Package 'Spectrum'

August 19, 2019

Title Fast Adaptive Spectral Clustering for Single and Multi-View Data

Version 0.8

Author Christopher R John, David Watson

Maintainer Christopher R John <chris.r.john86@gmail.com>

Description A self-tuning spectral clustering method for single or multi-view data. 'Spectrum' uses a new type of adaptive density aware kernel that strengthens connections in the graph based on common nearest neighbours. It uses a tensor product graph data integration and diffusion procedure to integrate different data sources and reduce noise. 'Spectrum' uses either the eigengap or multimodality gap heuristics to determine the number of clusters. The method is sufficiently flexible so that a wide range of Gaussian and non-Gaussian structures can be clustered with automatic selection of K.

Depends R (>= 3.5.0)

License AGPL-3

Encoding UTF-8

LazyData true

Imports ggplot2, Rtsne, ClusterR, umap, Rfast, RColorBrewer, diptest

Suggests knitr

VignetteBuilder knitr

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2019-08-19 09:40:05 UTC

R topics documented:

`blobs` *8 blob like structures*

Description

A simulated dataset of 8 Gaussian blobs. Simulated using the 'clusterlab' CRAN package.

Usage

```
blobs
```

Format

A data frame with 10 rows and 800 variables

`brain` *A brain cancer dataset*

Description

A dataset containing The Cancer Genome Atlas expression data. From this publication https://tcga-data.nci.nih.gov/docs/publications/lgggbm_2016/. The first data frame is a 5133X150 RNA-seq data matrix, the second is a 262X150 miRNA-seq data matrix, the third is 45X150 protein array data matrix. The data was all pre-normalised then subject to log transform.

Usage

```
brain
```

Format

A list of data frames

Source

https://gdac.broadinstitute.org/

circles *Three concentric circles*

Description

Simulated data using the 'clusterSim' CRAN package.

Usage

```
circles
```

Format

A data frame with 2 rows and 540 variables

CNN_kernel *CNN_kernel: fast adaptive density aware kernel*

Description

CNN_kernel: fast adaptive density aware kernel

Usage

```
CNN_kernel(mat, NN = 3, NN2 = 7)
```

Arguments

mat	Matrix: matrix should have samples as columns and rows as features
NN	Numerical value: the number of nearest neighbours to use when calculating local sigma
NN2	Numerical value: the number of nearest neighbours to use when calculating common nearest neighbours

Value

A kernel matrix

Examples

```
CNN_kern <- CNN_kernel(blobs[,1:50])
```

kernel_pca *kernel_pca: A kernel pca function*

Description

kernel_pca: A kernel pca function

Usage

```
kernel_pca(datam, labels = FALSE, axistextsize = 18,
  legendtextsize = 18, dotsize = 3, kernel = TRUE)
```

Arguments

datam	Dataframe or matrix: a data frame with samples as columns, rows as features, or a kernel matrix
labels	Factor: to label the plot with colours
axistextsize	Numerical value: axis text size
legendtextsize	Numerical value: legend text size
dotsize	Numerical value: dot size
kernel	Logical flag: whether the input is a kernel or not

Value

A kernel PCA plot

Examples

```
ex_kernel_pca <- kernel_pca(blobs[,1:50], kernel=FALSE)
```

pca *pca: A pca function*

Description

pca: A pca function

Usage

```
pca(mydata, labels = FALSE, dotsize = 3, axistextsize = 18,
  legendtextsize = 18)
```

Arguments

mydata	Data frame or matrix: matrix or data frame with samples as columns, features as rows
labels	Factor: to label the plot with colours
dotsize	Numerical value: dot size
axistextsize	Numerical value: axis text size
legendtextsize	Numerical value: legend text size

Value

A pca plot object

Examples

```
ex_pca <- pca(blobs[,1:50])
```

---

rbfkernel_b *rbfkernel_b: fast self-tuning kernel*

Description

rbfkernel_b: fast self-tuning kernel

Usage

```
rbfkernel_b(mat, K = 3, sigma = 1)
```

Arguments

mat	Matrix: matrix should have samples as columns and rows as features
K	Numerical value: the number of nearest neighbours to use when calculating local sigma
sigma	Numerical value: a global sigma, usually left to 1 which has no effect

Value

A kernel matrix

Examples

```
stsc_kern <- rbfkernel_b(blobs[,1:50])
```

Spectrum *Spectrum: Fast Adaptive Spectral Clustering for Single and Multi-view Data*

Description

Spectrum is a self-tuning spectral clustering method for single or multi-view data. Spectrum uses a new type of adaptive density aware kernel that strengthens connections between points that share common nearest neighbours in the graph. For integrating multi-view data and reducing noise a tensor product graph data integration and diffusion procedure is used. Spectrum analyses eigenvector variance or distribution to determine the number of clusters. Spectrum is well suited for a wide range of data, including both Gaussian and non-Gaussian structures.

Usage

```
Spectrum(data, method = 1, silent = FALSE, showres = TRUE,
  diffusion = TRUE, kerneltype = c("density", "stsc"), maxk = 10,
  NN = 3, NN2 = 7, showpca = FALSE, showheatmap = FALSE,
  showdimred = FALSE, visualisation = c("umap", "tsne"), frac = 2,
  thresh = 7, fontsize = 18, dotsize = 3, tunekernel = FALSE,
  clusteralg = "GMM", FASP = FALSE, FASPk = NULL, fixk = NULL,
  krangemax = 10, runrange = FALSE, diffusion_iters = 4,
  KNNs_p = 10, missing = FALSE)
```

Arguments

data	Data frame or list of data frames: contains the data with samples as columns and rows as features. For multi-view data a list of dataframes is to be supplied with the samples in the same order.
method	Numerical value: 1 = default eigengap method (Gaussian clusters), 2 = multimodality gap method (Gaussian/ non-Gaussian clusters), 3 = no automatic method (see fixk param)
silent	Logical flag: whether to turn off messages
showres	Logical flag: whether to show the results on the screen
diffusion	Logical flag: whether to perform graph diffusion to reduce noise, recommended for method 1 only
kerneltype	Character string: 'density' (default) = adaptive density aware kernel, 'stsc' = Zelnik-Manor self-tuning kernel
maxk	Numerical value: the maximum number of expected clusters (default = 10). This is data dependent, do not set excessively high.
NN	Numerical value: kernel param, the number of nearest neighbours to use sigma parameters (default = 3)
NN2	Numerical value: kernel param, the number of nearest neighbours to use for the common nearest neigbours (default = 7)

showpca	Logical flag: whether to show pca when running on one view
showheatmap	Logical flag: whether to show heatmap of similarity matrix when running on one view
showdimred	Logical flag: whether to show UMAP or t-SNE of final similarity matrix
visualisation	Character string: what kind of dimensionality reduction to run on the similarity matrix (umap or tsne)
frac	Numerical value: optk search param, fraction to find the last substantial drop (multimodality gap method param)
thresh	Numerical value: optk search param, how many points ahead to keep searching (multimodality gap method param)
fontsize	Numerical value: controls font size of the ggplot2 plots
dotsize	Numerical value: controls the dot size of the ggplot2 plots
tunekernel	Logical flag: whether to tune the kernel, only applies for method 2
clusteralg	Character string: clustering algorithm for eigenvector matrix (GMM or km)
FASP	Logical flag: whether to use Fast Approximate Spectral Clustering (for v. high sample numbers)
FASPk	Numerical value: the number of centroids to compute when doing FASP
fixk	Numerical value: if we are just performing spectral clustering without automatic selection of K, set this parameter and method to 3
krangemax	Numerical value: the maximum K value to iterate towards when running a range of K
runrange	Logical flag: whether to run a range of K or not (default=FALSE), puts Kth results into Kth element of list
diffusion_iters	Numerical value: number of diffusion iterations for the graph (default=5)
KNNs_p	Numerical value: number of KNNs when making KNN graph (default=10)
missing	Logical flag: whether to impute missing data in multi-view analysis

Value

A list, containing: 1) cluster assignments, in the same order as input data columns 2) eigenvector analysis results (either eigenvalues or dip test statistics) 3) optimal K 4) final similarity matrix 5) eigenvectors and eigenvalues of graph Laplacian

Examples

```
res <- Spectrum(brain[[1]][,1:50])
```

spirals — *Two spirals wrapped around one another*

Description

Simulated data using the 'mlbench' CRAN package.

Usage

```
spirals
```

Format

A data frame with 2 rows and 180 variables

tsne — *tsne: A tsne function for similarity matrices or ordinary data*

Description

tsne: A tsne function for similarity matrices or ordinary data

Usage

```
tsne(mydata, labels = FALSE, perplex = 15, seed = FALSE,
  axistextsize = 18, legendtextsize = 18, dotsize = 3,
  similarity = TRUE)
```

Arguments

mydata	Data frame or matrix: kernel matrix or data frame with samples as columns, features as rows
labels	Factor: to label the plot with colours
perplex	Numerical value: this is the perplexity parameter for tsne, it usually requires adjusting for each dataset
seed	Numerical value: to repeat the results exactly, setting seed is required
axistextsize	Numerical value: axis text size
legendtextsize	Numerical value: legend text size
dotsize	Numerical value: dot size
similarity	Logical flag: whether input is similarity matrix or not

Value

A tsne plot object

Examples

```
ex_tsne <- tsne(blobs[,1:50],perplex=15,similarity=FALSE)
```

---

umap *umap: A umap function for similarity matrices or ordinary data*

Description

umap: A umap function for similarity matrices or ordinary data

Usage

```
umap(mydata, labels = FALSE, dotsize = 3, similarity = TRUE,
  axistextsize = 18, legendtextsize = 18)
```

Arguments

mydata	Data frame or matrix: kernel matrix or data frame with samples as columns, features as rows
labels	Factor: to label the plot with colours
dotsize	Numerical value: dot size
similarity	Logical flag: whether input is similarity matrix or not
axistextsize	Numerical value: axis text size
legendtextsize	Numerical value: legend text size

Value

A umap plot object

Examples

```
ex_umap <- umap(blobs[,1:50],similarity=FALSE)
```

Index