Package 'Ball'

December 14, 2018

Type Package

Title Statistical Inference and Sure Independence Screening via Ball Statistics

Version 1.3.7

Date 2018-12-13

Author Xueqin Wang, Wenliang Pan, Heping Zhang, Hongtu Zhu, Yuan Tian, Weinan Xiao, Chengfeng Liu, Jin Zhu

Maintainer Jin Zhu <zhuj37@mail2.sysu.edu.cn>

Description Hypothesis tests and sure independence screening (SIS) procedure based on ball statistics, including ball divergence <doi:10.1214/17-AOS1579>, ball covariance, and ball correlation <doi:10.1080/01621459.2018.1462709>, are developed to analyze complex data. The ball divergence and ball covariance based distribution-free tests are implemented to detecting distribution difference and association in metric spaces <arXiv:1811.03750>. Furthermore, a generic non-parametric SIS procedure based on ball correlation and all of its variants are implemented to tackle the challenge in the context of ultra high dimensional data.

License GPL-3

RoxygenNote 6.1.1

Depends R (>= 2.10)

Imports utils, gam, survival, mvtnorm

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation yes

Encoding UTF-8

LazyData true

URL https://github.com/Mamba413/Ball

BugReports https://github.com/Mamba413/Ball/issues

Repository CRAN

Date/Publication 2018-12-14 06:00:07 UTC

R topics documented:

`ArcticLake` *Arctic lake sediment samples of different water depth*

Description

Sand, silt and clay compositions of 39 sediment samples of different water depth in an Arctic lake.

Format

`ArcticLake$depth`: water depth (in meters).

`ArcticLake$x`: compositions of three covariates: sand, silt, and clay.

Details

Sand, silt and clay compositions of 39 sediment samples at different water depth (in meters) in an Arctic lake. The additional feature is a concomitant variable or covariate, water depth, which may account for some of the variation in the compositions. In statistical terminology, we have a multivariate regression problem with sediment composition as predictors and water depth as a response. All row percentage sums to 100, except for rounding errors.

Note

Courtesy of J. Aitchison

Source

Aitchison: CODA microcomputer statistical package, 1986, the file name ARCTIC.DAT, here included under the GNU Public Library Licence Version 2 or newer.

References

Aitchison: The Statistical Analysis of Compositional Data, 1986, Data 5, pp5.

bcor *Ball Correlation and Covariance Statistics*

Description

Computes ball covariance and ball correlation statistics, which are multivariate measures of dependence in Banach space.

Usage

```
bcor(x, y, distance = FALSE, weight = FALSE)

bcov(x, y, distance = FALSE, weight = FALSE)
```

Arguments

x	a numeric vector, matirx, data.frame or dist object or list containing numeric vector, matrix, data.frame, or dist object.
y	a numeric vector, matirx, data.frame or dist object.
distance	if distance = TRUE, x and y will be considered as a distance matrix. Default: distance = FALSE
weight	a logical or character value used to choose the form of weight. If weight = FALSE, the ball covariance / correlation with constant weight is used. Alternatively, weight = TRUE and weight = "prob" indicates the probability weight is chosen while setting weight = "chisq" means select the Chi-square weight. Note that this arguments actually only influences the printed result in R console and is only available for the bcov.test function at present. Default: weight = FALSE

Details

bcov and bcor compute ball covariance and ball correlation statistics.

The sample sizes (number of rows or length of the vector) of the two variables must agree, and samples must not contain missing values. If we set distance = TRUE, arguments x, y can be a dist object or a symmetric numeric matrix recording distance between samples; otherwise, these arguments are treated as data.

Ball covariance is a generic non-parametric dependence measure in Banach space, introduced by Pan et al(2017). It is noteworthy that ball covariance enjoys the following properties:

(i) It is nonnegative, and holds the Cauchy-Schwartz type inequality;

(ii) It is nonparametric and makes fewer restrictive data assumptions even without finite moment conditions;

(iii) Its empirical version is feasible and can be used as a test statistic of independence with some desired test properties;

(iv) it is interesting that the HHG dependence measure is a special case of ball covariance.

Ball correlation, based on the normalized ball covariance, generalizes the idea of Pearson correlation in two fundamental ways:

(i) Ball correlation, $\mathbf{BCor}_{\omega}^{2}(X,Y)$, is defined for X and Y in arbitrary dimension in Banach space.

(ii) Ball correlation satisfies $0 \leq \mathbf{BCor}_{\omega}^{2}(X,Y) \leq 1$, and $\mathbf{BCor}_{\omega}^{2}(X,Y) = 0$ only if X and Y are independent.

The definitions of the sample version ball covariance and ball correlation are as follows. Suppose, we are given pairs of independent observations $\{(x_1,y_1),...,(x_n,y_n)\}$, where x_i and y_i can be of any dimension and the dimensionality of x_i and y_i need not be the same. Then, we define sample version ball covariance as:

$$\mathbf{BCor}_{\omega,n}^{2}(X,Y) = \frac{1}{n^2}\sum_{i,j=1}^{n}(\Delta_{ij,n}^{X,Y} - \Delta_{ij,n}^{X}\Delta_{ij,n}^{Y})^2$$

where:

$$\Delta_{ij,n}^{X,Y} = \frac{1}{n}\sum_{k=1}^{n}\delta_{ij,k}^{X}\delta_{ij,k}^{Y}, \Delta_{ij,n}^{X} = \frac{1}{n}\sum_{k=1}^{n}\delta_{ij,k}^{X}, \Delta_{ij,n}^{Y} = \frac{1}{n}\sum_{k=1}^{n}\delta_{ij,k}^{Y}$$

$$\delta_{ij,k}^{X} = I(x_k \in \bar{B}(x_i,\rho(x_i,x_j))), \delta_{ij,k}^{Y} = I(y_k \in \bar{B}(y_i,\rho(y_i,y_j)))$$

Among them, $\bar{B}(x_i,\rho(x_i,x_j))$ is a closed ball with center x_i and radius $\rho(x_i,x_j)$. Similarly, we can give the notations $\mathbf{BCov}_{\omega,n}^{2}(\mathbf{X},\mathbf{X})$ and $\mathbf{BCov}_{\omega,n}^{2}(\mathbf{Y},\mathbf{Y})$, which are the sample version of $\mathbf{BCov}_{\omega}^{2}(\mathbf{X},\mathbf{X})$ and $\mathbf{BCov}_{\omega}^{2}(\mathbf{Y},\mathbf{Y})$. We thus define the sample version ball correlation as follows.

$$\mathbf{BCor}_{\omega,n}^{2}(\mathbf{X},\mathbf{Y}) = \mathbf{BCov}_{\omega,n}^{2}(\mathbf{X},\mathbf{Y})/\sqrt{\mathbf{BCov}_{\omega,n}^{2}(\mathbf{X},\mathbf{X})\mathbf{BCov}_{\omega,n}^{2}(\mathbf{Y},\mathbf{Y})}$$

Moreover, it is natural to extend $\mathbf{BCov}_{\omega,n}$ to measure the mutual independence between K random variables:

$$\frac{1}{n^2}\sum_{i,j=1}^{n}\left[(\Delta_{ij,n}^{R_1,...,R_K} - \prod_{k=1}^{K}\Delta_{ij,n}^{R_k})^2\prod_{k=1}^{K}\hat{\omega}_k(R_{ki},R_{kj})\right]$$

where $R_k, k = 1,...K$ indicate random variables and $R_{ki}, i = 1,...,n$ denote i th random samples of R_k.

See `bcov.test` for a test of multivariate independence based on the ball covariance and ball correlation statistic.

Value

`bcor`	sample version of ball correlation.
`bcov`	sample version of ball covariance.

See Also

`bcov.test`, `bcorsis`

Examples

```
############# Ball Correlation #############
num <- 50
x <- 1:num
y <- 1:num
bcor(x, y)
bcor(x, y, weight = TRUE)
bcor(x, y, weight = "prob")
bcor(x, y, weight = "chisq")
############# Ball Covariance #############
n <- 50
x <- rnorm(n)
y <- rnorm(n)
bcov(x, y)
bcov(x, y, weight = TRUE)
bcov(x, y, weight = "prob")
bcov(x, y, weight = "chisq")
```

`bcorsis` *Ball Correlation Sure Independence Screening*

Description

Generic non-parametric sure independence screening procedure based on ball correlation. Ball correlation is a generic multivariate measure of dependence in Banach space.

Usage

```
bcorsis(x, y, d = "small", weight = FALSE, method = "standard",
  distance = FALSE, parms = list(d1 = 5, d2 = 5, df = 3),
  num.threads = 2)
```

Arguments

`x`	a numeric matirx or data.frame included n rows and p columns. Each row is an observation vector and each column corresponding to a explanatory variable, generally $p >> n$.
`y`	a numeric vector, matirx, data.frame or `dist` object.
`d`	the hard cutoff rule suggests selecting d variables. Setting `d = "large"` or `d = "small"` means `n-1` or `floor(n/log(n))` variables are selected. If `d` is a integer, `d` variables are selected. Default: `d = "small"`
`weight`	when `weight = TRUE`, weighted ball correlation is used instead of ball correlation. Default: `weight = FALSE`
`method`	method for sure independence screening procedure, include: `"standard"`, `"lm"`, `"gam"`, `"interaction"` and `"survival"`. Setting `method = "standard"` means standard sure independence screening procedure based on ball correlation while options `"lm"` and `"gam"` carry out iterative BCor-SIS procedure with ordinary

	linear regression and generalized additive models, respectively. Options "interaction" and "survival" are designed for detecting variables with potential linear interaction or associated with censored responses. Default: method = "standard"
distance	if distance = TRUE, y will be considered as a distance matrix. Arguments only available when method = "standard" and method = "interaction". Default: distance = FALSE
parms	parameters list only available when method = "lm" or "gam". It contains three parameters: d1, d2, and df. d1 is the number of initially selected variables, d2 is the number of variables collection size added in each iteration. df is degree freedom of basis in generalized additive models playing a role only when method = "gam". Default: parms = list(d1 = 5, d2 = 5, df = 3)
num.threads	Number of threads. Default num.threads = 1.

Details

bcorsis implements a model-free generic screening procedure, BCor-SIS, with fewer and less restrictive assumptions. The sample sizes (number of rows or length of the vector) of the two variables x and y must agree, and samples must not contain missing values.

BCor-SIS procedure for censored response is carried out when method = "survival". At that time, the matrix or data.frame pass to argument y must have exactly two columns and the first column is event (failure) time while the second column is censored status, a dichotomous variable.

If we set distance = TRUE, arguments y is considered as distance matrix, otherwise y is treated as data.

BCor-SIS is based on a recently developed universal dependence measure: Ball correlation (BCor). BCor efficiently measures the dependence between two random vectors, which is between 0 and 1, and 0 if and only if these two random vectors are independent under some mild conditions. (See the manual page for bcor.)

Theory and numerical result indicate that BCor-SIS has following advantages:

(i) It has a strong screening consistency property without finite sub-exponential moments of the data. Consequently, even when the dimensionality is an exponential order of the sample size, BCor-SIS still almost surely able to retain the efficient variables.

(ii) It is nonparametric and has the property of robustness.

(iii) It works well for complex responses and/or predictors, such as shape or survival data

(iv) It can extract important features even when the underlying model is complicated.

Value

ix	the vector of indices selected by ball correlation sure independence screening procedure.
method	the method used.
weight	the weight used.
complete.info	a list containing at least one $px3$ matrix, where each row is corresponding to variable and each column is corresponding to differe ball correlation weight. If method = "gam" or method = "lm", complete.info is empty list.

Author(s)

Wenliang Pan, Weinan Xiao, Xueqin Wang, Hongtu Zhu

References

Wenliang Pan, Xueqin Wang, Weinan Xiao & Hongtu Zhu (2018) A Generic Sure Independence Screening Procedure, Journal of the American Statistical Association, DOI: 10.1080/01621459.2018.1462709

Jin, Zhu, Wenliang Pan, Wei Zheng, and Xueqin Wang (2018). Ball: An R package for detecting distribution difference and association in metric spaces. arXiv preprint arXiv:1811.03750. URL http://arxiv.org/abs/1811.03750.

See Also

bcor

Examples

```
## Not run:

############### Quick Start for bcorsis function ###############
set.seed(1)
n <- 150
p <- 3000
x <- matrix(rnorm(n * p), nrow = n)
error <- rnorm(n)
y <- 3*x[, 1] + 5*(x[, 3])^2 + error
res <- bcorsis(y = y, x = x)
head(res[["ix"]])

############### BCor-SIS: Censored Data Example ###############
data("genlung")
result <- bcorsis(x = genlung[["covariate"]], y = genlung[["survival"]],
                  method = "survival")
index <- result[["ix"]]
top_gene <- colnames(genlung[["covariate"]])[index]
head(top_gene, n = 1)


############### BCor-SIS: Interaction Pursuing ###############
set.seed(1)
n <- 150
p <- 3000
x <- matrix(rnorm(n * p), nrow = n)
error <- rnorm(n)
y <- 3*x[, 1]*x[, 5]*x[, 10] + error
res <- bcorsis(y = y, x = x, method = "interaction")
head(res[["ix"]])

############### BCor-SIS: Iterative Method ###############
library(mvtnorm)
set.seed(1)
```

```
n <- 150
p <- 3000
sigma_mat <- matrix(0.5, nrow = p, ncol = p)
diag(sigma_mat) <- 1
x <- rmvnorm(n = n, sigma = sigma_mat)
error <- rnorm(n)
rm(sigma_mat); gc(reset = TRUE)
y <- 3*(x[, 1])^2 + 5*(x[, 2])^2 + 5*x[, 8] - 8*x[, 16] + error
res <- bcorsis(y = y, x = x, method = "lm", d = 15)
res <- bcorsis(y = y, x = x, method = "gam", d = 15)
res[["ix"]]

############### Weighted BCor-SIS: Probability weight ###############
set.seed(1)
n <- 150
p <- 3000
x <- matrix(rnorm(n * p), nrow = n)
error <- rnorm(n)
y <- 3*x[, 1] + 5*(x[, 3])^2 + error
res <- bcorsis(y = y, x = x, weight = "prob")
head(res[["ix"]])
# Alternative, chisq weight:
res <- bcorsis(y = y, x = x, weight = "chisq")
head(res[["ix"]])

## End(Not run)
```

`bcov.test` *Ball Covariance Test*

Description

Ball covariance test of multivariate independence. Ball covariance are generic multivariate measures of dependence in Banach space.

Usage

```
bcov.test(x, ...)

## Default S3 method:
bcov.test(x, y = NULL, num.permutations = 99,
  distance = FALSE, weight = FALSE, seed = 4, num.threads = 1, ...)

## S3 method for class 'formula'
bcov.test(formula, data, subset, na.action, ...)
```

Arguments

`x`	a numeric vector, matirx, data.frame or `dist` object or list containing numeric vector, matrix, data.frame, or `dist` object.

...	further arguments to be passed to or from methods.
y	a numeric vector, matirx, data.frame or dist object.
num.permutations	the number of permutation replications, when num.permutations equals to 0, the function returns the sample version of ball divergence. Default: num.permutations = 99
distance	if distance = TRUE, x and y will be considered as a distance matrix. Default: distance = FALSE
weight	a logical or character value used to choose the form of weight. If weight = FALSE, the ball covariance / correlation with constant weight is used. Alternatively, weight = TRUE and weight = "prob" indicates the probability weight is chosen while setting weight = "chisq" means select the Chi-square weight. Note that this arguments actually only influences the printed result in R console and is only available for the bcov.test function at present. Default: weight = FALSE
seed	the random seed.
num.threads	Number of threads. Default num.threads = 1.
formula	a formula of the form ~ u + v, where each of u and v are numeric variables giving the data values for one sample. The samples must be of the same length.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").

Details

bcov.test are non-parametric tests of multivariate independence in Banach space. The test decision is obtained via permutation, with num.permutations replicates.

If two samples are pass to arguments x and y, the sample sizes (i.e. number of rows or length of the vector) of the two variables must agree. If a list object is passed to x, each element must with same sample sizes. Moreover, data pass to x or y must not contain missing or infinite values. If we set distance = TRUE, arguments x, y can be a dist object or a symmetric numeric matrix recording distance between samples; otherwise, these arguments are treated as data.

The bcov.test statistic is bcov or bcor which are dependence measure in Banach space. The bcor test statistic is based on the normalized coefficient of ball covariance. (See the manual page for bcov or bcor.)

For the general problem of testing independence when the distributions of X and Y are unknown, the test based on bcov can be implemented as a permutation test. See (Jin et al 2018) for theoretical properties of the test, including statistical consistency.

Value

bcov.test returns a list with class "htest" containing the following components:

statistic	ball covariance or ball correlation statistic.

p.value	the p-value for the test.
replicates	permutation replications of the test statistic.
size	sample size.
complete.info	a list containing multiple statistics value and their corresponding p value.
alternative	a character string describing the alternative hypothesis.
method	a character string indicating what type of test was performed.
data.name	description of data.

Author(s)

Wenliang Pan, Xueqin Wang, Heping Zhang, Hongtu Zhu, Jin Zhu

References

Jin, Zhu, Wenliang Pan, Wei Zheng, and Xueqin Wang (2018). Ball: An R package for detecting distribution difference and association in metric spaces. arXiv preprint arXiv:1811.03750. URL http://arxiv.org/abs/1811.03750.

See Also

bcov, bcor

Examples

```
set.seed(1)

################## Quick Start #################
error <- runif(50, min = -0.3, max = 0.3)
x <- runif(50, 0, 4*pi)
y <- cos(x) + error
# plot(x, y)
bcov.test(x, y)

################## Quick Start #################
x <- matrix(runif(50 * 2, -pi, pi), nrow = 50, ncol = 2)
error <- runif(50, min = -0.3, max = 0.3)
y <- (sin((x[,1])^2 + x[,2])) + error
bcov.test(x = x, y = y)

################## Ball Covariance Test for Non-Hilbert Data #################
# load data:
data("ArcticLake")
# Distance matrix between y:
Dy <- nhdist(ArcticLake[["x"]], method = "compositional")
# Distance matrix between x:
Dx <- dist(ArcticLake[["depth"]])
# hypothesis test with BCov:
bcov.test(x = Dx, y = Dy, distance = TRUE)

################  Weighted Ball Covariance Test  #################
```

```
data("ArcticLake")
Dy <- nhdist(ArcticLake[["x"]], method = "compositional")
Dx <- dist(ArcticLake[["depth"]])
# hypothesis test with weighted BCov:
bcov.test(x = Dx, y = Dy, distance = TRUE, weight = TRUE)

################# Mutual Independence Test #################
x <- rnorm(30)
y <- (x > 0) * x + rnorm(30)
z <- (x <= 0) * x + rnorm(30)
data_list <- list(x, y, z)
bcov.test(data_list)

################# Mutual Independence Test for Meteorology data #################
data("meteorology")
bcov.test(meteorology)

################  Formula interface  ################
## independence test:
bcov.test(~ CONT + INTG, data = USJudgeRatings)
## mutual independence test:
bcov.test(~ CONT + INTG + DMNR, data = USJudgeRatings)
```

---

bd *Ball Divergence*

Description

Compute ball divergence statistic between two-sample or K-sample.

Usage

```
bd(x, y = NULL, distance = FALSE, size = NULL, num.threads = 1,
  kbd.type = "sum")
```

Arguments

x	a numeric vector, matrix, data.frame, `dist` object or list containing vector, matrix, or data.frame.
y	a numeric vector, matrix or data.frame.
distance	if `distance = TRUE`, `x` will be considered as a distance matrix. Default: `distance = FALSE`
size	a vector record sample size of each group.
num.threads	Number of threads. Default `num.threads = 1`.
kbd.type	a character value controlling the output information. Setting `kdb.type = "sum"`, `kdb.type = "summax"`, or `kdb.type = "max"`, the corresponding statistics value and p-value of K-sample test procedure are demonstrated. Note that this arguments actually only influences the printed result in R console. Default: `kdb.type = "sum"`

Details

Given the samples not containing missing values, bd returns sample version of ball divergence. If we set `distance = TRUE`, arguments `x`, `y` can be a `dist` object or a symmetric numeric matrix recording distance between samples; otherwise, these arguments are treated as data.

Ball divergence, introduced by Pan et al(2017), is a new concept to measure the difference between two probability distributions in separable Banach space. Ball divergence of two probability measures is proven to be zero if and only if they are identical.

The definitions of the sample version ball divergence are as follows. Given two independent samples $\{x_1, ..., x_n\}$ with the associated probability measure μ and $\{y_1, ..., y_m\}$ with ν, where the observations in each sample are *i.i.d.*

Also, let $\delta(x, y, z) = I(z \in \bar{B}(x, \rho(x, y)))$, where $\delta(x, y, z)$ indicates whether z is located in the closed ball $\bar{B}(x, \rho(x, y))$ with center x and radius $\rho(x, y)$. We denote:

$$A_{ij}^X = \frac{1}{n}\sum_{u=1}^{n}\delta(X_i, X_j, X_u), \quad A_{ij}^Y = \frac{1}{m}\sum_{v=1}^{m}\delta(X_i, X_j, Y_v)$$

$$C_{kl}^X = \frac{1}{n}\sum_{u=1}^{n}\delta(Y_k, Y_l, X_u), \quad C_{kl}^Y = \frac{1}{m}\sum_{v=1}^{m}\delta(Y_k, Y_l, Y_v)$$

A_{ij}^X represents the proportion of samples $\{x_1, ..., x_n\}$ located in the ball $\bar{B}(X_i, \rho(X_i, X_j))$ and A_{ij}^Y represents the proportion of samples $\{y_1, ..., y_m\}$ located in the ball $\bar{B}(X_i, \rho(X_i, X_j))$. Meanwhile, C_{kl}^X and C_{kl}^Y represent the corresponding proportions located in the ball $\bar{B}(Y_k, \rho(Y_k, Y_l))$.

we can define sample version ball divergence as:

$$D_{n,m} = A_{n,m} + C_{n,m}$$

BD can be generalized to the *K*-sample problem, i.e. if we have K group samples, each group include $n^{(k)}, k = 1, ..., K$ samples, then we can define sample version of generalized ball divergence for *K*-sample problem:

$$\sum_{1 \le k < l \le K} D_{n^{(k)}, n^{(l)}}$$

See `bd.test` for a test of multivariate independence based on the ball divergence.

Value

bd	sample version of ball divergence

Author(s)

Wenliang Pan, Yuan Tian, Xueqin Wang, Heping Zhang

References

Wenliang Pan, Yuan Tian, Xueqin Wang, Heping Zhang. (2017) Ball divergence: nonparametric two sample test, *The Annals of Statistics*, to appear

See Also

bd.test

Examples

```
############# Ball Divergence #############
x <- rnorm(50)
y <- rnorm(50)
bd(x, y)
```

---

bd.test *Ball Divergence based Equality of Distributions Test*

Description

Performs the nonparametric two-sample or K-sample ball divergence test for equality of multivariate distributions

Usage

```
bd.test(x, ...)

## Default S3 method:
bd.test(x, y = NULL, num.permutations = 99,
  distance = FALSE, size = NULL, seed = 4, num.threads = 1,
  kbd.type = "sum", ...)

## S3 method for class 'formula'
bd.test(formula, data, subset, na.action, ...)
```

Arguments

x	a numeric vector, matrix, data.frame, dist object or list containing vector, matrix, or data.frame.
...	further arguments to be passed to or from methods.
y	a numeric vector, matrix or data.frame.
num.permutations	the number of permutation replications, when num.permutations equals to 0, the function returns the sample version of ball divergence. Default: num.permutations = 99
distance	if distance = TRUE, x will be considered as a distance matrix. Default: distance = FALSE
size	a vector record sample size of each group.
seed	the random seed.
num.threads	Number of threads. Default num.threads = 1.

kbd.type	a character value controlling the output information. Setting kdb.type = "sum", kdb.type = "summax", or kdb.type = "max", the corresponding statistics value and p-value of K-sample test procedure are demonstrated. Note that this arguments actually only influences the printed result in R console. Default: kdb.type = "sum"
formula	a formula of the form response ~ group where response gives the data values and group a vector or factor of the corresponding groups.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").

Details

bd.test are ball divergence based multivariate nonparametric tests of two-sample or K-sample problem. If only x is given, the statistic is computed from the original pooled samples, stacked in matrix where each row is a multivariate observation, or from the distance matrix when distance = TRUE. The first sizes[1] rows of x are the first sample, the next sizes[2] rows of x are the second sample, etc. If x is a list, its elements are taken as the samples to be compared, and hence have to be numeric data vectors, matrix or data.frame.

Based on sample version ball divergence (see bd), the test is implemented by permutation with num.permutations times. The function simply returns the test statistic when num.permutations = 0.

Value

bd.test returns a list with class "htest" containing the following components:

statistic	ball divergence statistic.
p.value	the p-value for the test.
replicates	permutation replications of the test statistic.
size	sample sizes.
complete.info	a list containing multiple statistics value and their corresponding p value.
alternative	a character string describing the alternative hypothesis.
method	a character string indicating what type of test was performed.
data.name	description of data.

Author(s)

Wenliang Pan, Yuan Tian, Xueqin Wang, Heping Zhang

References

Pan, Wenliang; Tian, Yuan; Wang, Xueqin; Zhang, Heping. Ball Divergence: Nonparametric two sample test. Ann. Statist. 46 (2018), no. 3, 1109–1137. doi:10.1214/17-AOS1579. https://projecteuclid.org/euclid.aos/15253

Jin, Zhu, Wenliang Pan, Wei Zheng, and Xueqin Wang (2018). Ball: An R package for detecting distribution difference and association in metric spaces. arXiv preprint arXiv:1811.03750. URL http://arxiv.org/abs/1811.03750.

See Also

bd

Examples

```
################## Quick Start #################
x <- rnorm(50)
y <- rnorm(50, mean = 1)
# plot(density(x))
# lines(density(y), col = "red")
# ball divergence:
bd.test(x = x, y = y)

################## Quick Start #################
x <- matrix(rnorm(100), nrow = 50, ncol = 2)
y <- matrix(rnorm(100, mean = 3), nrow = 50, ncol = 2)
# Hypothesis test with Standard Ball Divergence:
bd.test(x = x, y = y)

################## Simlated Non-Hilbert data #################
data("bdvmf")
## Not run:
library(scatterplot3d)
scatterplot3d(bdvmf[["x"]], color = bdvmf[["group"]],
              xlab = "X1", ylab = "X2", zlab = "X3")

## End(Not run)
# calculate geodesic distance between sample:
Dmat <- nhdist(bdvmf[["x"]], method = "geodesic")
# hypothesis test with BD :
bd.test(x = Dmat, size = c(150, 150), num.permutations = 99, distance = TRUE)

################## Non-Hilbert Real Data #################
# load data:
data("macaques")
# number of femala and male Macaca fascicularis:
table(macaques[["group"]])
# calculate Riemannian shape distance matrix:
Dmat <- nhdist(macaques[["x"]], method = "riemann")
# hypothesis test with BD:
bd.test(x = Dmat, num.permutations = 99, size = c(9, 9), distance = TRUE)

################  K-sample Test  #################
```

```
n <- 150
bd.test(rnorm(n), size = c(40, 50, 60))
# alternative input method:
x <- lapply(c(40, 50, 60), rnorm)
bd.test(x)

################  Formula interface  ################
## Two-sample test
bd.test(extra ~ group, data = sleep)
## K-sample test
bd.test(Sepal.Width ~ Species, data = iris)
```

---

bdvmf *Simulated von Mises-Fisher Data*

Description

Simulated random vectors following the von Mises-Fisher distribution with mean direction $\mu_x = (1, 0, 0)$ and $\mu_y = (1, 1, 1)$, and concentration parameter is $\kappa = 3$.

Format

bdvmf$x: A 300×3 numeric matrix containing simulated von Mises-Fisher data.

bdvmf$group: A group index vector.

Details

In directional statistics, the von Mises–Fisher distribution (named after Ronald Fisher and Richard von Mises), is a probability distribution on the $(p-1)$-dimensional sphere in R^p

The parameters μ, and κ, are called the mean direction and concentration parameter, respectively. The greater the value of κ, the higher the concentration of the distribution around the mean direction μ,. The distribution is unimodal for κ, and is uniform on the sphere for $\kappa = 0$.

References

Embleton, N. I. Fisher, T. Lewis, B. J. J. (1993). Statistical analysis of spherical data (1st pbk. ed.). Cambridge: Cambridge University Press. pp. 115–116. ISBN 0-521-45699-1.

genlung *Lung cancer genomic data*

Description

Publicly available lung cancer genomic data from the Chemores Cohort Study, containing the expression levels of mRNA, miRNA, artificial noise variables as well as clinical variables and response.

Format

`genlung$survival`: A data.frame containing $n = 123$ complete observations. The first column is disease-free survival time and the second column is censoring status.

`genlung$covariate`: A data.frame containing $p = 2000$ covariates.

Details

Tissue samples were analysed from a cohort of 123 patients, who underwent complete surgical resection at the Institut Mutualiste Montsouris (Paris, France) between 30 January 2002 and 26 June 2006. The studied outcome was the "Disease-Free Survival Time". Patients were followed until the first relapse occurred or administrative censoring. In this genomic dataset, the expression levels of Agilent miRNA probes ($p = 939$) were included from the $n = 123$ cohort samples. The miRNA data contains normalized expression levels. See below the paper by Lazar et al. (2013) and Array Express data repository for the complete description of the samples, tissue preparation, Agilent array technology, and data normalization. In addition to the genomic data, five clinical variables, also evaluated on the cohort samples, are included as continuous variable ('Age') and nominal variables ('Type','KRAS.status','EGFR.status','P53.status'). See Lazar et al. (2013) for more details. Moreover, we add 1056 standard gaussian variables which are independent with the censored response as noise covariates. This dataset represents a situation where the number of covariates dominates the number of complete observations or $p >> n$ case.

References

Lazar V. et al. (2013). Integrated molecular portrait of non-small cell lung cancers. BMC Medical Genomics 6:53-65.

macaques *Male and Female macaque data*

Description

Male and female macaque skull data. 7 landmarks in 3 dimensions, 18 individuals (9 males, 9 females)

Format

macaques$x: An array of dimension $7 \times 3 \times 18$

macaques$group: A factor indicating the sex ('m' for male and 'f' for female)

Details

In an investigation into sex differences in the crania of a species of Macaca fascicularis (a type of monkey), random samples of 9 male and 9 female skulls were obtained by Paul O'Higgins (Hull-York Medical School) (Dryden and Mardia 1993). A subset of seven anatomical landmarks was located on each cranium and the three-dimensional (3D) coordinates of each point were recorded.

Note

Dryden, I.L. and Mardia, K.V. (1998). Statistical Shape Analysis, Wiley, Chichester.

References

Dryden, I. L. and Mardia, K. V. (1993). Multivariate shape analysis. Sankhya Series A, 55, 460-480.

---

meteorology *meteorological data*

Description

A meteorological data include 46 records about air, soil, humidity, wind and evaporation.

Format

meteorology$air: A data.frame containing 3 variables: maximum, minimum and average daily air temperature

meteorology$soil: A data.frame containing 3 covariates: maximum, minimum and average daily soil temperature

meteorology$humidity: A data.frame containing 3 covariates: maximum, minimum and average daily humidity temperature,

meteorology$wind: a vector object record total wind, measured in miles per day

meteorology$evaporation: a vector object record evaporation

Details

This meteorological data containing 46 observations on five groups of variables: air temperature, soil temperature, relative humidity, wind speed as well as evaporation. Among them, maximum, minimum and average value for air temperature, soil temperature, and relative humidity are recorded. As regards to wind speed and evaporation, there are univariate numerical variables. We desire to test the independence of these five groups of variables.

nhdist *Distance Matrix Computation for Non-Hilbert Data*

Description

This function computes and returns the numeric distance matrix computed by using the specified distance measure to compute the distances between the rows of a data matrix.

Usage

```
nhdist(x, method = "geodesic")
```

Arguments

x	a numeric matrix, data frame or numeric array of dimension $k \times m \times n$ containing n samples in $k \times m$ dimension.
method	the distance measure to be used. This must be one of "geodesic", "compositional", "riemann". Any unambiguous substring can be given.

Details

Available distance measures are geodesic, compositional and riemann. Denoting any two sample in the dataset as x and y, we give the definition of distance measures as follows.

geodesic:

The shortest route between two points on the Earth's surface, namely, a segment of a great circle.

$$acos(x^T y), \|x\|_2 = \|y\|_2 = 1$$

compositional:

First, we apply scale transformation to it, i.e., $(x_{i1}/t, ..., x_{ip}/t_i), t_i = \sum_{d=1}^{p} x_d$. Then, apply the square root transformation to data and calculate the geodesic distance between samples.

riemann:

$k \times m \times n$ array where k = number of landmarks, m = number of dimensions and n = sample size. Detail about riemannian shape distance was given in Kendall, D. G. (1984).

Value

$n \times n$ numeric distance matrix

References

Kendall, D. G. (1984). Shape manifolds, Procrustean metrics and complex projective spaces, Bulletin of the London Mathematical Society, 16, 81-121.

Examples

```
data('bdvmf')
Dmat <- nhdist(bdvmf[['x']], method = "geodesic")

data("ArcticLake")
Dmat <- nhdist(ArcticLake[['x']], method = "compositional")

data("macaques")
Dmat <- nhdist(macaques[["x"]], method = "riemann")

# unambiguous substring also available:
Dmat <- nhdist(macaques[["x"]], method = "rie")
```

Index