Package 'mSTEM'

October 2, 2019

Type Package

Title Multiple Testing of Local Extrema for Detection of Change Points

Version 1.0-1

Date 2019-09-1

Author Zhibing He and Dan Cheng

Maintainer Zhibing He <zhibingh@asu.edu>

Description A new approach to detect change points based on smoothing and multiple testing, which is for long data sequence modeled as piecewise constant functions plus stationary Gaussian noise, see Dan Cheng and Armin Schwartzman (2015) <arXiv:1504.06384>.

Depends R (>= 3.1.0)

Imports parallel, foreach, doParallel, latex2exp

URL https://arxiv.org/abs/1504.06384

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-10-02 10:00:05 UTC

R topics documented:

`ch.est` *Estimate* **s2**,**lambda2**,**lambda4**,**Delta**

Description

Estimate **s2**,**lambda2**,**lambda4**,**Delta**

Usage

```
ch.est(nu, gamma, size, B = 100)
```

Arguments

`nu`	bandwidth of Gaussian kernal applied to White-noise, Whitenoise error if `nu` = 0
`gamma`	bandwidth of nonparameter smoothing
`size`	sample size
`B`	Montelarlo iteration times

Value

a list of s2,lambda2,lambda4,Delta

References

Multiple Testing of Local Extrema for Detection of Change Points https://arxiv.org/abs/1504.06384

See Also

`which.cp`

Examples

```
ch.est(nu=2,gamma=4,size=1000,B=100)
```

conv — *Compute convolution function using FFT, similar to the function 'conv' in matlab*

Description

Compute convolution function using FFT, similar to the function 'conv' in matlab

Usage

```
conv(u, v, shape = c("same", "full"))
```

Arguments

u	vector
v	vector
shape	if 'same', return central part of the convolution,the same size as u; ortherwise return the whole sequence with size lenth(u)+length(v)-1

Value

a vector of convolution, as specified by shape.

References

Matlab document on 'conv' https://www.mathworks.com/help/matlab/ref/conv.html

Examples

```
u = c(-1,2,3,-2,0,1,2)
v = c(2,4,-1,1)
w = conv(u,v,'same')
```

Fdr — *Evaluate performance of estimated change points*

Description

Evaluate performance of estimated change points

Usage

```
Fdr(uh, b, th)
```

Arguments

uh	a vector of estimated change points locations
b	a scalar of location tolerance, specified by user
th	a vector of true change points locations

Value

a list of vector of FDR and Power

FDR	a scalar of fdr (false discovery rate)
Power	a scalar of power (true positive rate)

See Also

which.cp

Examples

```
Fdr(uh=c(7,15,32,47),b=4,th=c(10,20,30,40,50))
```

---

fdr.gam *Parallel computing fdr and power of change points estimation for different* gamma *and* nu

---

Description

Parallel computing fdr and power of change points estimation for different gamma and nu

Usage

```
fdr.gam(c, mu, Gamma, Nu, b, th, B = 100, level = 0.1, iter = 100)
```

Arguments

c	number of cpu cores used for parallel computing
mu	a vector of piecewise constant
Gamma	a vector of different gammas
Nu	a vector of different nus
b	a scalar of location tolerance, specified by user
th	a vector of true change points locations
B	Montelarlo iteration times
level	FDR control level
iter	iteration times for each combination of gamma and nu

Value

a list of matrix with the same length as Nu, FDR and Power for different Gamma are displayed within each matrix

Examples

```
size=12000
a = 1
A = a*(1:119)
H = seq(100,11900,100)
mu = GenMu(A,H,size=size)
z = GenZ(nu=2,size=size)
Gamma = seq(1,5,1)
Nu = seq(0,2,0.5)
model = fdr.gam(2,mu,Gamma,Nu,8,H,iter=100)
```

---

fdrBH *FDR threshold based on the Benjamini-Hochberg algorithm*

Description

FDR threshold based on the Benjamini-Hochberg algorithm

Usage

```
fdrBH(p, q)
```

Arguments

p	a vector of p-values
q	False Discovery Rate level

Value

p-value threshold based on independence or positive dependence

See Also

which.cp

Examples

```
fdrBH(seq(0.01,0.1,0.01),q=0.1)
```

GenDY *Generate first-order differential of a smoothed sequence Y*

Description

Generate first-order differential of a smoothed sequence Y

Usage

```
GenDY(mu, z, gamma)
```

Arguments

mu	a vector of piecewise constant
z	a vector of stationary Gaussian random error
gamma	bandwidth of nonparameter smoothing

Value

a vector of the differential of Y

See Also

GenMu/GenZ

Examples

```
mu = GenMu(x=1:10,pos=seq(10,100,10),size=150)
z = GenZ(nu=2,size=150)
GenDY(mu=mu,z=z,gamma=4)
```

GenMu *Generate a piecewise constant sequence starting from 0*

Description

Generate a piecewise constant sequence starting from 0

Usage

```
GenMu(x, pos, size)
```

Arguments

x	a vector containing all values of change points
pos	positions of change points, corresponding to x
size	sample size

Value

a piecewise constant sequence

See Also

GenDY

Examples

```
GenMu(x=1:10,pos=seq(10,100,10),size=150)
```

GenZ *Generate Gaussian autocorrelated random error sequence based on White-noise and Gaussian kernal*

Description

Generate Gaussian autocorrelated random error sequence based on White-noise and Gaussian kernal

Usage

```
GenZ(nu, size)
```

Arguments

nu	bandwidth of Gaussian kernal applied to White-noise, Whitenoise error if nu = 0
size	sample size

Value

a vector of random error

See Also

GenDY

Examples

```
GenZ(nu=2,size=1000)
```

`illu.plot` *Illustration plot of the procedure t0 detect change points*

Description

Illustration plot of the procedure t0 detect change points

Usage

```
illu.plot(mu, z, gamma, whichcp, b, Tmax, Tmin)
```

Arguments

`mu`	a vector of piecewise constant
`z`	a vector of stationary Gaussian random error
`gamma`	bandwidth of nonparameter smoothing
`whichcp`	output of the function `which.cp`
`b`	a scalar of location tolerance, specified by user
`Tmax`	a vector of true peak locations
`Tmin`	a vector true valley locations

Value

a figure plot showing detection of change points

Examples

```
set.seed(2019)
L = 1200
A = c(2.8,0,-2.4,0,-3,0.5,3,5,2,0)/1.5
Tmax = c(150,410,680,770,980)
Tmin = c(250,320,550,1000,1100)
H = c(150,250,320,410,550,680,770,980,1000,1100)
mu = GenMu(A,H,L); z = GenZ(nu=2,L)
y1 = GenDY(mu=mu,z=z,gamma=6)
chest = ch.est(nu=2,gamma=6,size=L,B=100)
chp= which.cp(y1,chest,level=0.1)
illu.plot(mu,z,gamma=6,chp,b=5,Tmax,Tmin)
```

`which.cp` *Find locations of change points*

Description

Find locations of change points

Usage

```
which.cp(y1, chest, level = 0.1)
```

Arguments

`y1`	a vector of the differential of sequence Y
`chest`	output of function `ch.est`
`level`	FDR control level

Value

a list of components

`peak`	a vector of peaks location
`vall`	a vector of valleys location
`pval`	a scalar of adjusted p-value based on FDR control
`thresh`	a scalar of threshold for `y1`

See Also

`ch.est`/`fdrBH`

Examples

```
mu = GenMu(x=1:10,pos=seq(10,100,10),size=150)
z = GenZ(nu=2,size=150)
y1 = GenDY(mu,z,gamma=4)
chest = ch.est(nu=2,gamma=8,size=150,B=100)
which.cp(y1,chest,level=0.1)
```

`which.peaks` *Find local maxima and minima in a sequence*

Description

Find local maxima and minima in a sequence

Usage

```
which.peaks(x, partial = FALSE, decreasing = FALSE)
```

Arguments

`x`	a vector with maxima or minima
`partial`	endpoints will be considered if 'true'
`decreasing`	find local minima if 'true', ortherwise local maxima

Value

a vector of positions of local maxima or minima

Examples

```
a = 100:1
which.peaks(a*sin(a/3))
```

Index