhomogeneous point patterns. Statistica Neerlandica, 54(3):329–350

Berman, M. and Turner, T. R. (1992). Approximating point process likelihoods with glim. Journal of the Royal Statistical Society: Series C (Applied Statistics), 41(1):31–38

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Raeisi, M., Bonneu, F., and Gabriel, E. (2021). A spatio-temporal multi-scale model for geyer saturation point process: application to forest fire occurrences. Spatial Statistics, 41:100492.

See Also

locstppm, AIC.stppm, BIC.stppm

Examples

```
## Homogeneous
set.seed(2)
ph <- rstpp(lambda = 200, nsim = 1, seed = 2, verbose = TRUE)
hom1 <- stppm(ph, formula = ~ 1)

## Inhomogeneous
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)
```

`summary.localdiag` *Summary of the diagnostics performed on a spatio-temporal point process model*

Description

It summarises the main information on the result of the local diagnostics performed with localdiag on either a `stp` or `stlp` object: whether the local test was run on point patterns lying on a linear network or not; the number of points in the analysed spatio-temporal point pattern `X`; the number of points of `X` which are identified as outlying by the previously performed local diagnostics.

Usage

```
## S3 method for class 'localdiag'
summary(object, ...)
```

Arguments

`object`	A `localdiag` object
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

Adelfio, G., Siino, M., Mateu, J., and Rodríguez-Cortés, F. J. (2020). Some properties of local weighted second-order statistics for spatio-temporal point processes. Stochastic Environmental Research and Risk Assessment, 34(1), 149-168.

D'Angelo, N., Adelfio, G. and Mateu, J. (2022) Local inhomogeneous second-order characteristics for spatio-temporal point processes on linear networks. Stat Papers. https://doi.org/10.1007/s00362-022-01338-4

See Also

infl, plot.localdiag, print.localdiag

Examples

```
#load data
set.seed(12345)
id <- sample(1:nrow(etasFLP::catalog.withcov), 200)
cat <- etasFLP::catalog.withcov[id, ]
stp1 <- stp(cat[, 5:3])

#fit two competitor models
# and extract the fitted spatio-temporal intensity

lETAS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, 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)$l

lPOIS <- etasFLP::etasclass(cat.orig = cat, magn.threshold = 2.5, magn.threshold.back = 3.9,
mu = 0.3, k0 = 0.02, c = 0.015, p = 1.01, gamma = 0, d = 1,
q = 1.5, params.ind = c(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
                        FALSE), 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)$l

# let's identify the outlying points at a .9 percentile

resETAS <- localdiag(stp1, lETAS, p = .9)
resPOIS <- localdiag(stp1, lPOIS, p = .9)

summary(resETAS)

# Points outlying from the 0.9 percentile
# of the anaysed spatio-temporal point pattern
# --------------------------------------
#   Analysed pattern X: 200 points
# 20 outlying points


summary(resPOIS)

# Points outlying from the 0.9 percentile
# of the anaysed spatio-temporal point pattern
# --------------------------------------
#   Analysed pattern X: 200 points
# 20 outlying points
```

summary.localtest *Summary of the result of the permutation local test*

Description

It summarises the main information on the result of the local permutation test performed with localtest on either a `stp` or `stlp` object: whether the local test was run on point patterns lying on a linear network or not; the number of points in the background `X` and alternative `Z` patterns; the number of points in `X` which exhibit local differences in the second-order structure with respect to `Z`, according to the performed test.

Usage

```
## S3 method for class 'localtest'
summary(object, ...)
```

Arguments

`object`	An object of class `localtest`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2021). Assessing local differences between the spatio-temporal second-order structure of two point patterns occurring on the same linear network. Spatial Statistics, 45, 100534.

Siino, M., Rodríguez-Cortés, F. J., Mateu, J. ,and Adelfio, G. (2018). Testing for local structure in spatiotemporal point pattern data. Environmetrics, 29(5-6), e2463.

See Also

localtest, print.localtest, plot.localtest

Examples

```
# Euclidean
# background pattern
set.seed(12345)
X <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(.05, 4),
           nsim = 1, seed = 2, verbose = TRUE)

# alternative pattern
set.seed(12345)
Z <- rstpp(lambda = 25, nsim = 1, seed = 2, verbose = TRUE)

# run the local test
test <- localtest(X, Z, method = "K", k = 9)



summary(test)

# Test for local differences between two
# spatio-temporal point patterns
# --------------------------------------
#   Backgound pattern X: 20
# Alternative pattern Z: 17
#
# 1 significant points at alpha = 0.05

# Linear networks

# background pattern
set.seed(12345)
X <- rETASlp(cat = NULL, params = c(0.078915 / 1.95, 0.003696,  0.013362,  1.2,
                                    0.424466,  1.164793),
                 betacov = 0.5, m0 = 2.5, b = 1.0789, tmin = 0, t.lag = 200,
                 xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300,
                 iprint = TRUE, covdiag = FALSE, covsim = FALSE, L = chicagonet)

# alternative pattern, on the same linear network
l <- 20 / (spatstat.geom::volume(chicagonet) * (200 - 25))
set.seed(12345)
stlppPOIS <- stlnpp::rpoistlpp(lambda = l, a = 25, b = 200, L = chicagonet)
Z <- as.stlp(stlppPOIS)

# run the local test
test <- localtest(X, Z, method = "K", k = 9)

summary(test)

# Test for local differences between two
# spatio-temporal point patterns on a linear network
```

```
# --------------------------------------------------
#   Backgound pattern X: 31
# Alternative pattern Z: 22
#
# 19 significant points at alpha = 0.05
```

`summary.locstppm` *Summary of a fitted local spatio-temporal Poisson process model*

Description

The function summarises the main information on the distribution of the parameters of a fitted local spatio-temporal Poisson process model.

Usage

```
## S3 method for class 'locstppm'
summary(object, ...)
```

Arguments

`object`	An object of class `locstppm`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

locstppm, print.locstppm, plot.locstppm

Examples

```
set.seed(2)
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
             nsim = 1, seed = 2, verbose = TRUE)
inh00_local <- locstppm(pin, formula = ~ 1)
inh01_local <- locstppm(pin, formula = ~ x)

summary(inh00_local)
summary(inh01_local)
```

`summary.stlgcppm` *Summary of a fitted LGCP model*

Description

The function Summarises the main information on the fitted model. provided. In this case of local parameters (both first- and second-order), the summary function contains information on their distributions.

Usage

```
## S3 method for class 'stlgcppm'
summary(object, ...)
```

Arguments

`object`	An object of class `stlgcppm`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo and Giada Adelfio

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

Siino, M., Adelfio, G., and Mateu, J. (2018). Joint second-order parameter estimation for spatio-temporal log-Gaussian Cox processes. Stochastic environmental research and risk assessment, 32(12), 3525-3539.

See Also

stlgcppm, print.stlgcppm, localsummary.stlgcppm, plot.stlgcppm, localplot.stlgcppm

Examples

```
# Example with complex seismic point pattern
data("greececatalog")

lgcp1 <- stlgcppm(greececatalog, formula = ~ 1, first = "global", second = "global")
lgcp2 <- stlgcppm(greececatalog, formula = ~ x, first = "local", second = "global")

summary(lgcp1)


# Joint minimum contrast fit
# for a log-Gaussian Cox process with
# global first-order intensity and
# global second-order intensity
# --------------------------------------------------
#   Homogeneous Poisson process
# with Intensity: 0.0064
#
# Estimated coefficients of the first-order intensity:
#   (Intercept)
# -5.052
# --------------------------------------------------
#   Covariance function: separable
#
# Estimated coefficients of the second-order intensity:
#   sigma  alpha   beta
# 10.984  0.224 47.076
# --------------------------------------------------
#   Model fitted in 0.873 minutes


summary(lgcp2)


# Joint minimum contrast fit
# for a log-Gaussian Cox process with
# local first-order intensity and
```

```
# global second-order intensity
# --------------------------------------------------
#   Inhomogeneous Poisson process
# with Trend: ~x
#
# Summary of estimated coefficients of the first-order intensity
# (Intercept)           x
# Min.   :-6.282   Min.   :-0.96831
# 1st Qu.:-2.387   1st Qu.:-0.36685
# Median : 2.122   Median :-0.25871
# Mean   : 2.052   Mean   :-0.26309
# 3rd Qu.: 4.569   3rd Qu.:-0.07325
# Max.   :17.638   Max.   : 0.10269
# --------------------------------------------------
#   Covariance function: separable
#
# Estimated coefficients of the second-order intensity:
#   sigma  alpha   beta
# 2.612  0.001 36.415
# --------------------------------------------------
#   Model fitted in 3.503 minutes
```

`summary.stlp` *Summary of a* `stlp` *object*

Description

It prints the main information on the spatio-temporal point pattern on a linear network stored in the `stlp` object: the number of points; vertices and lines of the linear network; the enclosing spatial window; the temporal time period.

Usage

```
## S3 method for class 'stlp'
summary(object, ...)
```

Arguments

`object`	An object of class `stlp`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

See Also

stp, plot.stlp, print.stlp, as.stlpp, as.stlp

Examples

```
set.seed(12345)
stlpp1 <- stlnpp::rpoistlpp(.2, a = 0, b = 5, L = stlnpp::easynet)
df0 <- cbind(stlpp1$data$x, stlpp1$data$y, stlpp1$data$t)
L0 <- stlpp1$domain
stlp1 <- stp(df0, L0)

summary(stlp1)
```

---

summary.stp *Summary of a `stp` object*

Description

It prints the summary statistics of the spatial and temporal coordinates of the spatio-temporal point pattern stored in the `stp` object.

Usage

```
## S3 method for class 'stp'
summary(object, ...)
```

Arguments

`object`	An object of class `stp`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

See Also

stp, print.stp, plot.stp, as.stp, as.stpp

Examples

```
set.seed(12345)
rpp1 <- stpp::rpp(lambda = 200, replace = FALSE)
df0 <- cbind(rpp1$xyt[, 1], rpp1$xyt[, 2], rpp1$xyt[, 3])
stp1 <- stp(df0)

summary(stp1)

#       x                  y                 t
# Min.   :0.001137   Min.   :0.01553   Min.   :0.004254
# 1st Qu.:0.290510   1st Qu.:0.27316   1st Qu.:0.266398
# Median :0.559250   Median :0.53679   Median :0.524548
# Mean   :0.536354   Mean   :0.52031   Mean   :0.513976
# 3rd Qu.:0.793498   3rd Qu.:0.78442   3rd Qu.:0.758390
# Max.   :0.993378   Max.   :0.99604   Max.   :0.996996
```

---

summary.stppm — *Summary of a fitted spatio-temporal Poisson process model*

Description

The function summarises the main information of the fitted model.

Usage

```
## S3 method for class 'stppm'
summary(object, ...)
```

Arguments

`object`	An object of class `stppm`
`...`	additional unused argument

Value

No return value, called for side effects

Author(s)

Nicoletta D'Angelo

References

D'Angelo, N., Adelfio, G., and Mateu, J. (2023). Locally weighted minimum contrast estimation for spatio-temporal log-Gaussian Cox processes. Computational Statistics & Data Analysis, 180, 107679.

See Also

stppm, print.stppm, plot.stppm, coef.stppm

Examples

```
## Homogeneous
set.seed(2)
ph <- rstpp(lambda = 200, nsim = 1, seed = 2, verbose = TRUE)
hom1 <- stppm(ph, formula = ~ 1)

summary(hom1)

# Homogeneous Poisson process
# with Intensity: 202.093
#
# Estimated coefficients:
#   (Intercept)
# 5.309

## Inhomogeneous
pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2]*x)}, par = c(2, 6),
nsim = 1, seed = 2, verbose = TRUE)
inh1 <- stppm(pin, formula = ~ x)

summary(inh1)

# Inhomogeneous Poisson process
# with Trend: ~x
#
# Estimated coefficients:
#   (Intercept)          x
# 2.180       5.783
#
```

`valenciacrimes` *Crimes in Valencia in 2019*

Description

A dataset in `data.frame` format containing the 10929 crimes occurred in Valencia, Spain, in 2019.

Usage

```
data(valenciacrimes)
```

Format

A `data.frame`

Details

The variables are as follows:

- crime_id.
- crime_date.
- crime_time.
- month.
- week.
- day.
- week_day.
- week_day_name.
- crime_hour.
- crime_lon.
- crime_lat.
- atm_dist.
- bank_dist.
- bar_dist.
- cafe_dist.
- industrial_dist.
- market_dist.
- nightclub_dist.
- police_dist.
- pub_dist.
- restaurant_dist.
- taxi_dist.
- x.
- y.
- t.

Author(s)

Nicoletta D'Angelo

Examples

```
data(valenciacrimes)
```

valencianet *Roads of Valencia, Spain*

Description

A linear network of class `linnet` of the roads of Valencia, Spain

Usage

```
data(valencianet)
```

Format

A linear network of class `linnet`

Author(s)

Nicoletta D'Angelo

Examples

```
data(valencianet)
```

Index