

Package ‘QuadratiK’

February 23, 2024

Type Package

Title Collection of Methods Constructed using Kernel-Based Quadratic Distances

Version 1.0.0

Date 2024-01-31

Maintainer Giovanni Saraceno <gsaracen@buffalo.edu>

Description It includes test for multivariate normality, test for uniformity on the Sphere, non-parametric two- and k-sample tests, random generation of points from the Poisson kernel-based density and clustering algorithm for spherical data. For more information see Saraceno, G., Markatou, M., Mukhopadhyay, R., Golzy, M. (2024) <[arxiv:2402.02290](#)>, Ding, Y., Markatou, M., Saraceno, G. (2023) <[doi:10.5705/ss.202022.0347](#)>, and Golzy, M., Markatou, M. (2020) <[doi:10.1080/10618600.2020.1740713](#)>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

LinkingTo Rcpp, RcppEigen

Depends R (>= 3.5.0)

Imports Rcpp, RcppEigen, sn, methods, stats, MASS, moments, mvtnorm, movMF, rlecuyer, foreach, doParallel, Tinflex, ggpp, ggplot2, rgl, clusterRepro, ggpubr, mclust, rrcov, cluster

RoxygenNote 7.2.3

VignetteBuilder knitr

Suggests rmarkdown, testthat (>= 3.0.0), roxygen2, knitr

Config/testthat/edition 3

NeedsCompilation yes

Author Giovanni Saraceno [aut, cre] (ORCID 000-0002-1753-2367),
Marianthi Markatou [aut],
Raktim Mukhopadhyay [aut],
Mojgan Golzy [ctb]

Repository CRAN

Date/Publication 2024-02-23 18:30:05 UTC

R topics documented:

QuadratiK-package	2
dpkb	3
kb.test	4
kb.test-class	7
pk.test	8
pk.test-class	9
pkbc	10
pkbc-class	12
sample_hypersphere	12
select_h	13
summary,kb.test-method	14
summary,pk.test-method	15
summary_stat	16
validation	17
wireless	18

Index	20
--------------	-----------

QuadratiK-package	<i>Collection of methods constructed using kernel-based quadratic distances</i>
-------------------	---

Description

The QuadratiK package is implemented in R and Python, providing a comprehensive set of goodness-of-fit tests and a clustering technique using kernel-based quadratic distances. This framework aims to bridge the gap between the statistical and machine learning literature. It includes:

- **Goodness-of-Fit Tests:** The software implements one, two, and k-sample tests for goodness of fit, offering an efficient and mathematically sound way to assess the fit of probability distributions. Expanded capabilities include supporting tests for uniformity on the d -dimensional sphere based on Poisson kernel densities.
- **Clustering Algorithm for Spherical Data:** the package incorporates a unique clustering algorithm specifically tailored for spherical data. This algorithm leverages a mixture of Poisson kernel-based densities on the sphere, enabling effective clustering of spherical data or data that has been spherically transformed. The package also provides the functions for density evaluation and random sampling from the Poisson kernel-based distribution.
- **Additional Features:** Alongside these functionalities, the software includes additional graphical functions, aiding users in validating and representing the cluster results as well as enhancing the interpretability and usability of the analysis.

Details

The work has been supported by Kaleida Health Foundation, National Science Foundation and Department of Biostatistics, Univerisity at Buffalo.

Author(s)

Giovanni Saraceno, Marianthi Markatou, Raktim Mukhopadhyay, Mojgan Golzy
 Maintainer: Giovanni Saraceno <gsaracen@buffalo.edu>

References

- Saraceno, G., Markatou, M., Mukhopadhyay, R., Golzy, M. (2023). Goodness-of-fit and clustering of spherical data: The QuadratiK package in R and Python. Technical Report, Department of Biostatistics, University at Buffalo.
- Ding, Y., Markatou, M., Saraceno, G. (2023). "Poisson Kernel-Based Tests for Uniformity on the d-Dimensional Sphere." *Statistica Sinica*. doi: doi:10.5705/ss.202022.0347.
- Golzy, M., Markatou, M. (2020) Poisson Kernel-Based Clustering on the Sphere: Convergence Properties, Identifiability, and a Method of Sampling, *Journal of Computational and Graphical Statistics*, 29:4, 758-770, DOI: 10.1080/10618600.2020.1740713.
- Markatou, M., Saraceno, G., Chen, Y. (2023). "A unified framework for multivariate two-sample and k-sample kernel-based quadratic distance goodness-of-fit tests" Manuscript, (Department of Biostatistics, University at Buffalo).

dpcb

*The Poisson kernel-based Distribution (PKBD)***Description**

Density function and random number generation from the Poisson kernel-based Distribution with mean direction vector μ and concentration parameter ρ .

Usage

```
dpcb(x, mu, rho, logdens = FALSE)

rpcb(n, mu, rho, method = "rejvmf")
```

Arguments

x	Matrix (or data.frame) with number of columns ≥ 2 .
mu	location vector parameter with length indicating the dimension of generated points.
rho	is the concentration parameter, with $0 \leq \rho < 1$.
logdens	Logical; if 'TRUE', densities d are given as $\log(d)$.
n	number of observations.
method	string that indicates the method used for sampling observations. The available methods are <ul style="list-style-type: none"> 'rejvmf' acceptance-rejection algorithm using von Mises-Fisher envelopes (Algorithm in Table 2 of Golzy and Markatou 2020);

- 'rejacg' using angular central Gaussian envelopes (Algorithm in Table 1 of Sablica et al. 2023);
- 'rejsaw' using projected Saw distributions (Algorithm in Table 2 of Sablica et al. 2023).

Value

dpkb gives the density value. rpkb generates random observations from the PKBD.

The number of observations generated is determined by n for rpkb. This function returns a list with the matrix of generated observations x, the number of tries numTries and the number of acceptances numAccepted.

References

Golzy, M., Markatou, M. (2020) Poisson Kernel-Based Clustering on the Sphere: Convergence Properties, Identifiability, and a Method of Sampling, *Journal of Computational and Graphical Statistics*, 29:4, 758-770, DOI: 10.1080/10618600.2020.1740713.

Sablica L., Hornik K., Leydold J. (2023) "Efficient sampling from the PKBD distribution", *Electronic Journal of Statistics*, 17(2), 2180-2209.

Examples

```
# Generate some data from pkbd density
pkbd_dat <- rpkb(10, c(0.5,0), 0.5)

# Calculate the PKBD density values
dens_val <- dpkb(pkbd_dat$x, c(0.5,0.5),0.5)
```

kb.test

Kernel-based quadratic distance Goodness-of-Fit tests

Description

This function performs the kernel-based quadratic distance goodness-of-fit tests using the Gaussian kernel with tuning parameter h.

Usage

```
kb.test(
  x,
  y = NULL,
  h = NULL,
  method = "subsampling",
  B = 150,
  b = NULL,
```

```

    Quantile = 0.95,
    mu_hat = NULL,
    Sigma_hat = NULL,
    centeringType = "Nonparam",
    K_threshold = 10,
    alternative = "skewness"
)

## S4 method for signature 'ANY'
kb.test(
  x,
  y = NULL,
  h = NULL,
  method = "subsampling",
  B = 150,
  b = 0.9,
  Quantile = 0.95,
  mu_hat = NULL,
  Sigma_hat = NULL,
  centeringType = "Nonparam",
  K_threshold = 10,
  alternative = "skewness"
)

## S4 method for signature 'kb.test'
show(object)

```

Arguments

x	Numeric matrix or vector of data values.
y	Numeric matrix or vector of data values. Depending on the input y, the corresponding test is performed. <ul style="list-style-type: none"> • if y = NULL, the function performs the tests for normality on x • if y is a data matrix, with same dimensions of x, the function performs the two-sample test between x and y. • if y is a numeric or factor vector, indicating the group memberships for each observation, the function performs the k-sample test.
h	Bandwidth for the kernel function. If a value is not provided, the algorithm for the selection of an optimal h is performed automatically. See the function select_h for more details.
method	The method used for critical value estimation ("subsampling", "bootstrap", or "permutation")(default: "subsampling").
B	The number of iterations to use for critical value estimation (default: 150).
b	The size of the subsamples used in the subsampling algorithm (default: 0.8).
Quantile	The quantile to use for critical value estimation, 0.95 is the default value.
mu_hat	Mean vector for the reference distribution.

Sigma_hat	Covariance matrix of the reference distribution.
centeringType	String indicating the method used for centering the normal kernel ('Param' or 'Nonparam').
K_threshold	maximum number of groups allowed. Default is 10. It is a control parameter. Change in case of more than 10 samples.
alternative	Family of alternative chosen for selecting h, between "location", "scale" and "skewness" (only if h is not provided).
object	Object of class kb.test

Details

The function `kb.test` performs the kernel-based quadratic distance tests using the Gaussian kernel with bandwidth parameter `h`. Depending on the shape of the input `y` the function performs the tests of multivariate normality, the non-parametric two-sample tests or the `k`-sample tests.

Value

An S4 object of class `kb.test` containing the results of the kernel-based quadratic distance tests, based on the normal kernel. The object contains the following slots:

- `method`: String indicating the normal kernel-based quadratic distance test performed.
- `Dn` The value of the test statistic.
- `H0` A logical value indicating whether or not the null hypothesis is rejected.
- `data` Data list of samples `X` (and `Y`).
- `CV` The critical value computed for the test.
- `cv_method` The method used to estimate the critical value (one of "subsampling", "permutation" or "bootstrap").
- `h` List with the value of bandwidth parameter used for the normal kernel function. If `select_h` is used, the matrix of computed power values and the corresponding power plot are also provided.
- `B` Number of bootstrap/permutation/subsampling replications.

References

Markatou, M., Saraceno, G., Chen Y (2023). "Two- and k-Sample Tests Based on Quadratic Distances." Manuscript, (Department of Biostatistics, University at Buffalo)

Lindsay, B.G., Markatou, M. & Ray, S. (2014) "Kernels, Degrees of Freedom, and Power Properties of Quadratic Distance Goodness-of-Fit Tests", *Journal of the American Statistical Association*, 109:505, 395-410, DOI: 10.1080/01621459.2013.836972

Examples

```
# create a kb.test object
x <- matrix(rnorm(100),ncol=2)
y <- matrix(rnorm(100),ncol=2)
# Normality test
```

```

my_test <- kb.test(x, h=0.5)
my_test
# Two-sample test
my_test <- kb.test(x,y,h=0.5, method="subsampling",b=0.9,centeringType = "Nonparam")
my_test
# k-sample test
z <- matrix(rnorm(100,2),ncol=2)
dat <- rbind(x,y,z)
group <- rep(c(1,2,3),each=50)
my_test <- kb.test(x=dat,y=group,h=0.5, method="subsampling",b=0.9)
my_test

```

kb.test-class

An S4 class for kernel-based distance tests with normal kernel

Description

A class to represent the results of Gaussian kernel-based quadratic distance tests. This includes the normality test, the two-sample test statistics and the k-sample tests.

Slots

method String indicating the normal kernel-based quadratic distance test performed.

Dn The value of the test statistic.

H0 A logical value indicating whether or not the null hypothesis is rejected.

data List of samples X (and Y).

CV The critical value computed for the test.

cv_method The method used to estimate the critical value (one of "subsampling", "permutation" or "bootstrap").

h A list with the value of bandwidth parameter used for the Gaussian kernel. If the function `select_h` is used, then also the matrix of computed power values and the resulting power plot are provided.

B Number of bootstrap/permutation/subsampling replications.

Examples

```

# create a kb.test object
x <- matrix(rnorm(100),ncol=2)
y <- matrix(rnorm(100),ncol=2)
# Normality test
kb.test(x, h=0.5)

# Two-sample test
kb.test(x,y,h=0.5, method="subsampling",b=0.9)

```

pk.test	<i>Poisson kernel-based quadratic distance test of Uniformity on the sphere</i>
---------	---

Description

This function performs the kernel-based quadratic distance goodness-of-fit tests for Uniformity for spherical data using the Poisson kernel with concentration parameter ρ .

Usage

```
pk.test(x, rho = NULL, B = 300, Quantile = 0.95)

## S4 method for signature 'ANY'
pk.test(x, rho = NULL, B = 300, Quantile = 0.95)

## S4 method for signature 'pk.test'
show(object)
```

Arguments

x	A numeric d-dim matrix of data points on the Sphere S^{d-1} .
rho	Concentration parameter of the Poisson kernel function.
B	Number of iterations for critical value estimation of U_n (default: 300).
Quantile	The quantile to use for critical value estimation, 0.95 is the default value.
object	Object of class <code>pk.test</code>

Value

An S4 object of class `pk.test` containing the results of the Poisson kernel-based tests. The object contains the following slots:

- `method`: String indicating that the Poisson Kernel-based test is performed.
- `x` Data matrix.
- `Un` The value of the U-statistic.
- `CV_Un` The empirical critical value for U_n .
- `H0_Vn` A logical value indicating whether or not the null hypothesis is rejected according to U_n .
- `Vn` The value of the V-statistic.
- `CV_Vn` The critical value for V_n computed following the asymptotic distribution.
- `H0_Vn` A logical value indicating whether or not the null hypothesis is rejected according to V_n .
- `rho` The value of concentration parameter used for the Poisson kernel function.
- `B` Number of replications for the critical value of the U-statistic.

References

Ding, Y., Markatou, M., Saraceno, G. (2023). "Poisson Kernel-Based Tests for Uniformity on the d-Dimensional Sphere." *Statistica Sinica*. doi: doi:10.5705/ss.202022.0347

Examples

```
# create a pk.test object
x_sp <- sample_hypersphere(3, n_points=100)
unif_test <- pk.test(x_sp, rho=0.8)
unif_test
```

pk.test-class

An S4 class for Poisson kernel-based quadratic distance tests.

Description

A class to represent the results of Poisson kernel-based quadratic distance tests for Uniformity on the sphere.

Slots

method The method used for the test ("Poisson Kernel-based quadratic distance test of Uniformity on the Sphere").

x Matrix of data

Un The value of the U-statistic.

CV_Un The critical value for Un computed through replications.

H0_Un A logical value indicating whether or not the null hypothesis is rejected according to Un.

Vn The value of the V-statistic.

CV_Vn The critical value for Vn computed following the asymptotic distribution.

H0_Vn A logical value indicating whether or not the null hypothesis is rejected according to Vn.

rho The concentration parameter of the Poisson kernel.

B Number of replications.

Examples

```
# create a pk.test object
d=3
size=200
x_sp <- sample_hypersphere(d, n_points=size)
pk.test(x_sp, rho=0.8)
```

pkbc

*Poisson kernel-based clustering on the sphere***Description**

The function pkbc performs the Poisson kernel-based clustering algorithm on the sphere based on the Poisson kernel-based densities.

Usage

```
pkbc(
  dat,
  nClust,
  maxIter = 300,
  stoppingRule = "loglik",
  initMethod = "sampleData",
  numInit = 10
)

## S4 method for signature 'ANY'
pkbc(
  dat,
  nClust,
  maxIter = 300,
  stoppingRule = "loglik",
  initMethod = "sampleData",
  numInit = 10
)
```

Arguments

dat	Data matrix of observations to be clustered.
nClust	Number of clusters. It can be a single value or a numeric vector.
maxIter	The maximum number of iterations before a run is terminated.
stoppingRule	String describing the stopping rule to be used within each run. Currently must be either 'max' (until the change in the log-likelihood is less than a given threshold 10^{-7}), 'membership' (until the membership is unchanged), or 'loglik' (based on a maximum number of iterations).
initMethod	String describing the initialization method to be used. Currently must be 'sampleData'.
numInit	Number of initializations.

Details

The function estimates the parameter of a mixture of Poisson kernel-based densities. The obtained estimates are used for assigning final memberships, identifying the nClust clusters.

Value

An S4 object of class pkbc containing the results of the clustering procedure based on Poisson kernel-based distributions. The object contains the following slots:

res_k: List of results of the Poisson kernel-based clustering algorithm for each value of number of clusters specified in nClust. Each object in the list contains:

- postProbs Posterior probabilities of each observation for the indicated clusters.
- LogLik Maximum value of log-likelihood function
- wcss Values of within-cluster sum of squares computed with Euclidean distance and cosine similarity, respectively.
- params List of estimated parameters of the mixture model
 - mu estimated centroids
 - rho estimated concentration parameters rho
 - alpha estimated mixing proportions
- finalMemb Vector of final memberships
- runInfo List of information of the EM algorithm iterations
 - logLikVec vector of log-likelihood values
 - numIterPerRun number of E-M iterations per run

input: List of input information.

References

Golzy, M., Markatou, M. (2020) Poisson Kernel-Based Clustering on the Sphere: Convergence Properties, Identifiability, and a Method of Sampling, *Journal of Computational and Graphical Statistics*, 29:4, 758-770, DOI: 10.1080/10618600.2020.1740713.

Examples

```
#We generate three samples of 100 observations from 3-dimensional
#Poisson kernel-based densities with rho=0.8 and different mean directions
size <- 100
groups<-c(rep(1, size), rep(2, size),rep(3,size))
rho=0.8
set.seed(081423)
data1<-rpkb(size, c(1,0,0),rho,method='rejvmf')
data2<-rpkb(size, c(0,1,0),rho,method='rejvmf')
data3<-rpkb(size, c(-1,0,0),rho,method='rejvmf')
dat<-rbind(data1$x,data2$x, data3$x)

#The following code can be used for plotting

library("rgl")
plot3d(dat,col = groups, size=4,xlab = "", ylab = "",zlab = "")
legend3d("right", legend = c("Class 1", "Class 2", "Class 3"), col = 1:3, pch = 20)

#Perform the clustering algorithm with number od clusters k=3.
pkbd<- pkbc(dat, 3)
```

pkbc-class	<i>A S4 class for the clustering algorithm on the sphere based on Poisson kernel-based distributions.</i>
------------	---

Description

A class to represent the results of Poisson kernel-based clustering procedure for spherical observations.

Details

See the function pkbc for more details.

Slots

res_k List of objects with the results of the clustering algorithm for each value of possible number of clusters considered.

input List of input data

Examples

```
data("wireless")
res <- pkbc(as.matrix(wireless[, -8]), 4)
```

sample_hypersphere	<i>Generate random sample from the hypersphere</i>
--------------------	--

Description

Generate random sample from the hypersphere

Usage

```
sample_hypersphere(d, n_points = 1)
```

Arguments

d Number of dimensions.
n_points Number of sampled observations.

Value

Data matrix with the sampled observations.

Examples

```
x_sp <- sample_hypersphere(3,100)
```

select_h	<i>Select the value of the kernel tuning parameter</i>
----------	--

Description

This function computes the kernel bandwidth of the Gaussian kernel for the two-sample and k-sample kernel-based quadratic distance (KBQD) tests.

Usage

```
select_h(
  x,
  y = NULL,
  alternative = NULL,
  method = "subsampling",
  b = 0.8,
  B = 100,
  delta_dim = 1,
  delta = NULL,
  h_values = NULL,
  Nrep = 50,
  n_cores = 2,
  Quantile = 0.95,
  power.plot = TRUE
)
```

Arguments

x	Data set of observations from X.
y	Data set of observations from Y.
alternative	Family of alternative chosen for selecting h, between "location", "scale" and "skewness".
method	The method used for critical value estimation ("subsampling", "bootstrap", or "permutation").
b	The size of the subsamples used in the subsampling algorithm .
B	The number of iterations to use for critical value estimation, B = 150 as default.
delta_dim	Vector of coefficient of alternative with respect to each dimension
delta	Vector of parameter values indicating chosen alternatives
h_values	Values of the tuning parameter used for the selection
Nrep	Number of bootstrap/permutation/subsampling replications.

n_cores	Number of cores used to parallel the h selection algorithm (default:2).
Quantile	The quantile to use for critical value estimation, 0.95 is the default value.
power.plot	Logical. If TRUE, it is displayed the plot of power for values in h_values and delta.

Details

The function performs the selection of the optimal value for the tuning parameter h of the normal kernel function, for the two-sample and k-sample KBQD tests. It performs a small simulation study, generating sample according to the family of alternative specified, for the chosen values of h_values and delta.

Value

A list with the following attributes:

- h_sel the selected value of tuning parameter h;
- power matrix of power values computed for the considered values of delta and h_values;
- power.plot power plots (if power.plot is TRUE).

References

Markatou, M., Saraceno, G., Chen, Y. (2023). "Two- and k-Sample Tests Based on Quadratic Distances." Manuscript, (Department of Biostatistics, University at Buffalo)

Examples

```
# Select the value of h using the mid-power algorithm

x <- matrix(rnorm(100),ncol=2)
y <- matrix(rnorm(100),ncol=2)
h_sel <- select_h(x,y,"skewness")
h_sel
```

summary,kb.test-method

Summarizing kernel-based quadratic distance results

Description

summary method for the class kb.test

Usage

```
## S4 method for signature 'kb.test'
summary(object)
```

Arguments

object Object of class kb.test

Value

List with the following components:

- `summary_tables` Table of computed descriptive statistics per variable (and per group if available).
- `test_results` Data frame with the results of the performed kernel-based quadratic distance test.
- `qqplots` Figure with qq-plots for each variable.

Examples

```
# create a kb.test object
x <- matrix(rnorm(100),ncol=2)
y <- matrix(rnorm(100),ncol=2)
# Normality test
my_test <- kb.test(x, h=0.5)
summary(my_test)
```

summary.pk.test-method

Summarizing kernel-based quadratic distance results

Description

summary method for the class pk.test

Usage

```
## S4 method for signature 'pk.test'
summary(object)
```

Arguments

object Object of class pk.test

Value

List with the following components:

- `summary_tables` Table of computed descriptive statistics per variable.
- `test_results` Data frame with the results of the performed Poisson kernel-based test.
- `qqplots` Figure with qq-plots for each variable against the uniform distribution.

Examples

```
# create a pk.test object
x_sp <- sample_hypersphere(3, n_points=100)
unif_test <- pk.test(x_sp, rho=0.8)
summary(unif_test)
```

summary_stat

Descriptive statistics for Poisson kernel-based clustering results.

Description

Method for objects of class pkbc which computes and displays some descriptive for each variable with respect to the detected groups. It also generates a plot of data points colored by the final membership.

Usage

```
summary_stat(object, k, true_label = NULL)
```

Arguments

object	Object of class pkbc.
k	Number of clusters to be used.
true_label	Vector of true memberships to clusters (default: NULL).

Details

The function computes mean, standard deviation, median, inter-quantile range, minimum and maximum for each variable in the data set given the final membership assigned by the clustering algorithm. If dimension is equal to 2 or 3, points are displayed on the circle and sphere, respectively. If dimension is greater than 3, the spherical Principal Component procedure proposed by Locantore et al., (1999) is applied for dimensionality reduction and the first three principal components are normalized and displayed on the sphere.

Value

List with computed descriptive statistics for each variable. For $d > 3$, the complete results from the PcaLocantore function (package rrcov) are also returned.

References

Locantore, N., Marron, J.S., Simpson, D.G. et al. (1999) "Robust principal component analysis for functional data." *Test* 8, 1–73. <https://doi.org/10.1007/BF02595862>

Examples

```

#We generate three samples of 100 observations from 3-dimensional
#Poisson kernel-based densities with rho=0.8 and different mean directions
size <- 100
groups<-c(rep(1, size), rep(2, size),rep(3,size))
rho=0.8
set.seed(081423)
data1<-rpkb(size, c(1,0,0),rho,method='rejvmf')
data2<-rpkb(size, c(0,1,0),rho,method='rejvmf')
data3<-rpkb(size, c(-1,0,0),rho,method='rejvmf')
data<-rbind(data1$x,data2$x, data3$x)

#Perform the clustering algorithm
pkbc_res<- pkbc(data, 3)
summary_stat(pkbc_res, 3)

```

validation

*Validation of Poisson kernel-based clustering results***Description**

Method for objects of class pkbc which displays the elbow plots for the given values of number of clusters and compute evaluation measures for clustering results.

Usage

```
validation(object, true_label = NULL, elbow.plot = TRUE, h = 1.5)
```

Arguments

object	Object of class pkbc
true_label	Vector of true membership to clusters (if available)
elbow.plot	Logical, if TRUE the function returns the elbow plots computed with the Euclidean distance and cosine similarity (default: TRUE).
h	Tuning parameter of the k-sample test. (default: 1.5)

Details

The function extracts the within-cluster sum of squares and displays the obtained elbow plots. The following evaluation measures are computed: k-sample test, In-Group Proportion. If true label are provided, ARI, Average Silhouette Width, Macro-Precision and Macro-Recall are computed.

Value

List with the following components:

- `metrics` Table of computed evaluation measures.
- `IGP` List of in-group proportions for each value of number of clusters specified.
- `wcss` Table of collected values of within-cluster sum of squares.

References

Kapp, A.V., Tibshirani, R. (2007) "Are clusters found in one dataset present in another dataset?", *Biostatistics*, 8(1), 9–31, <https://doi.org/10.1093/biostatistics/kxj029>

Rousseeuw, P.J. (1987) Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20, 53–65.

Examples

```
#We generate three sample of 100 observations from 3-dimensional
#Poisson kernel-based densities with rho=0.8 and different mean directions

size <- 20
groups<-c(rep(1, size), rep(2, size),rep(3,size))
rho=0.8
set.seed(081423)
data1<-rpkb(size, c(1,0,0),rho,method='rejvmf')
data2<-rpkb(size, c(0,1,0),rho,method='rejvmf')
data3<-rpkb(size, c(-1,0,0),rho,method='rejvmf')
data<-rbind(data1$x,data2$x, data3$x)

#Perform the clustering algorithm
pkbc_res<- pkbc(data, 2:4)
validation(pkbc_res)
```

wireless

Wireless Indoor Localization

Description

The wireless data frame has 2000 rows and 8 columns. The first 7 variables report the measurements of the Wi-Fi signal strength received from 7 Wi-Fi routers in an office location in Pittsburgh (USA). The last column indicates the class labels.

Usage

```
wireless
```

Format

A data frame containing the following columns:

- V1 Signal strength from router 1.
- V2 Signal strength from router 2.
- V3 Signal strength from router 3.
- V4 Signal strength from router 4.
- V5 Signal strength from router 5.
- V6 Signal strength from router 6.
- V7 Signal strength from router 7.
- V8 Group memberships, from 1 to 4.

Details

The Wi-Fi signal strength is measured in dBm, decibel milliwatts, which is expressed as a negative value ranging from -100 to 0. The labels correspond to 4 different rooms. In total, we have 4 groups with 500 observations each.

Source

Bhatt,Rajen. (2017). Wireless Indoor Localization. UCI Machine Learning Repository.
<https://doi.org/10.24432/C51880>.

References

Rohra, J.G., Perumal, B., Narayanan, S.J., Thakur, P., Bhatt, R.B. (2017). "User Localization in an Indoor Environment Using Fuzzy Hybrid of Particle Swarm Optimization & Gravitational Search Algorithm with Neural Networks". In: Deep, K., et al. Proceedings of Sixth International Conference on Soft Computing for Problem Solving. Advances in Intelligent Systems and Computing, vol 546. Springer, Singapore. https://doi.org/10.1007/978-981-10-3322-3_27

Examples

```
data(wireless)
summary(wireless)
```

Index

* datasets

wireless, [18](#)

dpkb, [3](#)

kb.test, [4](#)

kb.test, ANY-method (kb.test), [4](#)

kb.test-class, [7](#)

pk.test, [8](#)

pk.test, ANY-method (pk.test), [8](#)

pk.test-class, [9](#)

pkbc, [10](#)

pkbc, ANY-method (pkbc), [10](#)

pkbc-class, [12](#)

QuadratiK-package, [2](#)

rpkb (dpkb), [3](#)

sample_hypersphere, [12](#)

select_h, [5](#), [13](#)

show, kb.test-method (kb.test), [4](#)

show, pk.test-method (pk.test), [8](#)

summary, kb.test-method, [14](#)

summary, pk.test-method, [15](#)

summary_stat, [16](#)

validation, [17](#)

wireless, [18](#)