Package 'l1spectral'

July 22, 2025

Title An L1-Version of the Spectral Clustering

Version 0.99.6

Description Provides an l1-version of the spectral clustering algorithm devoted to robustly clustering highly perturbed graphs using l1-penalty. This algorithm is described with more details in the preprint C. Champion, M. Champion, M. Blazère, R. Burcelin and J.M. Loubes, ``l1-spectral clustering algorithm: a spectral clustering method using l1-regularization" (2022).

License GPL-2

Imports Rcpp (>= 0.12.5), stats, dplyr, graphics, igraph, Matrix, aricode, grDevices, caret, glmnet, ggplot2, cvTools

LinkingTo Rcpp, RcppArmadillo

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

NeedsCompilation yes

Author Camille Champion [aut], Magali Champion [aut, cre]

Maintainer Magali Champion <magali.champion@u-paris.fr>

Repository CRAN

Date/Publication 2022-01-26 17:12:46 UTC

Contents

`l1spectral-package` *Description of the package*

Description

Provides an l1-version of the spectral clustering algorithm devoted to robustly clustering highly perturbed graphs using l1-penalty. This algorithm is described with more details in the preprint C. Champion, M. Champion, M. Blazère, R. Burcelin and J.M. Loubes, "l1-spectral clustering algorithm: a spectral clustering method using l1-regularization" (2022).

Details

l1-spectral clustering is an l1-penalized version of the spectral clustering algorithm, which aims at robustly detecting cluster structure of perturbed graphs by promoting sparse eigenbases solutions of specific l1-minimization problems.

The DESCRIPTION file:

Package:	l1spectral
Title:	An L1-Version of the Spectral Clustering
Version:	0.99.6
Authors@R:	c(person("Camille", "Champion", role = "aut"), person("Magali", "Champion", role = c("aut","cre"),email="m
Description:	Provides an l1-version of the spectral clustering algorithm devoted to robustly clustering highly perturbed gra
License:	GPL-2
Imports:	Rcpp (>= 0.12.5), stats, dplyr, graphics, igraph, Matrix, aricode, grDevices, caret, glmnet, ggplot2, cvTools
LinkingTo:	Rcpp, RcppArmadillo
Encoding:	UTF-8
LazyData:	true
Roxygen:	list(markdown = TRUE)
RoxygenNote:	7.1.2
Author:	Camille Champion [aut], Magali Champion [aut, cre]
Maintainer:	Magali Champion <magali.champion@u-paris.fr>

Author(s)

NA

References

C. Champion, M. Champion, M. Blazère, R. Burcelin, J.M. Loubes, l1-spectral clustering algorithm: a robust spectral clustering using Lasso regularization, Preprint (2021).

See Also

`l1_spectralclustering`

Examples

```
#####################################################
# Performing the l1-spectral clustering on the graph
#####################################################

data(ToyData)

# if desired, the number of clusters and representative elements can be provided,
# otherwise remove
results2 <- l1_spectralclustering(A = ToyData$A_hat, pen = "lasso")
results2$comm

# when desired, the number of clusters and representative elements can also be provided
results2 <- l1_spectralclustering(A = ToyData$A_hat, pen = "lasso",
             k=2, elements = c(1,4))
```

`ComputePerformances` *Compute the performances of the l1-spectral clustering algorithm*

Description

This function computes the performances of the l1-spectral clustering algorithm in terms of Normalized Mutualized Information (NMI).

Usage

```
ComputePerformances(Results, A)
```

Arguments

`Results`	Output of the function `l1_spectralclustering()`.
`A`	The adjacency matrix of the graph to cluster.

Value

The Normalized Mutualized Information (NMI), Adjusted Mutualized Information (AMI) and Adjusted Rand Index (ARI) scores.

Author(s)

Camille Champion, Magali Champion

See Also

`l1_spectralclustering`, `l1spectral`.

Examples

```
############################################################
# Computing the performances
############################################################

data(ToyData)

results <- l1_spectralclustering(A = ToyData$A_hat, pen = "lasso",
            k=2, elements = c(1,4))

ComputePerformances(Results=results,A=ToyData$A)
```

`CreateDataSet` *Create data set*

Description

This function generates toy data that can be used to run the l1-spectal clustering algorithm: the adjacency matrix of a graph with `n` nodes and its perturbed version.

Usage

```
CreateDataSet(k, n, p, print.plot = TRUE, ClustersLength = NULL)
```

Arguments

`k`	True number of clusters.
`n`	Number of nodes.
`p`	List of probabilities of perturbations (inside and outside clusters).
`print.plot`	TRUE/FALSE indicated whether the graph should be plotted.
`ClustersLength`	Length of the `k` clusters (not necessary needed). If not provided, randomly chosen in such a way that `sum(ClustersLength)=n`.

Value

A list with the following elements:

- `A` Adjacency matrix of the generated graph.
- `A_hat` Adjacency matrix of the perturbed version of the generated graph.
- `ClustersLength` Length of the `k` clusters.

Author(s)

Camille Champion, Magali Champion

See Also

l1_spectralclustering, l1spectral.

Examples

```
############################################################
# Generating toy data
############################################################
Data <- CreateDataSet(k=3, n=20, p=list(p_inside=0.1,p_outside=0.1))

# Data is a list of three objects:
# - Data$A is an nxn matrix corresponding to the adjacency matrix of a graph
# with n nodes and k clusters,
# - Data$A_hat is a perturbed version of this graph with a probability
# p_inside of removing an edge inside clusters and
# p_outside of adding an edge between clusters,
# - Data$ClustersLength is a vector indicating the length of the clusters.

Data <- CreateDataSet(k=3, n=20, p=list(p_inside=0.1,p_outside=0.1), print.plot=TRUE)

# The same as above but the true graph and its perturbed version are both plotted.
```

FindElement *Find the representative elements of the clusters*

Description

This internal function of the l1-spectral clustering algorithm finds representative elements of the clusters, that is nodes belonging to the clusters.

Usage

```
FindElement(A, structure, clusters, elements = NULL)
```

Arguments

A	The adjacency matrix
structure	Output of the function FindStructure().
clusters	Output of the function FindNbrClusters().
elements	The representative elements of the clusters (not necessary needed). If not provided, chosen using the betweeness centrality score.

Value

A list with the following elements:

- score The edge betweenness score of all nodes,
- Nodes Vector of the representative elements.

Author(s)

Camille Champion, Magali Champion

See Also

l1_spectralclustering, l1spectral.

Examples

```
#####################################################
# Finding the representative elements of the clusters
#####################################################

# 1st: create data (not perturbed graph)
Data <- CreateDataSet(k=3, n=20, p=list(p_inside=0,p_outside=0))

# 2nd: find the structure of the graph
Structure <- FindStructure(Data$A_hat)

# 3rd: find the optimal number of clusters (here, 3 clusters)
Clusters <- FindNbrClusters(A = Data$A_hat, structure = Structure, k=3)

# 4th: find the representative elements of the clusters
Elements <- FindElement(A = Data$A_hat, structure = Structure, clusters = Clusters)
# if elements is not provided, the representative elements of each component are chosen
# by maximizing the edge betweenness score

Elements <- FindElement(A = Data$A_hat, structure = Structure,
                  clusters = Clusters, elements = c(1,5,12))
```

FindNbrClusters *Find the optimal number of clusters*

Description

This internal function of the l1-spectral algorithm finds the optimal number of clusters to build.

Usage

```
FindNbrClusters(A, structure, k = NULL, k_max = NULL)
```

Arguments

A	The adjacency matrix
structure	Output of the function FindStructure().
k	True number of clusters (not necessarily needed). If not provided, k is chosen by spectral eigengap.
k_max	Maximal number of clusters to form (not necessarily needed). If not provided, k_max is set to the number of nodes.

Value

A list with the following elements:

- `nbr_clusters` Optimal number of clusters by component,
- `nbr_clusters_total` Optimal total number of clusters.

Author(s)

Camille Champion, Magali Champion

See Also

`l1_spectralclustering`, `l1spectral`.

Examples

```
########################################
# Finding the optimal number of clusters
########################################

# 1st example: non-perturbed graph
Data <- CreateDataSet(k=3, n=20, p=list(p_inside=0,p_outside=0))

Structure <- FindStructure(Data$A_hat)

Clusters <- FindNbrClusters(A = Data$A_hat, structure = Structure, k=3)
# The number of clusters is provided (3): each of the 3 components will be divided into 1 cluster

Clusters <- FindNbrClusters(A = Data$A_hat, structure = Structure, k=5)
# The number of clusters is provided (5) and larger than the number of components (3),
# the spectral eigengap method is used to find the optimal number of clusters of each component.

# 2nd example: perturbed graph
Data <- CreateDataSet(k=3, n=20, p=list(p_inside=0.1,p_outside=0.1))

Structure <- FindStructure(Data$A_hat) # there are less than 3 components

Clusters <- FindNbrClusters(A = Data$A_hat, structure = Structure)
# The number of clusters is optimized using the spectral eigengap method
```

`FindStructure` *Find the structure of the graph from the adjacency matrix*

Description

This internal function of the spectral clustering algorithm finds the structure of the graph to cluster (number of nodes and connected components).

Usage

```
FindStructure(A)
```

Arguments

A	The adjacency matrix

Value

A list with the following elements:

- graph igraph object derived from A,
- groups List of connected components and corresponding nodes.

Author(s)

Camille Champion, Magali Champion

See Also

l1_spectralclustering, l1spectral.

Examples

```
#################################################################
# Finding the structure of the graph from the adjacency matrix
#################################################################

# 1st example: non-perturbed graph
Data <- CreateDataSet(k=3, n=20, p=list(p_inside=0,p_outside=0))

Structure <- FindStructure(Data$A_hat)
Structure$groups # the graph is not perturbed, there are 3 connected components

# 2nd example: highly-perturbed graph
Data <- CreateDataSet(k=3, n=20, p=list(p_inside=0.5,p_outside=0.5))

Structure <- FindStructure(Data$A_hat)
Structure$groups # the graph is higlhy perturbed, there are less than 3 connected components
```

---

l1_spectralclustering *Run the l1-spectral clustering algorithm*

Description

This function runs the l1-spectral algorithm, an l1-penalized version of the spectral clustering that aims at robustly clustering perturbed graphs.

Usage

```
l1_spectralclustering(
  A,
  k = NULL,
  k_max = NULL,
  elements = NULL,
  pen,
  stab = TRUE
)
```

Arguments

A	The adjacency matrix of the graph to cluster.
k	True number of clusters (not necessarily needed). If not provided, k is chosen by spectral eigengap.
k_max	Maximal number of clusters to form (not necessarily needed). If not provided, k_max is set to the number of nodes.
elements	The representative elements of the clusters (not necessary needed). If not provided, index are chosen using the betweeness centrality score.
pen	The penalty (to be chosen among "lasso" or "thresholdedLS").
stab	TRUE/FALSE indicated whether the indices should be stabilized (TRUE by default)

Value

A list with the following elements:

- comm The community matrix,
- structure The structure of the graph to cluster,
- clusters The number of clusters,
- elements The chosen representative elements of the clusters.

Author(s)

Camille Champion, Magali Champion

See Also

ComputePerformances, l1spectral.

Examples

```
#####################################################
# Performing the l1-spectral clustering on the graph
#####################################################

data(ToyData)
```

```
# if desired, the number of clusters and representative elements can be provided, otherwise, remove
results2 <- l1_spectralclustering(A = ToyData$A_hat, pen = "lasso")
results2$comm

# when desired, the number of clusters and representative elements can also be provided
results2 <- l1_spectralclustering(A = ToyData$A_hat, pen = "lasso",
            k=2, elements = c(1,4))
```

ToyData *Toy data for running the l1-spectral clustering algorithm*

Description

An example of data for running the l1-spectral clustering algorithm.

Usage

```
ToyData
```

Format

A list of three variables containing the adajcency matrix `A` of a 5-nodes graph, the adjacency matrix `A_hat` of a perturbed version of the same graph and the length of the two inherent clusters.

Value

No value returned, as this is a dataset.

Examples

```
data(ToyData)
A <- ToyData$A
A_hat <- ToyData$A_hat
clusters <- ToyData$clusters
```

Index