Package 'T4cluster'

September 23, 2020

Type Package

Title Tools for Cluster Analysis

Version 0.1.0

Encoding UTF-8

Description Cluster analysis is one of the most fundamental problems in data science. We provide a variety of algorithms from clustering to the learning on the space of partitions. See Hennig, Meila, and Rocci (2016, ISBN:9781466551886) for general exposition to cluster analysis.

License MIT + file LICENSE

Imports Rcpp (>= 1.0.5), Rdpack, Rdimtools, maotai, stats, utils

URL http://kyoustat.com/T4cluster/

BugReports https://github.com/kyoustat/T4cluster/issues

LinkingTo Rcpp, RcppArmadillo

RdMacros Rdpack

RoxygenNote 7.1.1

NeedsCompilation yes

Author Kisung You [aut, cre] (<https://orcid.org/0000-0002-8584-459X>)

Maintainer Kisung You <kyoustat@gmail.com>

Repository CRAN

Date/Publication 2020-09-23 08:30:02 UTC

R topics documented:

`kmeans` *K-Means Clustering*

Description

K-means algorithm we provide is a wrapper to the **Armadillo**'s k-means routine. Two types of initialization schemes are employed. Please see the parameters section for more details.

Usage

```
kmeans(data, k = 2, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations.
`k`	the number of clusters (default: 2).
`...`	extra parameters including
	init initialization method; either `"random"` for random initialization, or `"plus"` for k-means++ starting.
	maxiter the maximum number of iterations (default: 10).
	nstart the number of random initializations (default: 5).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1 : k$).

mean a $(k \times p)$ matrix where each row is a class mean.

wcss within-cluster sum of squares (WCSS).

algorithm name of the algorithm.

References

Sanderson C, Curtin R (2016). "Armadillo: a template-based C++ library for linear algebra." *The Journal of Open Source Software*, **1**(2), 26. ISSN 2475-9066, doi: 10.21105/joss.00026.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = kmeans(X, k=2)$cluster
cl3 = kmeans(X, k=3)$cluster
cl4 = kmeans(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="k-means: k=2")
plot(X2d, col=cl3, pch=19, main="k-means: k=3")
plot(X2d, col=cl4, pch=19, main="k-means: k=4")
par(opar)
```

---

kmeanspp *K-Means++ Clustering*

Description

K-means++ algorithm is usually used as a fast initialization scheme, though it can still be used as a standalone clustering algorithms by first choosing the centroids and assign points to the nearest centroids.

Usage

```
kmeanspp(data, k = 2)
```

Arguments

data	an $(n \times p)$ matrix of row-stacked observations.
k	the number of clusters (default: 2).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1:k$).

algorithm name of the algorithm.

References

Arthur D, Vassilvitskii S (2007). "K-Means++: The advantages of careful seeding." In *Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms*, SODA '07, 1027–1035. ISBN 978-0-89871-624-5, Number of pages: 9 Place: New Orleans, Louisiana.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = kmeanspp(X, k=2)$cluster
cl3 = kmeanspp(X, k=3)$cluster
cl4 = kmeanspp(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="k-means++: k=2")
plot(X2d, col=cl3, pch=19, main="k-means++: k=3")
plot(X2d, col=cl4, pch=19, main="k-means++: k=4")
par(opar)
```

---

pcm *Compute Pairwise Co-occurrence Matrix*

Description

Let *clustering* be a label from data of N observations and suppose we are given M such labels. Co-occurrent matrix counts the number of events where two observations X_i and X_j belong to the same category/class. *PCM* serves as a measure of uncertainty embedded in any algorithms with non-deterministic components.

Usage

```
pcm(partitions)
```

Arguments

partitions	partitions can be provided in either (1) an $(M \times N)$ matrix where each row is a clustering for N objects, or (2) a length-M list of length-N clustering labels.

Value

an $(N \times N)$ matrix, whose elements (i, j) are counts for how many times observations i and j belong to the same cluster, ranging from 0 to M.

See Also

psm

Examples

```
# ------------------------------------------------------------
#               PSM with 'iris' dataset + k-means++
# ------------------------------------------------------------
## PREPARE WITH SUBSET OF DATA
data(iris)
X     = as.matrix(iris[,1:4])
lab   = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## RUN K-MEANS++ 100 TIMES
partitions = list()
for (i in 1:100){
  partitions[[i]] = kmeanspp(X)$cluster
}

## COMPUTE PCM
iris.pcm = pcm(partitions)

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,2), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
image(iris.pcm[,150:1], axes=FALSE, main="PCM")
par(opar)
```

psm *Compute Posterior Similarity Matrix*

Description

Let *clustering* be a label from data of N observations and suppose we are given M such labels. Posterior similarity matrix, as its name suggests, computes posterior probability for a pair of observations to belong to the same cluster, i.e.,

$$P_{ij} = P(\text{label}(X_i) = \text{label}(X_j))$$

under the scenario where multiple clusterings are samples drawn from a posterior distribution within the Bayesian framework. However, it can also be used for non-Bayesian settings as `psm` is a measure of uncertainty embedded in any algorithms with non-deterministic components.

Usage

```
psm(partitions)
```

Arguments

`partitions`	partitions can be provided in either (1) an $(M \times N)$ matrix where each row is a clustering for N objects, or (2) a length-M list of length-N clustering labels.

Value

an $(N \times N)$ matrix, whose elements (i, j) are posterior probability for an observation i and j belong to the same cluster.

See Also

`pcm`

Examples

```
# ------------------------------------------------------------
#               PSM with 'iris' dataset + k-means++
# ------------------------------------------------------------
## PREPARE WITH SUBSET OF DATA
data(iris)
X     = as.matrix(iris[,1:4])
lab   = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## RUN K-MEANS++ 100 TIMES
partitions = list()
for (i in 1:100){
```

```
  partitions[[i]] = kmeanspp(X)$cluster
}

## COMPUTE PSM
iris.psm = psm(partitions)

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,2), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
image(iris.psm[,150:1], axes=FALSE, main="PSM")
par(opar)
```

`sc05Z` *Spectral Clustering by Zelnik-Manor and Perona (2005)*

Description

Zelnik-Manor and Perona proposed a method to define a set of data-driven bandwidth parameters where σ_i is the distance from a point x_i to its nnbd-th nearest neighbor. Then the affinity matrix is defined as

$$A_{ij} = \exp(-d(x_i, d_j)^2/\sigma_i\sigma_j)$$

and the standard spectral clustering of Ng, Jordan, and Weiss (`scNJW`) is applied.

Usage

```
sc05Z(data, k = 2, nnbd = 7, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
`k`	the number of clusters (default: 2).
`nnbd`	neighborhood size to define data-driven bandwidth parameter (default: 7).
`...`	extra parameters including
	algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.
	maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1 : k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

Zelnik-manor L, Perona P (2005). "Self-tuning spectral clustering." In Saul LK, Weiss Y, Bottou L (eds.), *Advances in neural information processing systems 17*, 1601–1608. MIT Press. http://papers.nips.cc/paper/2619-self-tuning-spectral-clustering.pdf.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = sc05Z(X, k=2)$cluster
cl3 = sc05Z(X, k=3)$cluster
cl4 = sc05Z(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="sc05Z: k=2")
plot(X2d, col=cl3, pch=19, main="sc05Z: k=3")
plot(X2d, col=cl4, pch=19, main="sc05Z: k=4")
par(opar)
```

`sc09G` *Spectral Clustering by Gu and Wang (2009)*

Description

The algorithm defines a set of data-driven bandwidth parameters where σ_i is the average distance from a point x_i to its nnbd-th nearest neighbor. Then the affinity matrix is defined as

$$A_{ij} = \exp(-d(x_i, d_j)^2/\sigma_i\sigma_j)$$

and the standard spectral clustering of Ng, Jordan, and Weiss (scNJW) is applied.

Usage

```
sc09G(data, k = 2, nnbd = 7, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
`k`	the number of clusters (default: 2).
`nnbd`	neighborhood size to define data-driven bandwidth parameter (default: 7).
`...`	extra parameters including
	algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.
	maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1:k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

Gu R, Wang J (2009). "An Improved Spectral Clustering Algorithm Based on Neighbour Adaptive Scale." In *2009 International Conference on Business Intelligence and Financial Engineering*, 233–236. ISBN 978-0-7695-3705-4, doi: 10.1109/BIFE.2009.62.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = sc09G(X, k=2)$cluster
cl3 = sc09G(X, k=3)$cluster
cl4 = sc09G(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="sc09G: k=2")
```

```
plot(X2d, col=cl3, pch=19, main="sc09G: k=3")
plot(X2d, col=cl4, pch=19, main="sc09G: k=4")
par(opar)
```

`sc10Z` *Spectral Clustering by Zhang et al. (2010)*

Description

The algorithm defines a set of data-driven bandwidth parameters p_{ij} by constructing a similarity matrix. Then the affinity matrix is defined as

$$A_{ij} = \exp(-d(x_i, d_j)^2/2p_{ij}$$

and the standard spectral clustering of Ng, Jordan, and Weiss (`scNJW`) is applied.

Usage

```
sc10Z(data, k = 2, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
`k`	the number of clusters (default: 2).
`...`	extra parameters including
	algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.
	maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1 : k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

Zhang Y, Zhou J, Fu Y (2010). "Spectral clustering algorithm based on adaptive neighbor distance sort order." In *The 3rd International Conference on Information Sciences and Interaction Sciences*, 444–447. ISBN 978-1-4244-7384-7, doi: 10.1109/ICICIS.2010.5534786.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = sc10Z(X, k=2)$cluster
cl3 = sc10Z(X, k=3)$cluster
cl4 = sc10Z(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="sc10Z: k=2")
plot(X2d, col=cl3, pch=19, main="sc10Z: k=3")
plot(X2d, col=cl4, pch=19, main="sc10Z: k=4")
par(opar)
```

---

sc11Y *Spectral Clustering by Yang et al. (2011)*

Description

As a data-driven method, the algorithm recovers geodesic distance from a k-nearest neighbor graph scaled by an (exponential) parameter ρ and applies random-walk spectral clustering. Authors referred their method as density sensitive similarity function.

Usage

```
sc11Y(data, k = 2, nnbd = 7, rho = 2, ...)
```

Arguments

data	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
k	the number of clusters (default: 2).
nnbd	neighborhood size to define data-driven bandwidth parameter (default: 7).
rho	exponent scaling parameter (default: 2).
...	extra parameters including

algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.

maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1:k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

Yang P, Zhu Q, Huang B (2011). "Spectral clustering with density sensitive similarity function." *Knowledge-Based Systems*, **24**(5), 621–628. ISSN 09507051, doi: 10.1016/j.knosys.2011.01.009.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = sc11Y(X, k=2)$cluster
cl3 = sc11Y(X, k=3)$cluster
cl4 = sc11Y(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="sc11Y: k=2")
plot(X2d, col=cl3, pch=19, main="sc11Y: k=3")
plot(X2d, col=cl4, pch=19, main="sc11Y: k=4")
par(opar)
```

`sc12L` *Spectral Clustering by Li and Guo (2012)*

Description

Li and Guo proposed to construct an affinity matrix

$$A_{ij} = \exp(-d(x_i, d_j)^2/2\sigma^2)$$

and adjust the matrix by neighbor propagation. Then, standard spectral clustering from the symmetric, normalized graph laplacian is applied.

Usage

```
sc12L(data, k = 2, sigma = 1, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
`k`	the number of clusters (default: 2).
`sigma`	common bandwidth parameter (default: 1).
`...`	extra parameters including
	algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.
	maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1 : k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

Li X, Guo L (2012). "Constructing affinity matrix in spectral clustering based on neighbor propagation." *Neurocomputing*, **97**, 125–130. ISSN 09252312, doi: 10.1016/j.neucom.2012.06.023.

See Also

`scNJW`

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = sc12L(X, k=2)$cluster
cl3 = sc12L(X, k=3)$cluster
cl4 = sc12L(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="sc12L: k=2")
plot(X2d, col=cl3, pch=19, main="sc12L: k=3")
plot(X2d, col=cl4, pch=19, main="sc12L: k=4")
par(opar)
```

---

scNJW *Spectral Clustering by Ng, Jordan, and Weiss (2002)*

Description

The version of Ng, Jordan, and Weiss first constructs the affinity matrix

$$A_{ij} = \exp(-d(x_i, d_j)^2/\sigma^2)$$

where σ is a common bandwidth parameter and performs k-means (or possibly, GMM) clustering on the row-space of eigenvectors for the symmetric graph laplacian matrix

$$L = D^{-1/2}(D - A)D^{-1/2}$$

.

Usage

```
scNJW(data, k = 2, sigma = 1, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
`k`	the number of clusters (default: 2).
`sigma`	bandwidth parameter (default: 1).
`...`	extra parameters including
	algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.
	maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1:k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

Ng AY, Jordan MI, Weiss Y (2002). "On spectral clustering: Analysis and an algorithm." In Dietterich TG, Becker S, Ghahramani Z (eds.), *Advances in neural information processing systems 14*, 849–856. MIT Press. http://papers.nips.cc/paper/2092-on-spectral-clustering-analysis-and-an-algorithm.pdf.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = scNJW(X, k=2)$cluster
cl3 = scNJW(X, k=3)$cluster
cl4 = scNJW(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
```

```
plot(X2d, col=cl2, pch=19, main="scNJW: k=2")
plot(X2d, col=cl3, pch=19, main="scNJW: k=3")
plot(X2d, col=cl4, pch=19, main="scNJW: k=4")
par(opar)
```

---

scSM *Spectral Clustering by Shi and Malik (2000)*

Description

The version of Shi and Malik first constructs the affinity matrix

$$A_{ij} = \exp(-d(x_i, d_j)^2/\sigma^2)$$

where σ is a common bandwidth parameter and performs k-means (or possibly, GMM) clustering on the row-space of eigenvectors for the random-walk graph laplacian matrix

$$L = D^{-1}(D - A)$$

.

Usage

```
scSM(data, k = 2, sigma = 1, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
`k`	the number of clusters (default: 2).
`sigma`	bandwidth parameter (default: 1).
`...`	extra parameters including
	algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.
	maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1 : k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

Jianbo Shi, Malik J (2000). "Normalized cuts and image segmentation." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(8), 888–905. ISSN 01628828, doi: 10.1109/34.868688.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE WITH SUBSET OF DATA
data(iris)
sid = sample(1:150, 50)
X   = as.matrix(iris[sid,1:4])
lab = as.integer(as.factor(iris[sid,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = scSM(X, k=2)$cluster
cl3 = scSM(X, k=3)$cluster
cl4 = scSM(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="scSM: k=2")
plot(X2d, col=cl3, pch=19, main="scSM: k=3")
plot(X2d, col=cl4, pch=19, main="scSM: k=4")
par(opar)
```

---

scUL *Spectral Clustering with Unnormalized Laplacian*

Description

The version of Shi and Malik first constructs the affinity matrix

$$A_{ij} = \exp(-d(x_i, d_j)^2/\sigma^2)$$

where σ is a common bandwidth parameter and performs k-means (or possibly, GMM) clustering on the row-space of eigenvectors for the unnormalized graph laplacian matrix

$$L = D - A$$

.

Usage

```
scUL(data, k = 2, sigma = 1, ...)
```

Arguments

`data`	an $(n \times p)$ matrix of row-stacked observations or S3 `dist` object of n observations.
`k`	the number of clusters (default: 2).
`sigma`	bandwidth parameter (default: 1).
`...`	extra parameters including
	algclust method to perform clustering on embedded data; either `"kmeans"` (default) or `"GMM"`.
	maxiter the maximum number of iterations (default: 10).

Value

a named list of S3 class `T4cluster` containing

cluster a length-n vector of class labels (from $1 : k$).

eigval eigenvalues of the graph laplacian's spectral decomposition.

embeds an $(n \times k)$ low-dimensional embedding.

algorithm name of the algorithm.

References

von Luxburg U (2007). "A tutorial on spectral clustering." *Statistics and Computing*, **17**(4), 395–416. ISSN 0960-3174, 1573-1375, doi: 10.1007/s112220079033z.

Examples

```
# ------------------------------------------------------------
#            clustering with 'iris' dataset
# ------------------------------------------------------------
## PREPARE
data(iris)
X   = as.matrix(iris[,1:4])
lab = as.integer(as.factor(iris[,5]))

## EMBEDDING WITH PCA
X2d = Rdimtools::do.pca(X, ndim=2)$Y

## CLUSTERING WITH DIFFERENT K VALUES
cl2 = scUL(X, k=2)$cluster
cl3 = scUL(X, k=3)$cluster
cl4 = scUL(X, k=4)$cluster

## VISUALIZATION
opar <- par(no.readonly=TRUE)
```

```
par(mfrow=c(1,4), pty="s")
plot(X2d, col=lab, pch=19, main="true label")
plot(X2d, col=cl2, pch=19, main="scUL: k=2")
plot(X2d, col=cl3, pch=19, main="scUL: k=3")
plot(X2d, col=cl4, pch=19, main="scUL: k=4")
par(opar)
```

Index