

Package ‘dipSFP’

May 5, 2010

Type Package

Title SFP detection based on dip statistic

Version 1.0

Date 2010-05-05

Author Na You

Maintainer Na You <nayou@ucr.edu>

Description Given a matrix of microarray data, SFPs are detected based on the dip statistic. The null distribution of dip statistic is estimated by smooth bootstrap algorithm. P values are adjusted using false discovery rate criteria for FDR control.

Depends diptest

License

LazyLoad yes

R topics documented:

c.gamma	2
c.lognorm	2
c.norm	3
c.t	3
dipSFP	4
FMbw	5
minimax	6
modes	7
sign.plus	8
Index	9

`c.gamma`*Factor 'c' for Gamma density*

Description

Compute the value of factor 'c' for Gamma density.

Usage

```
c.gamma(k, theta)
```

Arguments

<code>k</code>	shape parameter
<code>theta</code>	scale parameter

Value

The value of factor 'c'.

Author(s)

Na You

References

Cui et al. Single Feature Polymorphisms Detection using Recombinant Inbred Line microarray expression data.

`c.lognorm`*Factor 'c' for log-normal density*

Description

Compute the value of factor 'c' for log-normal density.

Usage

```
c.lognorm(mu, sigma)
```

Arguments

<code>mu, sigma</code>	parameters
------------------------	------------

Value

The value of factor 'c'.

Author(s)

Na You

References

Cui et al. Single Feature Polymorphpism Detection using Recombinant Inbred Line microarray expression data.

c.norm	<i>Factor 'c' for normal density</i>
--------	--------------------------------------

Description

Compute the value of factor 'c' for normal density.

Usage

```
c.norm(mu, sigma)
```

Arguments

mu	mean
sigma	standard deviation

Value

The value of factor 'c'.

Author(s)

Na You

References

Cui et al. Single Feature Polymorphpism Detection using Recombinant Inbred Line microarray expression data.

c.t	<i>Factor 'c' for student's t density</i>
-----	---

Description

Compute the value of factor 'c' for student's t density.

Usage

```
c.t(v)
```

Arguments

v	degree freedom
---	----------------

Value

The value of factor ‘c’.

Author(s)

Na You

References

Cui et al. Single Feature Polymorphisms Detection using Recombinant Inbred Line microarray expression data.

dipSFP

SFP detection procedure based on dip statistic

Description

SFP detection procedure based on dip statistic, where the null distribution of dip statistic is estimated by smooth bootstrap algorithm.

Usage

```
dipSFP(dat, nboot, lambda0 = NULL, outliers.ratio = NULL, m0 = NULL,
pvalue = TRUE, tol = 1e-06)
```

Arguments

<code>dat</code>	a matrix of microarray data to be analyzed, for example, the estimated binding affinity values or SFPdev values, where each row corresponds to one probe and each column corresponds to one chip.
<code>nboot</code>	the number of bootstrap replications.
<code>lambda0, outliers.ratio, m0</code>	tuning parameters, see ‘Details’ in <code>FMbw</code> .
<code>pvalue</code>	logical; will p values be computed?
<code>tol</code>	the desired accuracy (convergence tolerance) in computing the modified critical bandwidth by bisection method.

Details

Dip statistic is computed by `dip` in R package ‘`diptest`’.

Value

A list with four components if `pvalue` is `TRUE`,

<code>dip.statistics</code>	dip statistic values for probes
<code>null.distribution</code>	estimated null distribution of dip statistic
<code>p.values</code>	p values

`adjusted.pvalues`

adjusted p values using the false discovery rate criterion

If `pvalue` is `FALSE`, only `dip.statistics` and `null.distribution` are returned.

Author(s)

Na You

References

Cui et al. Single Feature Polymorphism Detection using Recombinant Inbred Line microarray expression data.

Fisher, N. I. and Marron, J. S. (2001). Mode testing via the excess mass estimate. *Biometrika*, 88(2):499–517.

Hartigan, J. A. and Hartigan, P. M. (1985). The dip test of unimodality. *The Annals of Statistics*, 13(1):70–84.

See Also

[FMbw](#)

Examples

```
n <- 100
m.nonSFP <- 98
m.SFP <- 2

dat1 <- matrix(rgamma(m.nonSFP*n,2,4),m.nonSFP,n)
dat2 <- cbind(matrix(rgamma(m.SFP*n/2,2,4),m.SFP,n/2),
               matrix(rgamma(m.SFP*n/2,20,4),m.SFP,n/2))
dat <- rbind(dat1,dat2)

dipSFP(dat,nboot=20)
dipSFP(dat,nboot=20,outliers.ratio=0.01)
```

FMbw

Modified critical bandwidth

Description

Modified critical bandwidth in Fisher and Marron (2001).

Usage

```
FMbw(vec, k, lambda0 = NULL, outliers.ratio = NULL, m0 = NULL, tol = 1e-06)
```

Arguments

<code>vec</code>	a vector.
<code>k</code>	the number of modes.
<code>lambda0, outliers.ratio, m0</code>	tuning parameters, see ‘Details’ below.
<code>tol</code>	the desired accuracy (convergence tolerance) in computing the modified critical bandwidth by bisection method.

Details

Fisher and Marron (2001) proposed a modified critical bandwidth instead of Silverman critical bandwidth (1981) to estimate the underlying data distribution and generate bootstrap resamples in smooth bootstrap algorithm. Two tuning parameters ‘lambda0’ and ‘m0’ were used to deal with the outliers problems and distinguish minor or major modes. As suggested in Fisher and Marron (2001), ‘lambda0’ can be given directly by user or determined by the ratio of outliers ‘outliers.ratio’. If both of them are given, then the maximum of ‘lambda0’ and ‘outliers.ratio/h/(2*pi)^(1/2)’ is used in computation, where ‘h’ is the bandwidth at which the statistic S_k is calculated. The defaults of ‘lambda0’ and ‘outliers.ratio’ are both zero. The default of ‘m_0’ is 0.003.

Value

The value of modified critical bandwidth.

Author(s)

Na You

References

Fisher, N. I. and Marron, J. S. (2001). Mode testing via the excess mass estimate. *Biometrika*, 88(2):499–517.

Examples

```
x <- rnorm(100, 0, 1)
FMbw(x, 1)
FMbw(x, 1, m0=0)
```

minimax

Minimax function

Description

Determine if each point in a line is a local maximum or minimum.

Usage

```
minimax(func)
```

Arguments

`func` A matrix. Each row indicates the (x,y)-coordinate of one point.

Value

A vector with the same length as the first dimension of matrix `func`, composed of -1, 1, and 0, indicating respectively the corresponding point is a local minimum, maximum or none of them.

Author(s)

Na You

References

Fisher, N. I. and Marron, J. S. (2001). Mode testing via the excess mass estimate. *Biometrika*, 88(2):499–517.

Examples

```
x <- rnorm(10,0,1)
minimax(cbind(seq(x), x))
```

modes	<i>Modes of density function</i>
-------	----------------------------------

Description

Given a vector of data, find the ‘major’ modes of its Gaussian kernel density estimate with specified bandwidth, excluding the modes caused by outliers and ‘minor’ modes with small masses.

Usage

```
modes(vec, bw, lambda0 = NULL, outliers.ratio = NULL, m0 = NULL)
```

Arguments

`vec` a vector.
`bw` bandwidth used in the Gaussian kernel density estimate.
`lambda0, outliers.ratio, m0` tuning parameters. See ‘Details’ in `FMbw`.

Value

A vector, each element of which indicates the location of one mode.

Author(s)

Na You

References

Fisher, N. I. and Marron, J. S. (2001). Mode testing via the excess mass estimate. *Biometrika*, 88(2):499–517.

See Also[FMbw](#)**Examples**

```
x <- rnorm(100,0,1)
h1 <- FMbw(x,k=1)
modes(x,bw=h1)

h2 <- FMbw(x,k=2)
modes(x,bw=h2)
```

sign.plus*Positive truncation function*

Description

Given a vector, its negative elements are replaced by zero.

Usage

```
sign.plus(x)
```

Arguments

x A vector.

Value

A vector with the same length as x.

Author(s)

Na You

Index

`c.gamma`, [1](#)
`c.lognorm`, [2](#)
`c.norm`, [3](#)
`c.t`, [3](#)

`dipSFP`, [4](#)

`FMbw`, [5](#), [5](#), [7](#)

`minimax`, [6](#)
`modes`, [7](#)

`sign.plus`, [8](#)