Package ‘NScluster’

September 18, 2012

Version 1.0.0

Title Simulation and Estimation of the Neyman-Scott Type Spatial Cluster Models

Author The Institute of Statistical Mathematics, based on the program by Ushio Tanaka

Maintainer Masami Saga <msaga@mtb.biglobe.ne.jp>

Depends R (>= 2.9.0)

Description This package simulate and estimate spatial clustered point patterns of the Neyman-Scott model and their extensions.

License GPL (>= 2)

BugReports ismrp@jasp.ism.ac.jp

R topics documented:

`NScluster-package` *Simulation and estimation of the Neyman-Scott type spatial cluster models*

Description

This package provides functions for simulation and estimation of spatial cluster point pattern of Neyman-Scott models and their extensions. We adopt the Simplex estimation to maximize the Palm likelihood function.

Details

The documentation 'NScluster: An R Package for Simulation and Estimation of the Neyman-Scott type spatial cluster models' is available in `../doc/NScluster-guide.pdf`.

PDF version of reference manual is available in `../doc/NScluster-manual.pdf`

Simulation :

`SimulateThomas`, `SimulateIP`, `SimulateTypeA`, `SimulateTypeB` and `SimulateTypeC` simulate spatial cluster point pattern of Neyman-Scott models and their extensions. We describe overview of those models briefly in the NScluster documentation `../doc/NScluster-guide.pdf`.

Simulation method of each model is described under the corresponding topic.

Parameter estimation :

We adopt the Simplex estimation to maximize the Palm likelihood function (or minimize the negative Palm likelihood function). The $maximum\, Palm\, likelihood\, estimators$ are called MPLEs, for short. The Palm intensity function and the analytical form of the Palm log-likelihood of the Tomas model, Type B model and Type C model are described under the topic `SimplexThomas`, `SimplexTypeB` and `SimplexTypeC`, respectively. On the other hand, for `SimplexIP` and `SimplexTypeA`, we need to take the alternative form without explicit representation of the Palm intensity function, which need very long c.p.u. time in the minimization procedure.

`PalmThomas`, `PalmIP`, `PalmTypeA`, `PalmTypeB` and `PalmTypeC` calculate the non-parametric Palm intensity function estimated directory from a set of point pattern data.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44, The Institute of Statistical Mathematics, Tokyo. `http://www.ism.ac.jp/editsec/csm/index.html`

U.Tanaka, Y. Ogata and D. Stoyan, Parameter estimation and model selection for Neyman-Scott point processes, *Biometrical Journal*, **50**, 2008, 43-57.

PalmIP *Non-Parametric Estimate of The Palm Intensity of The Inverse-Power Type*

Description

Calculate the non-parametric Palm intensity function of the inverse-power type estimated directly from a set of point pattern data.

Usage

```
PalmIP(offspring, pa, delta, Ty, x2, plot=TRUE)
```

Arguments

`offspring`	the list of coordinates (x, y) of offspring points.
`pa`	the parameters (μ_i, ν_i, p_i, c_i), $i = 1, 2, ..., I \leq 7$.
`delta`	a width for the non-parametric Palm intensity function.
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`x2`	upper limit value in place of ∞.
`plot`	logical. If TRUE (default) the non-parametric estimate and the curves of the true and MPLE (the maximum Palm likelihood estimator) parameters are shown.

Value

`r`	the distance $r = j\Delta$, where $j = 1, 2, ..., [R/\Delta]$, where [ ] is the Gauss' symbol and $R = 1/2$ is given in the program for the normalized rectangular region for the point pattern.
`np.palm`	the corresponding values of the non-parametric Palm intensity function of r, which is normalized by the total intensity estimate of the point pattern data.
`palm.normal`	the normalized Palm intensity functions $\lambda_0(r)/\hat{\lambda}$ calculated from the given sets of parameter values (μ_i, ν_i, p_i, c_i).

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
## simulation
seeds <- c(822, 913, 905)
pa1 <- c(50, 30, 1.5, 0.005)
Ty <- 1
z <- SimulateIP( seeds, pa1, Ty )

## estimation
delta <- 0.001
x2 <- 0.3
pa2 <- c(0.5651715e+02, 0.2392856e+02, 0.1600531e+01, 0.5565606e-02)
pa <- matrix(c(pa1,pa2), 2, 4, byrow=TRUE)
PalmIP( z$offspring, pa, delta, Ty, x2 )
```

`PalmThomas` *Non-Parametric Estimate of The Palm Intensity of The Thomas Model*

Description

Calculate the non-parametric Palm intensity function of the Thomas model estimated directly from a set of point pattern data.

Usage

```
PalmThomas(offspring, pa, delta, Ty, plot=TRUE)
```

Arguments

`offspring`	the list of coordinates (x, y) of offspring points.
`pa`	the parameters (μ_i, ν_i, σ_i), $i = 1, 2, ..., I \leq 8$.
`delta`	a width for the non-parametric Palm intensity function.
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`plot`	logical. If TRUE (default) the non-parametric estimate and the curves of the true and MPLE (the maximum Palm likelihood estimator) parameters are shown.

Value

`r`	the distance $r = j\Delta$, where $j = 1, 2, ..., [R/\Delta]$, where [ ] is the Gauss' symbol and $R = 1/2$ is given in the program for the normalized rectangular region for the point pattern.
`np.palm`	the corresponding values of the non-parametric Palm intensity function of r, which is normalized by the total intensity estimate of the point pattern data.
`palm.normal`	the normalized Palm intensity functions $\lambda_0(r)/\hat{\lambda}$ calculated from the given sets of parameter values (μ_i, ν_i, σ_i).

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
## simulation
seeds <- c(822, 913, 905)
pa1 <- c(50, 30, 0.03)
Ty <- 1
z <- SimulateThomas( seeds, pa1, Ty )

## estimation
delta <- 0.001
pa2 <- c(0.51311e+02, 0.26104e+02, 0.29071e-01)
pa <- matrix(c(pa1,pa2), 2, 3, byrow=TRUE)
PalmThomas( z$offspring, pa, delta, Ty )
```

`PalmTypeA` *Non-Parametric Estimate of The Palm Intensity of Type A Model*

Description

Calculate the non-parametric Palm intensity function of Type A Model estimated directly from a set of point pattern data.

Usage

```
PalmTypeA(offspring, pa, delta, Ty, x2, plot=TRUE)
```

Arguments

`offspring`	the list of coordinates (x, y) of offspring points.
`pa`	the parameters $(\mu_i, \nu_i, a_i, \sigma_{1,i}, \sigma_{2,i})$, $i = 1, 2, ..., I \leq 7$.
`delta`	a width for the non-parametric Palm intensity function.
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`x2`	upper limit value in place of ∞.
`plot`	logical. If TRUE (default) the non-parametric estimate and the curves of the true and MPLE (the maximum Palm likelihood estimator) parameters are shown.

Value

- `r` — the distance $r = j\Delta$, where $j = 1, 2, ..., [R/\Delta]$, where [ ] is the Gauss' symbol and $R = 1/2$ is given in the program for the normalized rectangular region for the point pattern.
- `np.palm` — the corresponding values of the non-parametric Palm intensity function of r, which is normalized by the total intensity estimate of the point pattern data.
- `palm.normal` — the normalized Palm intensity functions $\lambda_0(r)/\hat{\lambda}$ calculated from the given sets of parameter values $(\mu_i, \nu_i, a_i, \sigma_{1,i}, \sigma_{2,i})$.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
## simulation
seeds <- c(822, 913, 905)
pa1 <- c(50, 30, 0.3, 0.005, 0.1)
Ty <- 1
z <- SimulateTypeA( seeds, pa1, Ty, 100, 150 )

## estimation
delta <- 0.001
x2 <- 0.3
pa2 <- c(0.5639245e+02, 0.2356894e+02, 0.3549922, 0.5203741e-02, 0.1070969)
pa <- matrix(c(pa1,pa2), 2, 5, byrow=TRUE)
PalmTypeA( z$offspring, pa, delta, Ty, x2 )
```

`PalmTypeB` *Non-Parametric Estimate of The Palm Intensity of Type B Model*

Description

Calculate the non-parametric Palm intensity function of Type B Model estimated directly from a set of point pattern data.

Usage

```
PalmTypeB(offspring, pa, delta, Ty, plot=TRUE)
```

Arguments

`offspring` the list of coordinates (x, y) of offspring points.

`pa` the parameters $(\mu_i, \nu_i, a_i, \sigma_{1,i}, \sigma_{2,i})$, $i = 1, 2, ..., I \leq 8$.

`delta` a width for the non-parametric Palm intensity function.

`Ty` the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.

`plot` logical. If TRUE (default) the non-parametric estimate and the curves of the true and MPLE (the maximum Palm likelihood estimator) parameters are shown.

Value

`r` the distance $r = j\Delta$, where $j = 1, 2, ..., [R/\Delta]$, where [ ] is the Gauss' symbol and $R = 1/2$ is given in the program for the normalized rectangular region for the point pattern.

`np.palm` the corresponding values of the non-parametric Palm intensity function of r, which is normalized by the total intensity estimate of the point pattern data.

`palm.normal` the normalized Palm intensity functions $\lambda_0(r)/\hat{\lambda}$ calculated from the given sets of parameter values $(\mu_i, \nu_i, a_i, \sigma_{1,i}, \sigma_{2,i})$.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
## simulation
seeds <- c(822, 913, 905)
mu1 <- 10; mu2 <- 40; nu <- 30; sig1 <- 0.01; sig2 <- 0.03
Ty <- 1
z <- SimulateTypeB( seeds, c(mu1,mu2,nu,sig1,sig2 ), Ty, 100, 150 )

## estimation
delta <- 0.001
pa1 <- c(50, 30, 0.2, 0.01, 0.03)
pa2 <- c(0.44100e+02, 0.30691e+02, 0.15119, 0.94494e-02, 0.26822e-01)
pa <- matrix(c(pa1,pa2), 2, 5, byrow=TRUE)
PalmTypeB( z$offspring, pa, delta, Ty )
```

PalmTypeC *Non-Parametric Estimate of The Palm Intensity of Type C Model*

Description

Calculate the non-parametric Palm intensity function of Type C Model estimated directly from a set of point pattern data.

Usage

```
PalmTypeC(offspring, pa, delta, Ty, plot=TRUE)
```

Arguments

`offspring`	the list of coordinates (x, y) of offspring points.
`pa`	the parameters $(\lambda_i, \nu_{1,i}, a_i, \sigma_{1,i}, \sigma_{2,i})$, $i = 1, 2, ..., I \leq 8$.
`delta`	a width for the non-parametric Palm intensity function.
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`plot`	logical. If TRUE (default) the non-parametric estimate and the curves of the true and MPLE (the maximum Palm likelihood estimator) parameters are shown.

Value

`r`	the distance $r = j\Delta$, where $j = 1, 2, ..., [R/\Delta]$, where [ ] is the Gauss' symbol and $R = 1/2$ is given in the program for the normalized rectangular region for the point pattern.
`np.palm`	the corresponding values of the non-parametric Palm intensity function of r, which is normalized by the total intensity estimate of the point pattern data.
`palm.normal`	the normalized Palm intensity functions $\lambda_0(r)/\hat{\lambda}$ calculated from the given sets of parameter values $(\lambda_i, \nu_{1,i}, a_i, \sigma_{1,i}, \sigma_{2,i})$.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
## simulation
seeds <- c(822, 913, 905)
mu1 <- 5;  nu1 <- 30;  sig1 <- 0.01
mu2 <- 9;  nu2 <- 150;  sig2 <- 0.05
Ty <- 1
z <- SimulateTypeC( seeds, c(mu1,nu1,sig1), c(mu2,nu2,sig2), Ty, 200, 300 )
```

```
## estimation
delta <- 0.001
pa1 <- c(1500, 30, 0.1, 0.01, 0.05)
pa2 <- c(0.1281e+04, 0.43134e+02, 0.30049, 0.14926e-01, 0.51719e-01)
pa <- matrix(c(pa1,pa2), 2, 5, byrow=TRUE)
PalmTypeC( z$offspring, pa, delta, Ty )
```

SimplexIP *Parameter estimation of The Inverse-Power Type*

Description

Parameter estimation of the inverse-power type via numerical calculation of the Ripley's K-function.

Usage

```
SimplexIP(offspring, pa, Ty=1, x2, skip=1, eps=0.1e-2, process=0, plot=TRUE)
```

Arguments

`offspring`	the list of coordinates (x, y) of offspring points.
`pa`	the initial guesses of the parameters (μ, ν, p, c).
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`x2`	upper limit value in place of ∞ in the integral in distribution function below.
`skip`	the variable for the fast likelihood but rough approximation of the initial estimates. The `skip` calculate the Palm intensity function in the log-likelihood function for every `skip`-th r_{ij} in the ordered distances of the pairs i and j.
`eps`	the optimization procedure is iterated at most 1000 times until `stderr` becomes smaller than `eps`.
`process`	repot the process of minimizing. Allowed values are

0 : no report.
1 : output the process of minimizing the negative Palm log-likelihood function until the values converge to the MPLE values for given data.
2 : output the process of optimizing by the simplex with the normalized parameters depending on `pa`. The actual estimates are obtained by the indicated x-values times `pa`.
3 : output the both processes.

`plot`	plot the process of optimizing by the simplex with the normalized parameters depending on `pa`.

Value

logL.p	the minimized -log L in the process of minimizing the negative Palm log-likelihood function.
mple	the MPLE (maximum Palm likelihood estimator) values corresponding to the above logL.p.
logL.s	the minimized -log L by the simplex method.
stderr	the standard deviations among function values of the vertices $x(i)$ of the simplex.
pa.normal	the values of the corresponding normalized parameters $x(i)$. the last values show the normalized variables corresponding to the MPLE (the maximum Palm likelihood estimator) values relative to the initial estimates.

Note

SimplexIP and SimplexTypeA have to use numerical integration and difference to compute the functions, which need very long c.p.u. time in the minimization procedure.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
  ## simulation
  seeds <- c(822, 913, 905)
  mu <- 50;  nu <- 30;  p <- 1.5;  c <- 0.005
  z <- SimulateIP( seeds, c(mu,nu,p,c) )

  ## Not run:
    ## estimation
    ## need very long c.p.u time in the minimization procedure
    SimplexIP( z$offspring, c(mu,nu,p,c), x2=0.3, skip=100, process=3 )

## End(Not run)
```

SimplexThomas *Parameter estimation of The Thomas Model*

Description

Parameter estimation of the Thomas model by using the Palm log-likelihood function.

Usage

```
SimplexThomas(offspring, pa, Ty=1, eps=0.1e-2, process=0, plot=TRUE)
```

Arguments

`offspring` the list of coordinates (x, y) of offspring points.

`pa` the initial guesses of the parameters (μ, ν, σ).

`Ty` the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.

`eps` the optimization procedure is iterated at most 1000 times until `stderr` becomes smaller than `eps`.

`process` repot the process of minimizing. Allowed values are

0 : no report.
1 : output the process of minimizing the negative Palm log-likelihood function until the values converge to the MPLE values for given data.
2 : output the process of optimizing by the simplex with the normalized parameters depending on `pa`. The actual estimates are obtained by the indicated x-values times `pa`.
3 : output the both processes.

`plot` plot the process of optimizing by the simplex with the normalized parameters depending on `pa`.

Details

The Palm intensity function of the Thomas model is calculated as follows:

For any $r \geq 0$,

$$\lambda_0(r) = \mu\nu + \frac{\nu}{4\pi\sigma^2} \exp\left(-\frac{r^2}{4\sigma^2}\right).$$

The Palm log-likelihood function of the Thomas model is analytically calculated as follows:

$$\log L(\mu, \nu, \sigma) = \sum_{\{i,j;i \neq j, r_{ij} < R\}} \log \nu \left\{\mu + \frac{1}{4\pi\sigma^2} \exp\left(-\frac{r_{ij}^2}{4\sigma^2}\right)\right\}$$

$$-N(W)\nu \left\{\pi\mu R^2 + 1 - \exp\left(-\frac{R^2}{4\sigma^2}\right)\right\},$$

with $R = 1/2$ which means the half of the t_x (TX) in side length of the normalized rectangular.

Value

`logL.p` the minimized -log L in the process of minimizing the negative Palm log-likelihood function.

`mple` the MPLE (maximum Palm likelihood estimator) values corresponding to the above `logL.p`.

`logL.s` the minimized -log L by the simplex method.

`stderr` the standard deviation.

`pa.normal` the normalized variables corresponding to the MPLE values relative to the initial estimates.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
## simulation
seeds <- c(822, 913, 905)
mu <- 50;  nu <- 30;  sig <- 0.03
z <- SimulateThomas( seeds, c(mu,nu,sig) )

## estimation
SimplexThomas( z$offspring, c(mu,nu,sig), process=3 )
```

`SimplexTypeA` *Parameter estimation of Type A model*

Description

Parameter estimation of type A model via numerical calculation of the Ripley's K-function.

Usage

```
SimplexTypeA(offspring, pa, Ty=1, x2, skip=1, eps=0.1e-2, process=0, plot=TRUE)
```

Arguments

`offspring`	the list of coordinates (x, y) of offspring points.
`pa`	the initial guesses of the parameters $(\mu, \nu, a, \sigma_1, \sigma_2)$.
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`x2`	upper limit value in place of ∞ in the integral in the distribution function.
`skip`	the variable for the fast likelihood but rough approximation of the initial estimates. The `skip` calculate the Palm intensity function in the log-likelihood unction for every `skip`-th r_{ij} in the ordered distances of the pairs i and j.
`eps`	the optimization procedure is iterated at most 1000 times until `stderr` becomes smaller than `eps`.
`process`	repot the process of minimizing. Allowed values are

0 : no report.
1 : output the process of minimizing the negative Palm log-likelihood function until the values converge to the MPLE values for given data.
2 : output the process of optimizing by the simplex with the normalized parameters depending on `pa`. The actual estimates are obtained by the indicated x-values times `pa`.
3 : output the both processes.

`plot`	plot the process of optimizing by the simplex with the normalized parameters depending on `pa`.

Value

`logL.p`	the minimized -log L in the process of minimizing the negative Palm log-likelihood function.
`mple`	the MPLE (maximum Palm likelihood estimator) values corresponding to the above `logL.p`.
`logL.s`	the minimized -log L by the simplex method.
`stderr`	the standard deviations.
`pa.normal`	the normalized variables corresponding to the MPLE values relative to the initial estimates.

Note

`SimplexIP` and `SimplexTypeA` have to use numerical integration and difference to compute the functions, which need very long c.p.u. time in the minimization procedure.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
  ## simulation
  seeds <- c(822, 913, 905)
  mu <- 50;  nu <- 30;  a <- 0.3;  sig1 <- 0.005;  sig2 <- 0.1
  z <- SimulateTypeA( seeds, c(mu,nu,a,sig1,sig2), pmax=100, omax=150 )

  ## Not run:
    ## estimation
    ## need very long c.p.u time in the minimization procedure
    SimplexTypeA( z$offspring, c(mu,nu,a,sig1,sig2), x2=0.3, skip=1000, process=3 )

## End(Not run)
```

`SimplexTypeB` *Parameter estimation of Type B model*

Description

Parameter estimation of Type B Model by using The Palm Log-Likelihood Function.

Usage

```
SimplexTypeB(offspring, pa, Ty=1, eps=0.1e-2, process=0, plot=TRUE)
```

Arguments

`offspring` the list of coordinates (x, y) of offspring points.

`pa` the initial guesses of the parameters $(\mu_1, \mu_2, \nu, \sigma_1, \sigma_2)$.

`Ty` the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.

`eps` the optimization procedure is iterated at most 1000 times until `stderr` becomes smaller than `eps`.

`process` repot the process of minimizing. Allowed values are

0 : no report.
1 : output the process of minimizing the negative Palm log-likelihood function until the values converge to the MPLE values for given data.
2 : output the process of optimizing by the simplex with the normalized parameters depending on `pa`. The actual estimates are obtained by the indicated x-values times `pa`.
3 : output the both processes.

`plot` plot the process of optimizing by the simplex with the normalized parameters depending on `pa`.

Details

The Palm intensity function of the Type B model is calculated as follows: For any $r \geq 0$,

$$\lambda_0(r) = \lambda + \frac{\nu}{4\pi} \left\{ \frac{a}{\sigma_1^2} \exp\left(-\frac{r^2}{4\sigma_1^2}\right) + \frac{(1-a)}{\sigma_2^2} \exp\left(-\frac{r^2}{4\sigma_2^2}\right) \right\},$$

where $\lambda = \nu(\mu_1 + \mu_2)$ is the total population size and $a = \mu_1/(\mu_1 + \mu_2)$ is the ratio of the parent points of the smaller sized cluster to the total ones.

The Palm log-likelihood function of the Type B model is analytically calculated as follows:

$$\log L(\lambda, \alpha, \beta, \sigma_1, \sigma_2)$$

$$= \sum_{\{i,j;i\neq j,r_{ij}<R\}} \log\left[\lambda + \frac{1}{4\pi}\left\{\frac{\alpha}{\sigma_1^2}\exp\left(-\frac{r_{ij}^2}{4\sigma_1^2}\right) + \frac{\beta}{\sigma_2^2}\exp\left(-\frac{r_{ij}^2}{4\sigma_2^2}\right)\right\}\right]$$

$$-N(W)\left[\pi R^2\lambda + \alpha\left\{1 - \exp\left(-\frac{R^2}{4\sigma_1^2}\right)\right\} + \beta\left\{1 - \exp\left(-\frac{R^2}{4\sigma_2^2}\right)\right\}\right],$$

with $R = 1/2$, where $\alpha = a\nu$ and $\beta = (1-a)\nu$.

Value

`logL.p`	the minimized -log L in the process of minimizing the negative Palm log-likelihood function.
`mple`	the MPLE (maximum Palm likelihood estimator) values corresponding to the above `logL.p`.
`logL.s`	the minimized -log L by the simplex method.
`stderr`	the standard deviations.
`pa.normal`	the normalized variables corresponding to the MPLE values relative to the initial estimates.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
  ## simulation
  seeds <- c(822, 913, 905)
  mu1 <- 10;  mu2 <- 40; nu <- 30;  sig1 <- 0.01;  sig2 <- 0.03
  z <- SimulateTypeB( seeds, c(mu1,mu2,nu,sig1,sig2), pmax=100, omax=150 )

  ## Not run:
    ## estimation
    ## need very long c.p.u time in the minimization procedure
    SimplexTypeB( z$offspring, c(mu1,mu2,nu,sig1,sig2), process=3 )

## End(Not run)
```

`SimplexTypeC` *Parameter estimation of Type C model*

Description

Parameter estimation of Type C Model by using The Palm Log-Likelihood Function.

Usage

```
SimplexTypeC(offspring, pa1, pa2, Ty=1, eps=0.1e-2, process=0, plot=TRUE)
```

Arguments

offspring	the list of coordinates (x, y) of offspring points.
pa1	the initial guesses of the parameters (μ_1, ν_1, σ_1).
pa2	the initial guesses of the parameters (μ_2, ν_2, σ_2).
Ty	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
eps	the optimization procedure is iterated at most 1000 times until `stderr` becomes smaller than `eps`.
process	repot the process of minimizing. Allowed values are

0 : no report.
1 : output the process of minimizing the negative Palm log-likelihood function until the values converge to the MPLE values for given data.
2 : output the process of optimizing by the simplex with the normalized parameters depending on `pa`. The actual estimates are obtained by the indicated x-values times `pa`.
3 : output the both processes.

plot	plot the process of optimizing by the simplex with the normalized parameters depending on `pa`.

Details

The Palm intensity function of the Type C model is calculated as follows: For any $r \geq 0$,

$$\lambda_0(r) = \lambda + \frac{1}{4\pi}\left\{\frac{a\nu_1}{\sigma_1^2}\exp\left(-\frac{r^2}{4\sigma_1^2}\right) + \frac{(1-a)\nu_2}{\sigma_2^2}\exp\left(-\frac{r^2}{4\sigma_2^2}\right)\right\},$$

where $\lambda = \mu_1\nu_1 + \mu_2\nu_2$ is the total population size and $a = \mu_1\nu_1/\lambda$ is the ratio of the all offspring points of smaller sized cluster to the total population size.

The Palm log-likelihood function of the Type C model is analytically calculated as follows:

$$\log L(\lambda, \alpha, \beta, \sigma_1, \sigma_2)$$

$$= \sum_{\{i,j;i\neq j,r_{ij}<R\}} \log\left[\lambda + \frac{1}{4\pi}\left\{\frac{\alpha}{\sigma_1^2}\exp\left(-\frac{r_{ij}^2}{4\sigma_1^2}\right) + \frac{\beta}{\sigma_2^2}\exp\left(-\frac{r_{ij}^2}{4\sigma_2^2}\right)\right\}\right]$$

$$-N(W)\left[\pi R^2\lambda + \alpha\left\{1 - \exp\left(-\frac{R^2}{4\sigma_1^2}\right)\right\} + \beta\left\{1 - \exp\left(-\frac{R^2}{4\sigma_2^2}\right)\right\}\right],$$

with $R = 1/2$, where $\alpha = a\nu_1$ and $\beta = (1-a)\nu_2$.

Value

`logL.p`	the minimized -log L in the process of minimizing the negative Palm log-likelihood function.
`mple`	the MPLE (maximum Palm likelihood estimator) values corresponding to the above `logL.p`.
`logL.s`	the minimized -log L by the simplex method.
`stderr`	the standard deviations.
`pa.normal`	the normalized variables corresponding to the MPLE values relative to the initial estimates.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
  ## simulation
  seeds <- c(822, 913, 905)
  mu1 <- 5;  nu1 <- 30;  sig1 <- 0.01
  mu2 <- 9;  nu2 <- 150;  sig2 <- 0.05
  z <- SimulateTypeC( seeds, c(mu1,nu1,sig1), c(mu2,nu2,sig2), pmax=200, omax=300 )

  ## Not run:
    ## estimation
    ## need long c.p.u time in the minimization procedure
    SimplexTypeC( z$offspring, c(mu1,nu1,sig1), c(mu2,nu2,sig2), process=3 )

## End(Not run)
```

SimulateIP *Simulation of the Inverse-Power Type Model*

Description

Simulation of the Inverse-Power Type Model.

Usage

```
SimulateIP(seeds, pa, Ty=1, pmax=100, omax=3000, plot=TRUE)
```

Arguments

`seeds`	the three positive integer variables, which are initial seeds for a sequence of uniform random numbers.
`pa`	the parameter values (μ, ν, p, c).
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`pmax`	maximum number of parent points.
`omax`	maximum number of offspring points.
`plot`	logical. If TRUE (default) simulated parent points and offspring points are plotted.

Details

Let random variable U be independently and uniformly distributed in [0,1]. For any $r \geq 0$,

$$Q_{p,c}(r) := \int_0^r q_{p,c}(t)dt$$
$$= c^{p-1}(p-1)\frac{(r+c)^{1-p} - c^{1-p}}{1-p}$$
$$= 1 - c^{p-1}(r+c)^{1-p}.$$

Here, we put $Q_{p,c}(r) = U$. From this, we have

$$r = c\{(1-U)^{1/(1-p)} - 1\}.$$

Similarly, coordinate of the offspring points $(x_j^i, y_j^i), j = 1, 2, ..., Poisson(\nu)$ with the inverse-power type is given for each $i = 1, 2, ..., Poisson(\mu)$,

$$x_j^i = x_i^p + r\cos(2\pi U),$$
$$y_j^i = y_i^p + r\sin(2\pi U),$$

using series of different uniform random number $\{U\}$ for different i and j.

Value

`n.parents`	the number of simulated parent points.
`parents`	the coordinates of simulated parent points.
`n.offspring`	the total number of simulated offspring points.
`offspring`	the coordinates of simulated offspring points.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
seeds <- c(822, 913, 905)
mu <- 50;  nu <- 30;  p <- 1.5;  c <- 0.005
SimulateIP( seeds, c(mu,nu,p,c) )
```

`SimulateThomas` *Simulation of the Thomas Model*

Description

Simulation of the Thomas Model.

Usage

```
SimulateThomas(seeds, pa, Ty=1, pmax=100, omax=3000, plot=TRUE)
```

Arguments

`seeds`	the three positive integer variables, which are initial seeds for a sequence of uniform random numbers.
`pa`	the parameter values (μ, ν, σ).
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0,1] \times [0, T_y]$ without loss of generality except for the scaling.
`pmax`	maximum number of parent points.
`omax`	maximum number of offspring points.
`plot`	logical. If TRUE (default) simulated parent points and offspring points are plotted.

Details

Let random variable U be independently and uniformly distributed in [0,1]. We put

$$U = \int_0^r q_\sigma(t)dt = 1 - \exp\left(-\frac{r^2}{2\sigma^2}\right).$$

Then we have

$$r = \sigma\sqrt{-2\log(1-U)}.$$

Let $(x_i^p, y_i^p), i = 1, 2, ..., I$ be a coordinate of each parent point where the integer I is generated from the Poisson random variable $Poisson(\mu)$ with mean μ from now on. Then, for each i, the number of offspring J_i is generated by the random variable $Poisson(\nu)$ with mean ν. Then, using series of different uniform random numbers $\{U\}$ for different i and j, the offspring coordinates $(x_j^i, y_j^i), j = 1, 2, ..., J_i$ is given by

$$x_j^i = x_i^p + r\cos(2\pi U),$$

$$y_j^i = y_i^p + r\sin(2\pi U),$$

owing to the isotropy condition of the distribution.

Given a positive number ν and let a sequence of a random variable $\{U_k\}$ be independently and uniformly distributed in [0,1], the Poisson random number M is the smallest integer such that

$$\sum_{k=1}^{M+1} -\log U_k > \nu,$$

where $\log$ represents natural logarithm.

Value

`n.parents`	the number of simulated parent points.
`parents`	the coordinates of simulated parent points.
`n.offspring`	the total number of simulated offspring points.
`offspring`	the coordinates of simulated offspring points.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
seeds <- c(822, 913, 905)
mu <- 50;  nu <- 30;  sig <- 0.03
SimulateThomas( seeds, c(mu,nu,sig) )
```

SimulateTypeA — *Simulation of the Generalized Thomas Model of Type A*

Description

Simulation of the Generalized Thomas Model of Type A.

Usage

```
SimulateTypeA(seeds, pa, Ty=1, pmax=100, omax=3000, plot=TRUE)
```

Arguments

`seeds`	the three positive integer variables, which are initial seeds for a sequence of uniform random numbers.
`pa`	the parameter values $(\mu, \nu, a, \sigma_1, \sigma_2)$ for the random variable Poisson.
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0,1] \times [0, T_y]$ without loss of generality except for the scaling.
`pmax`	maximum number of parent points.
`omax`	maximum number of offspring points.
`plot`	logical. If TRUE (default) simulated parent points and offspring points are plotted.

Details

Parents' configuration and numbers of the offspring cluster sizes are generated by the same way as the Thomas model.

Let random variable $U_k, k = 1, 2$ be independently and uniformly distributed in [0,1]. Then r satisfies as follows:

$$r = \sigma_1 \sqrt{-2\log(1 - U_1)},\ U_2 \leq a,$$
$$r = \sigma_2 \sqrt{-2\log(1 - U_1)},\ otherwise.$$

Then, by the isotropy condition, for $i = 1, 2, ..., Poisson(\mu)$, coordinate of the offspring points $(x_j^i, y_j^i), j = 1, 2, ..., Poisson(\nu)$ is given for each $i = 1, ..., Poisson(\mu)$,

$$x_j^i = x_i^p + r\cos(2\pi U),$$
$$y_j^i = y_i^p + r\sin(2\pi U),$$

using series of different uniform random numbers $\{U_1, U_2, U\}$ for different i and j.

Value

`n.parents`	the number of simulated parent points.
`parents`	the coordinates of simulated parent points.
`n.offspring`	the total number of simulated offspring points.
`offspring`	the coordinates of simulated offspring points.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
seeds <- c(822, 913, 905)
mu <- 50;  nu <- 30;  a <- 0.3;  sig1 <- 0.005;  sig2 <- 0.1
SimulateTypeA( seeds, c(mu,nu,a,sig1,sig2), pmax=100, omax=150 )
```

`SimulateTypeB` *Simulation of the Generalized Thomas Model of Type B*

Description

Simulation of the Generalized Thomas Model of Type B.

Usage

```
SimulateTypeB(seeds, pa, Ty=1, pmax=100, omax=3000, plot=TRUE)
```

Arguments

`seeds`	the three positive integer variables, which are initial seeds for a sequence of uniform random numbers.
`pa`	the parameter values $(\mu_1, \mu_2, \nu, \sigma_1, \sigma_2)$ for the random variable Poisson.
`Ty`	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
`pmax`	maximum number of parent points.
`omax`	maximum number of offspring points.
`plot`	logical. If TRUE (default) simulated parent points and offspring points are plotted.

Details

Consider the two type of the Thomas model with parameters (μ_1, ν, σ_1) and (μ_2, ν, σ_2). Parents' configuration and numbers of the offspring cluster sizes are generated by the two types of uniformly distributed parents (x_i^k, x_i^k) with $i = 1, 2, ..., Poisson(\mu_k)$ for $k = 1, 2$, respectively.

Then, using series of different uniform random numbers $\{U\}$ for different i and j, the offspring coordinates $(x_j^{k,i}, y_j^{k,i})$ of the parents (k, i) with $k = 1, 2$ and $j = 1, 2, ..., Poisson(\nu)$ is given by

$$x_j^{k,i} = x_i^k + r_k \cos(2\pi U),$$

$$y_j^{k,i} = y_i^k + r_k \sin(2\pi U),$$

where

$$r_k = \sigma_k \sqrt{-2 \log(1 - U_k)},\ k = 1, 2,$$

with different random numbers $\{U_k, U\}$ for different k, i, and j

Value

n.parents1	the number of simulated parent points with μ_1.
n.offspring1	the total number of simulated offspring points with μ_1.
n.parents2	the number of simulated parent points with μ_2.
n.offspring2	the total number of simulated offspring points with μ_2.
parents	the coordinates of simulated parent points with μ_1 and μ_2.
offspring	the coordinates of simulated offspring points with $\mu_1, \mu_2, \nu, \sigma_1, \sigma_2$.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
seeds <- c(822, 913, 905)
mu1 <- 10; mu2 <- 40; nu <- 30; sig1 <- 0.01; sig2 <- 0.03
SimulateTypeB( seeds, c(mu1,mu2,nu,sig1,sig2), pmax=100, omax=150 )
```

SimulateTypeC — *Simulation of the Generalized Thomas Model of Type C*

Description

Simulation of the Generalized Thomas Model of Type C.

Usage

```
SimulateTypeC(seeds, pa1, pa2, Ty=1, pmax=100, omax=3000, plot=TRUE)
```

Arguments

seeds	the three positive integer variables, which are initial seeds for a sequence of uniform random numbers.
pa1	the parameter values (μ_1, ν_1, σ_1) for the random variable Poisson.
pa2	the parameter values (μ_2, ν_2, σ_2) for the random variable Poisson.
Ty	the variable for the standardized coordinates of points in the rectangular region $[0, 1] \times [0, T_y]$ without loss of generality except for the scaling.
pmax	maximum number of parent points.
omax	maximum number of offspring points.
plot	logical. If TRUE (default) simulated parent points and offspring points are plotted.

Details

Consider the two type of the Thomas model with parameters (μ_1, ν_1, σ_1) and (μ_2, ν_2, σ_2). Parents' configuration and numbers of the offspring cluster sizes are generated by the two types of uniformly distributed parents (x_i^k, x_i^k) with $i = 1, 2, ..., Poisson(\mu_k)$ for k=1,2, respectively.

Then, using series of different uniform random numbers $\{U\}$ for different i and j, the offspring coordinates $(x_j^{k,i}, y_j^{k,i}), j = 1, 2, ..., Poisson(\nu_k)$ of the parents (k, i) with $k = 1, 2$ is given by

$$x_j^{k,i} = x_i^k + r_k \cos(2\pi U),$$

$$y_j^{k,i} = y_i^k + r_k \sin(2\pi U),$$

where

$$r_k = \sigma_k \sqrt{-2\log(1 - U_k)},\ k = 1, 2,$$

with different random numbers $\{U_k, U\}$ for different k, i and j.

Value

`n.parents1`	the number of simulated parent points with μ_1.
`parents1`	the coordinates of simulated parent points with μ_1.
`n.offspring1`	the total number of simulated offspring points with μ_1.
`offspring1`	the coordinates of simulated offspring points with μ_1.
`n.parents2`	the number of simulated parent points with μ_2.
`parents2`	the coordinates of simulated parent points μ_2.
`n.offspring2`	the total number of simulated offspring points μ_2.
`offspring2`	the coordinates of simulated offspring points μ_2.

References

U. Tanaka, Y. Ogata and K. Katsura, Simulation and estimation of the Neyman-Scott type spatial cluster models, *Computer Science Monographs* **No.34**, 2008, 1-44. The Institute of Statistical Mathematics.

Examples

```
seeds <- c(822, 913, 905)
mu1 <- 5;  nu1 <- 30;  sig1 <- 0.01
mu2 <- 9;  nu2 <- 150;  sig2 <- 0.05
SimulateTypeC( seeds, c(mu1,nu1,sig1), c(mu2,nu2,sig2), pmax=200, omax=300 )
```

Index