

Package ‘clues’

October 14, 2016

Version 0.5.9

Date 2016-10-13

Title Clustering Method Based on Local

Author Fang Chang [aut, ctb],
Weiliang Qiu [aut, cre],
Vincent Carey [aut, ctb],
Ruben H. Zamar [aut, ctb],
Ross Lazarus [aut, ctb],
Xiaogang Wang [aut, ctb]

Maintainer Weiliang Qiu <stwxq@channing.harvard.edu>

Depends R (>= 3.1.0), stats, utils, graphics

Suggests cluster

Description We developed the clues R package to provide functions for automatically estimating the number of clusters and getting the final cluster partition without any input parameter except the stopping rule for convergence. The package also provides functions to evaluate and compare the performances of partitions of a data set both numerically and graphically.

License GPL (>= 2)

Repository CRAN

Date/Publication 2016-10-14 08:16:08

NeedsCompilation yes

R topics documented:

adjustedRand	2
clues	3
clustering	5
compClust	6
Curve	8
get_CH	9

get_Silhouette	10
Maronna	11
plot.clues	12
plotAvgCurves	13
plotClusters	14
plotCurves	15
print.clues	17
Ring	17
shrinking	18
summary.clues	19
Vowel	20

Index	22
--------------	-----------

adjustedRand	<i>Calculate Agreement Indices Between Two Partitions for a Data Set</i>
--------------	--

Description

Calculate the five agreement indices: Rand index, Hubert and Arabie's adjusted Rand index, Morey and Agresti's adjusted Rand index, Fowlkes and Mallows's index, and Jaccard index, which measure the agreement between any two partitions for a data set.

Usage

```
adjustedRand(c11, c12, randMethod = c("Rand", "HA", "MA", "FM", "Jaccard"))
```

Arguments

c11	partition 1 of a data set.
c12	partition 2 of a data set. c12 must have the same length as c11, but could have different number of clusters.
randMethod	specifies the preferred external index to measure the agreement between the two partitions c11 and c12. Available indices are: "Rand", "HA" (Hubert and Arabie's adjusted Rand index), "MA" (Morey and Agresti's adjusted Rand index), "FM" (Fowlkes and Mallows's index), "Jaccard" (Jaccard index). By default, all 5 indices will be output.

Value

Returns a vector of the index values.

References

- Milligan, G.W. and Cooper, M.C. (1986) A study of the comparability of external criteria for hierarchical cluster analysis. *Multivariate Behavioral Research* **21**, 441–458.
- Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```

c11 <- c(1, 1, 1, 2, 2, 2, 2, 2)
c12 <- c(1, 2, 1, 2, 1, 2, 1, 3)
adjustedRand(c11, c12)

# perfect agreement
c11 <- c(1, 1, 2, 2)
c12 <- c11
adjustedRand(c11, c12)

```

clues

Clustering Method Based on Local Shrinking

Description

Automatically estimate the number of clusters for a given data set and get a partition.

Usage

```

clues(y, n0 = 5, alpha = 0.05, eps = 1.0e-4, itmax = 20,
      K2.vec = NULL, strengthMethod = "sil", strengthIni = -3,
      disMethod = "Euclidean", quiet = TRUE)

```

Arguments

y	data matrix which is an R matrix object (for dimension > 1) or vector object (for dimension=1) with rows being observations and columns being variables.
n0	a guess for the number of clusters.
alpha	speed factor.
eps	a small positive number. A value is regarded as zero if it is less than eps.
itmax	maximum number of iterations allowed.
K2.vec	range for the number of nearest neighbors for the second pass of the iteration.
strengthMethod	specifies the preferred measure of the strength of the clusters (i.e., compactness of the clusters). Two available methods are "sil" (Silhouette index) and "CH" (CH index).
strengthIni	initial value for the lower bound of the measure of the strength for the clusters. Any negative values will do.
disMethod	specification of the dissimilarity measure. The available measures are "Euclidean" and "1-corr".
quiet	logical. Indicates if intermediate results should be output.

Value

K	number of nearest neighbors can be used to get final clustering.
size	vector of the number of data points for clusters.
mem	vector of the cluster membership of data points. The cluster membership takes values: 1, 2, . . . , g , where g is the estimated number of clusters.
g	an estimate of the number of clusters.
CH	CH index value for the final partition if strengthMethod is “CH”.
avg.s	average of the Silhouette index value for the final partition if strengthMethod is “sil”.
s	vector of Silhouette indices for data points if strengthMethod is “sil”.
K.vec	number of nearest neighbors used for each iteration.
g.vec	number of clusters obtained in each iteration.
myupdate	logical. Indicates if the partition obtained in the first pass is the same as that obtained in the second pass.
y.old1	data used for shrinking and clustering.
y.old2	data returned after shrinking and clustering.
y	a copy of the data from the input.
strengthMethod	a copy of the strengthMethod from the input.
disMethod	a copy of the dissimilarity measure from the input

Note

Occasionally, the number of clusters estimated by `clues` will be equal to the number of data points (that is, each data point forms a cluster). In this case, the estimated number of clusters was set to be equal to one. And the CH index or Silhouette index will be set to be equal to NULL since CH index and Silhouette index are not defined when the number of clusters is equal to one.

References

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
# Maronna data set
data(Maronna)
# data matrix
maronna <- Maronna$maronna

# partition by clues
res <- clues(maronna, quiet = TRUE)

# get summary statistics
summary(res)

# scatter plots and plot of trajectories
## Not run: plot(res)
```

clustering

Data Clustering (After Data Shrinking)

Description

Data clustering (after data shrinking).

Usage

```
clustering(y, disMethod = "Euclidean")
```

Arguments

<code>y</code>	data matrix which is an R matrix object (for dimension > 1) or vector object (for dimension=1) with rows be observations and columns be variables.
<code>disMethod</code>	specification of the dissimilarity measure. The available measures are “Euclidean” and “1-corr”.

Details

We first store the first observation (data point) in `point[1]`. We then get the nearest neighbor of `point[1]`. Store it in `point[2]`. Store the dissimilarity between `point[1]` and `point[2]` to `db[1]`. We next remove `point[1]`. We then find the nearest neighbor of `point[2]`. Store it in `point[3]`. Store the dissimilarity between `point[2]` and `point[3]` to `db[2]`. We then remove `point[2]` and find the nearest neighbor of `point[3]`. We repeat this procedure until we find `point[n]` and `db[n-1]` where `n` is the total number of data points.

Next, we calculate the interquartile range (IQR) of the vector `db`. We then check which elements of `db` are larger than $\text{avg} + 1.5\text{IQR}$ where `avg` is the average of the vector `db`. The minimum value of these outlier dissimilarities will be stored in `omin`. An estimate of the number of clusters is `g` where `g-1` is the number of the outlier dissimilarities. The position of an outlier dissimilarity indicates the end of a cluster and the start of a new cluster.

To get a reasonable clustering result, data sharpening (shrinking) is recommended before data clustering.

Value

<code>mem</code>	vector of the cluster membership of data points. The cluster membership takes values: 1, 2, . . . , <code>g</code> , where <code>g</code> is the estimated number of clusters.
<code>size</code>	vector of the number of data points for clusters.
<code>g</code>	an estimate of the number of clusters.
<code>db</code>	vector of dissimilarities between sorted consecutive data points (c.f. details).
<code>point</code>	vector of sorted consecutive data points (c.f. details).
<code>omin</code>	The minimum value of the outlier dissimilarities (c.f. details).

References

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
# Maronna data set
data(Maronna)
# data matrix
maronna <- Maronna$maronna

tt <- shrinking(maronna, K = 50, itmax = 20)
tt2 <- clustering(tt)

# Plot of dissimilarities between the sorted consecutive data points
# versus the sorted consecutive data points
# This plot can be used to assess the estimated number of clusters
db <- tt2$db
point <- tt2$point
n <- length(point)
plot(1:(n - 1), db, type = "l",
     xlab = "sorted consecutive data points",
     ylab = "dissimilarities between the sorted consecutive data points",
     xlim = c(0, n), axes = FALSE)
box()
axis(side = 2)
axis(side = 1, at = c(0, 1:(n - 1)), labels = point)
```

compClust

Compare different partitions for a data set

Description

Compare different partitions for a data set based on agreement indices, average silhouette index and CH index.

Usage

```
compClust(y, memMat, disMethod = "Euclidean")
```

Arguments

y	data matrix which is an R matrix object (for dimension > 1) or vector object (for dimension = 1) with rows being observations and columns being variables.
memMat	cluster membership matrix. Each column corresponds to a partition of the matrix y. The numbers of clusters for different partitions can be different. The cluster membership of a g-cluster data set should take values: 1, 2, ..., g.
disMethod	specification of the dissimilarity measure. The available measures are "Euclidean" and "1-corr".

Value

avg.s	a vector of average silhouette indices for the different partitions in memMat.
CH	a vector of CH indices for the different partitions in memMat.
Rand	a matrix of Rand indices measuring the pair-wise agreement among the different partitions in memMat.
HA	a matrix of Hubert and Arabie's adjusted Rand indices measuring the pair-wise agreement among the different partitions in memMat.
MA	a matrix of Morey and Agresti's adjusted Rand indices measuring the pair-wise agreement among the different partitions in memMat.
FM	a matrix of Fowlkes and Mallows's indices measuring the pair-wise agreement among the different partitions in memMat.
Jaccard	a matrix of Jaccard indices measuring the pair-wise agreement among the different partitions in memMat.

References

- Calinski, R.B., Harabasz, J., (1974). A dendrite method for cluster analysis. *Communications in Statistics*, Vol. 3, pages 1-27.
- Kaufman, L., Rousseeuw, P.J., (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York.
- Milligan, G.W. and Cooper, M.C. (1986) A study of the comparability of external criteria for hierarchical cluster analysis. *Multivariate Behavioral Research* **21**, 441–458.
- Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
# Maronna data set
data(Maronna)
# data matrix
maronna <- Maronna$maronna
# cluster membership
maronna.mem <- Maronna$maronna.mem

# partition by clues and kmeans
res_CH <- clues(maronna, strengthMethod = "CH", quiet = TRUE)
res_sil <- clues(maronna, strengthMethod = "sil", quiet = TRUE)
res_km_HW <- kmeans(maronna, 4, algorithm = "Hartigan-Wong")
res_km_L <- kmeans(maronna, 4, algorithm = "Lloyd")
res_km_F <- kmeans(maronna, 4, algorithm = "Forgy")
res_km_M <- kmeans(maronna, 4, algorithm = "MacQueen")

memMat <- cbind(maronna.mem, res_CH$mem, res_sil$mem,
               res_km_HW$cluster, res_km_L$cluster,
               res_km_F$cluster, res_km_M$cluster)
```

```
colnames(memMat) <- c("true", "clues_CH", "clues_sil",
  "km_HW", "km_L", "km_F", "km_M")

res <- compClust(maronna, memMat)

print(sapply(res, function(x) {round(x,1)}))
```

Curve

Curve Data Set

Description

A toy example used to illustrate curve clustering.

Usage

```
data(Curve)
```

Format

A list contains a 300 by 10 data matrix (`curve`) and a 300 by 1 cluster membership vector (`curve.mem`). There are 3 clusters, each containing 100 data points, respectively, in a 10-dimensional space.

Details

cluster one is generated from the model:

$$y_{ik} = \sin(2 * PI * x_k) + e_{ik}, x_k \sim N(0, 1), e_{ik} \sim N(0, 0.1), i = 1, \dots, 100, k = 1, \dots, 10.$$

cluster two is generated from the model:

$$y_{ik} = \cos(2 * PI * x_k) + e_{ik}, x_k \sim N(0, 1), e_{ik} \sim N(0, 0.1), i = 1, \dots, 100, k = 1, \dots, 10.$$

cluster three is generated from the model:

$$y_{ik} = e_{ik}, e_{ik} \sim N(0, 1), i = 1, \dots, 100, k = 1, \dots, 10.$$

Examples

```
data(Curve)

# data matrix
curve <- Curve$curve

# 'true' cluster membership
curve.mem <- Curve$curve.mem
# 'true' number of clusters
nClust <- length(unique(curve.mem))

# plot average trajectories
plotAvgCurves(curve, curve.mem)
```

get_CH	<i>Compute CH Index</i>
--------	-------------------------

Description

Compute CH index for a given partition of a data set.

Usage

```
get_CH(y, mem, disMethod = "Euclidean")
```

Arguments

y	data matrix which is an R matrix object (for dimension > 1) or vector object (for dimension=1) with rows be observations and columns be variables.
mem	vector of the cluster membership of data points. The cluster membership takes values: 1, 2, . . . , g , where g is the estimated number of clusters.
disMethod	specification of the dissimilarity measure. The available measures are “Euclidean” and “1-corr”.

Value

The value of the CH index.

References

Calinski, R.B., Harabasz, J., (1974). A dendrite method for cluster analysis. *Communications in Statistics*, Vol. 3, pages 1-27.

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
data(Maronna)
# data matrix
maronna <- Maronna$maronna

# cluster membership
maronna.mem <- Maronna$maronna.mem

get_CH(maronna, maronna.mem)
```

<code>get_Silhouette</code>	<i>Compute Silhouette Index</i>
-----------------------------	---------------------------------

Description

Compute Silhouette index for a given partition of a data set.

Usage

```
get_Silhouette(y, mem, disMethod = "Euclidean")
```

Arguments

<code>y</code>	data matrix which is an R matrix object (for dimension > 1) or vector object (for dimension = 1) with rows be observations and columns be variables.
<code>mem</code>	vector of the cluster membership of data points. The cluster membership takes values: 1, 2, . . . , g , where g is the estimated number of clusters.
<code>disMethod</code>	specification of the dissimilarity measure. The available measures are “Euclidean” and “1-corr”.

Value

A list of 3 elements:

<code>avg.s</code>	average Sihouette index.
<code>s</code>	vector of Sihouette indices for data points.
<code>neighbor</code>	a vector, the i -th element of which indicates which cluster is the nearest neighbor cluster of the i -th data point.

References

Kaufman, L., Rousseeuw, P.J., (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York.

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
data(Maronna)
# data matrix
maronna <- Maronna$maronna

# cluster membership
maronna.mem <- Maronna$maronna.mem

tt <- get_Silhouette(maronna, maronna.mem)
tt$avg.s
```

Maronna

The Maronna Data Set

Description

The Maronna data set.

Usage

```
data(Maronna)
```

Value

A list contains a 200 by 2 data matrix and a 200 by 1 cluster membership vector. There are 4 equally distributed clusters in a two-dimensional space. The first cluster consists of the first 50 data points, the second cluster corresponds to the next 50 data points and so on.

Note

We generated this data set by using the models and parameter settings described in Maronna and Jacovkis (1974).

References

Maronna, R., Jacovkis, P.M, 1974. Multivariate clustering procedures with variable metrics. *Biometrics* 30, 499-505.

Examples

```
data(Maronna)
# data matrix
maronna <- Maronna$maronna

# cluster membership
maronna.mem <- Maronna$maronna.mem
# 'true' number of clusters
nClust <- length(unique(maronna.mem))

# scatter plots
plotClusters(maronna, maronna.mem)
```

plot.clues

Draw Scatter Plots or Plot of Average Trajectory per Cluster, or Both

Description

Draw scatter plots, or plot of average trajectory per cluster, or both. A simple menu will prompt for the user to choose what will be plotted.

Usage

```
## S3 method for class 'clues'
plot(
  x, ask = TRUE, plot.dim = NULL,
  xlab = NULL, ylab = NULL,
  xlim = NULL, ylim = NULL, cex = NULL,
  las = NULL, lwd = NULL,
  xlab.avg.curve = "variable",
  ylab.avg.curve = "average observation", ...)
```

Arguments

x	an object returned by the function <code>clues</code> .
ask	logical; if true, <code>plot.clues</code> operates in interactive mode, via menu .
plot.dim	specifies the dimensions to be plot in pair-wise scatter plots. The number of dimensions specified can be more than 2. If <code>plot.dim</code> is NULL, then pair-wise scatter plots of all dimensions will be plotted.
xlab	a title for the x axis. If <code>xlab = NULL</code> , then <code>xlab</code> will be set to be "".
ylab	a title for the y-axis. If <code>ylab = NULL</code> , then <code>ylab</code> will be set to be "".
xlim	range of x-axis. If <code>xlim = NULL</code> , then <code>xlim</code> will be set to be the range of the matrix <code>y</code> in the dimensions specified by <code>plot.dim</code> .
ylim	range of y-axis. If <code>ylim = NULL</code> , then <code>ylim</code> will be set to be the range of the matrix <code>y</code> in the dimensions specified by <code>plot.dim</code> .
cex	A numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. (see par). If <code>cex = NULL</code> , then <code>cex</code> will be set to be 2.
las	numeric in 0, 1, 2, 3; the style of axis labels (see par . 0 means that the labels always parallel to the axis [default]; 1 means that the labels always horizontal; 2 means that the labels always perpendicular to the axis, 3 means that the labels always vertical. If <code>las = NULL</code> , then <code>las</code> will be set to be 2.
lwd	the line width, a positive number, defaulting to 1 (see par . If <code>lwd = NULL</code> , the <code>lwd</code> will be set to be 3.
xlab.avg.curve	label for x-axis in the plot of average trajectory per cluster.
ylab.avg.curve	label for y-axis in the plot of average trajectory per cluster.
...	graphical parameters (see par).

Examples

```

# Maronna data set
data(Maronna)
# data matrix
maronna <- Maronna$maronna
# apply clues to maronna data set
res <- clues(maronna)
# plot
## Not run: plot(res)

```

plotAvgCurves

Average Trajectory Plot for Each Cluster

Description

Average trajectory plot for each cluster.

Usage

```

plotAvgCurves(y, mem, xlab = NULL, ylab = NULL,
              xlim = NULL, ylim = NULL, las = NULL, lwd = NULL, ...)

```

Arguments

<code>y</code>	data matrix which is an R matrix object (for dimension > 1) or vector object (for dimension=1) with rows being observations and columns being variables.
<code>mem</code>	vector of the cluster membership of data points. The cluster membership takes values: 1, 2, ..., g , where g is the estimated number of clusters.
<code>xlab</code>	a title for the x axis. If <code>xlab = NULL</code> , then <code>xlab</code> will be set to be “variable”.
<code>ylab</code>	a title for the y-axis. If <code>ylab = NULL</code> , then <code>ylab</code> will be set to be “average observation”.
<code>xlim</code>	range of x-axis. If <code>xlim = NULL</code> , then <code>xlim</code> will be set to be $(0, nc + 1)$, where nc is the number of columns of the matrix <code>y</code> .
<code>ylim</code>	range of y-axis. If <code>ylim = NULL</code> , then <code>ylim</code> will be set to be the range of the matrix <code>y</code> .
<code>las</code>	numeric in 0, 1, 2, 3; the style of axis labels (see par . 0 means that the labels always parallel to the axis [default]; 1 means that the labels always horizontal; 2 means that the labels always perpendicular to the axis, 3 means that the lables always vertical. If <code>las = NULL</code> , then <code>las</code> will be set to be 2.
<code>lwd</code>	the line width, a positive number, defaulting to 1 (see par . If <code>lwd = NULL</code> , the <code>lwd</code> will be set to be 3.
<code>...</code>	graphical parameters (see par).

References

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
data(Curve)

# data matrix
curve <- Curve$curve

# 'true' cluster membership
curve.mem <- Curve$curve.mem

# 'true' number of clusters
nClust <- length(unique(curve.mem))

# plot average trajectories
plotAvgCurves(curve, curve.mem)
```

plotClusters

Scatter Plots of Clusters in Specified Dimensions

Description

Scatter plots of clusters in specified dimensions. If more than two dimensions are specified, pair-wise scatter plots will be plotted.

Usage

```
plotClusters(y, mem, plot.dim = NULL,
             xlab = NULL, ylab = NULL,
             xlim = NULL, ylim = NULL, cex = NULL,
             cex.points = 1, ...)
```

Arguments

y	data matrix with rows being a collection of observations and columns being a list of variables.
mem	vector of the cluster membership of data points. The cluster membership takes values: 1, 2, ..., g, where g is the estimated number of clusters.
plot.dim	specifies the dimensions to be plot. If plot.dim is NULL, then pair-wise scatter plots of all dimensions will be plotted.
xlab	a title for the x axis. If xlab = NULL, then xlab will be set to "".
ylab	a title for the y-axis. If ylab = NULL, then ylab will be set to "".

xlim	range of x-axis. If xlim = NULL, then xlim will be set to be the range of the matrix y in the dimensions specified by plot.dim.
ylim	range of y-axis. If ylim = NULL, then ylim will be set to be the range of the matrix y in the dimensions specified by plot.dim.
cex	A numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. (see par). If cex = NULL, then cex will be set to be 2.
cex.points	A numerical value indicating the pointsize for points in clusters 2, 3, ..., g. Note points in cluster 1 won't be affected. This will help to visualize the overlap among clusters.
...	graphical parameters (see par).

References

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
# Maronna data set
data(Maronna)

# data matrix
maronna <- Maronna$maronna

# cluster membership
maronna.mem <- Maronna$maronna.mem

plotClusters(maronna, maronna.mem)
```

plotCurves

Plot Trajectories for Each Cluster

Description

Plot trajectories for each cluster.

Usage

```
plotCurves(y, mem, xlab = NULL, ylab = NULL,
            xlim = NULL, ylim = NULL,
            las = NULL, lwd = NULL, ...)
```

Arguments

<code>y</code>	data matrix which is an R matrix object (for dimension > 1) or vector object (for dimension=1) with rows being observations and columns being variables.
<code>mem</code>	vector of the cluster membership for data points. The cluster membership takes values: 1, 2, . . . , g , where g is the estimated number of clusters.
<code>xlab</code>	a title for the x axis. If <code>xlab = NULL</code> , then <code>xlab</code> will be set to be “variable”.
<code>ylab</code>	a title for the y-axis. If <code>ylab = NULL</code> , then <code>ylab</code> will be set to be “average observation”.
<code>xlim</code>	range of x-axis. If <code>xlim = NULL</code> , then <code>xlim</code> will be set to be $(0, nc + 1)$, where nc is the number of columns of the matrix <code>y</code> .
<code>ylim</code>	range of y-axis. If <code>ylim = NULL</code> , then <code>ylim</code> will be set to be the range of the matrix <code>y</code> .
<code>las</code>	numeric in 0, 1, 2, 3; the style of axis labels (see <code>par</code> . 0 means that the labels always parallel to the axis [default]; 1 means that the labels always horizontal; 2 means that the labels always perpendicular to the axis, 3 means that the labels always vertical. If <code>las = NULL</code> , then <code>las</code> will be set to be 2.
<code>lwd</code>	the line width, a positive number, defaulting to 1 (see <code>par</code> . If <code>lwd = NULL</code> , the <code>lwd</code> will be set to be 3.
<code>...</code>	graphical parameters (see <code>par</code>).

References

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
data(Curve)

# data matrix
curve <- Curve$curve

# 'true' cluster membership
curve.mem <- Curve$curve.mem
# 'true' number of clusters
nClust <- length(unique(curve.mem))

# plot average trajectories
plotCurves(curve, curve.mem)
```

`print.clues`*Print Method for CLUES Objects*

Description

Prints the number of data points, number of variables, number of clusters, cluster sizes, strength method, strength value, dissimilarity measurement, components of the object returned by `clues`.

This is a method for the function `print()` for objects inheriting from class `clues`.

Usage

```
## S3 method for class 'clues'  
print(x, ...)
```

Arguments

`x` a `clues` object.
`...` potential further arguments (required by generic).

See Also

[clues](#), [summary.clues](#).

Examples

```
# Maronna data set  
data(Maronna)  
# data matrix  
maronna <- Maronna$maronna  
# apply clues to maronna data set  
res <- clues(maronna)  
  
print(res)
```

`Ring`*The Broken-Ring Data Set*

Description

The broken-ring data set studied in Wang et al. (2007).

Usage

```
data(Ring)
```

Value

A list contains a 800 by 2 data matrix and a 800 by 1 cluster membership vector. There are 5 clusters containing 200, 157, 151, 151 and 141 data points, respectively, in a two-dimensional space.

References

Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
data(Ring)

# data matrix
ring <- Ring$ring

# 'true' cluster membership
ring.mem <- Ring$ring.mem

# 'true' number of clusters
nClust <- length(unique(ring.mem))

# scatter plots
plotClusters(ring, ring.mem)
```

 shrinking

Data Sharpening Based on K-nearest Neighbors

Description

Data sharpening based on K-nearest neighbors.

Usage

```
shrinking(y, K, disMethod = "Euclidean", eps = 1e-04, itmax = 20)
```

Arguments

y	data matrix with rows being the observations and columns being variables.
K	number of nearest neighbors.
disMethod	specification of the dissimilarity measure. The available measures are “Euclidean” and “1-corr”.
eps	a small positive number. A value is regarded as zero if it is less than eps.
itmax	maximum number of iterations allowed.

Details

Within each iteration, each data point is replaced by the vector of the coordinate-wise medians of its K nearest neighbors. Data points will move toward the locally most dense data point by this shrinking process.

Value

Sharpened data set.

Examples

```
# Maronna data set
data(Maronna)
# data matrix
maronna <- Maronna$maronna

# cluster membership
maronna.mem <- Maronna$maronna.mem

tt <- shrinking(maronna, K = 5, itmax=1)
plotClusters(tt, maronna.mem)
```

summary.clues

Summary Method for CLUES Objects

Description

Summarize a [clues](#) object and return an object of class `summary.clues`. There's a [print](#) method for the latter.

Usage

```
## S3 method for class 'clues'
summary(object, ...)
## S3 method for class 'summary.clues'
print(x, ...)
```

Arguments

`x`, `object` a [clues](#) object.
`...` potential further arguments (require by generic).

See Also

[clues](#), [print.clues](#).

Examples

```
# Maronna data set
data(Maronna)
# data matrix
maronna <- Maronna$maronna
# apply clues to maronna data set
res <- clues(maronna)

summary(res)
```

Vowel

The Vowel Data Set

Description

The vowel data set studied in Hastie, Tibshirani and Friedman (2001).

Usage

```
data(Vowel)
```

Details

In the original vowel data set, the 11 different words, with each one representing a vowel sound, correspond to 11 different clusters. There are 528 training observations and 462 testing observations. The training observations are employed to assess the performance of the clues algorithm in Wang et al. (2007) and referred as the Vowel data set. The 10 dimensional data was projected into a 2-dimensional space for examination purpose according to the method described in Hastie, Tibshirani and Friedman (pages 84, 92, 2001).

Value

A list consists of a 528 by 2 data matrix and a 528 by 1 cluster membership vector. There are 11 clusters each containing 48 data points in a two-dimensional space.

References

Hastie, T., Tibshirani, R., Friedman, J. (2001) *The elements of statistical learning*. Springer-Verlag.
Wang, S., Qiu, W., and Zamar, R. H. (2007). CLUES: A non-parametric clustering method based on local shrinking. *Computational Statistics & Data Analysis*, Vol. 52, issue 1, pages 286-298.

Examples

```
data(Vowel)

# data matrix
vowel <- Vowel$vowel
```

```
# 'true' cluster membership  
vowel.mem <- Vowel$vowel.mem  
  
# scatter plots  
plotClusters(vowel, vowel.mem)
```

Index

*Topic **cluster**

- adjustedRand, 2
- clues, 3
- clustering, 5
- compClust, 6
- get_CH, 9
- get_Silhouette, 10
- Maronna, 11
- plot.clues, 12
- plotAvgCurves, 13
- plotClusters, 14
- plotCurves, 15
- print.clues, 17
- Ring, 17
- shrinking, 18
- summary.clues, 19
- Vowel, 20

*Topic **datasets**

- Curve, 8

*Topic **print**

- print.clues, 17

adjustedRand, 2

clues, 3, 17, 19

clustering, 5

compClust, 6

Curve, 8

curve (Curve), 8

get_CH, 9

get_Silhouette, 10

Maronna, 11

maronna (Maronna), 11

menu, 12

par, 12, 13, 15, 16

plot.clues, 12

plotAvgCurves, 13

plotClusters, 14

plotCurves, 15

print, 17, 19

print.clues, 17, 19

print.summary.clues (summary.clues), 19

Ring, 17

ring (Ring), 17

shrinking, 18

summary.clues, 17, 19

Vowel, 20

vowel (Vowel), 20