

Package ‘Rfolding’

October 12, 2022

Type Package

Title The Folding Test of Unimodality

Version 1.0

Description The basic algorithm to perform the folding test of unimodality.

Given a dataset X (d dimensional, n samples), the test checks whether the distribution of the data are rather unimodal or rather multimodal. This package stems from the following research publication:

Siffer Alban, Pierre-Alain Fouque, Alexandre Termier, and Christine Largouët.

“Are your data gathered?” In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining, pp. 2210-2218. ACM, 2018.

[doi:10.1145/3219819.3219994](https://doi.org/10.1145/3219819.3219994).

Encoding UTF-8

License GPL-3

LazyData true

RoxygenNote 6.1.0

Suggests testthat,MASS,knitr,rmarkdown

Imports stats

VignetteBuilder knitr

NeedsCompilation no

Author Alban Siffer [aut, cre],
Amossys [cph, fnd]

Maintainer Alban Siffer <alban.siffer@irisa.fr>

Repository CRAN

Date/Publication 2018-09-30 13:10:12 UTC

R topics documented:

folding.ratio	2
folding.statistics	2
folding.test	3
folding.test.bound	4

folding.test.pvalue	4
pivot.approx	5

Index	6
--------------	----------

folding.ratio	<i>Computes the folding ratio of the input data</i>
---------------	---

Description

Computes the folding ratio of the input data

Usage

```
folding.ratio(X)
```

Arguments

X nxd matrix (n observations, d dimensions)

Value

the folding ratio

Examples

```
X = matrix(runif(n = 1000, min = 0., max = 1.), ncol = 1)
phi = folding.statistics(X)
```

folding.statistics	<i>Computes the folding statistics of the input data</i>
--------------------	--

Description

Computes the folding statistics of the input data

Usage

```
folding.statistics(X)
```

Arguments

X nxd matrix (n observations, d dimensions)

Value

the folding statistics

Examples

```
library(MASS)
mu = c(0,0)
Sigma = matrix(c(1,0.5,1,0.5), ncol = 2)
X = mvrnorm(n = 5000, mu = mu, Sigma = Sigma)
Phi = folding.statistics(X)
```

folding.test	<i>Perform the folding test of unimodality</i>
--------------	--

Description

Perform the folding test of unimodality

Usage

```
folding.test(X)
```

Arguments

X \$nxd\$ matrix (n observations, d dimensions)

Value

1 if unimodal, 0 if multimodal

Examples

```
library(MASS)
n = 10000
d = 3
mu = c(0,0,0)
Sigma = matrix(c(1,0.5,0.5,0.5,1,0.5,0.5,0.5,1), ncol = d)
X = mvrnorm(n = n, mu = mu, Sigma = Sigma)
m = folding.test(X)
```

folding.test.bound *Computes the confidence bound for the significance level p*

Description

Computes the confidence bound for the significance level p

Usage

```
folding.test.bound(n, d, p)
```

Arguments

n	sample size
d	dimension
p	significance level (between 0 and 1, the lower, the more significant)

Value

the confidence bound q (the bounds are 1-q and 1+q)

Examples

```
n = 2000 # number of observations
d = 2    # 2 dimensional data
p = 0.05 # we want the bound at the level 0.05 (classical p-value)
q = folding.test.bound(n,d,p)
```

folding.test.pvalue *Computes the p-value of the folding test*

Description

Computes the p-value of the folding test

Usage

```
folding.test.pvalue(Phi, n, d)
```

Arguments

Phi	the folding statistics
n	sample size
d	dimension

Value

the p-value (the lower, the more significant)

Examples

```
library(MASS)
n = 5000
d = 2
mu = c(0,0)
Sigma = matrix(c(1,0.5,1,0.5), ncol = d)
X = mvrnorm(n = n, mu = mu, Sigma = Sigma)
Phi = folding.statistics(X)
p = folding.test.pvalue(Phi,n,d)
```

<code>pivot.approx</code>	<i>Computes the pivot s_2 (approximate pivot)</i>
---------------------------	--

Description

Computes the pivot s_2 (approximate pivot)

Usage

```
pivot.approx(X)
```

Arguments

X nxd matrix (n observations, d dimensions)

Value

the approximate pivot

Examples

```
library(MASS)
mu = c(0,0)
Sigma = matrix(c(1,0.5,1,0.5), ncol = 2)
X = mvrnorm(n = 5000, mu = mu, Sigma = Sigma)
Phi = pivot.approx(X)
```

Index

fold`ing`.ratio, 2
fold`ing`.statistics, 2
fold`ing`.test, 3
fold`ing`.test.bound, 4
fold`ing`.test.pvalue, 4

pivot.approx, 5