Package 'gsbm'

December 9, 2020

Type Package

Title Estimate Parameters in the Generalized SBM

Version 0.2.1

Maintainer Solenne Gaucher <solenne.gaucher@math.u-psud.fr>

Description
Given an adjacency matrix drawn from a Generalized Stochastic Block Model with missing observations, this package robustly estimates the probabilities of connection between nodes and detects outliers nodes, as describes in Gaucher, Klopp and Robin (2019) <arXiv:1911.13122>.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Depends R (>= 3.5.0)

Imports softImpute, RSpectra, doParallel, Matrix, foreach

Suggests knitr, rmarkdown, igraph, missSBM, RColorBrewer, combinat, testthat

VignetteBuilder knitr

NeedsCompilation no

Author Genevieve Robin [aut], Solenne Gaucher [aut, cre]

Repository CRAN

Date/Publication 2020-12-09 10:30:02 UTC

R topics documented:

`blogosphere` *Political blogs network*

Description

Network of political blogs, first analyzed in "The political blogosphere and the 2004 US Election" by Lada A. Adamic and Natalie Glance, in Proceedings of the WWW-2005 Workshop on the Weblogging Ecosystem (2005). This data set, collected before the 2004 American presidential election, records hyperlinks connecting political blogs to one another. These blogs have been labeled manually as either "liberal" or "conservative".

Usage

```
data(blogosphere)
```

Format

A list with 3 attributes:

A adjacency matrix of the graph. A binary matrix encoding 16714 connections between 1222 nodes

names vector of the names political blogs corresponding to the rows and columns of the adjacency matrix

opinion vector of the political orientation of the blogs (0 for liberal, 1 for conservative).

Source

http://www-personal.umich.edu/~mejn/netdata/

`crossval` *Parameter selection by cross validation*

Description

Selection by cross validation of the regularization parameters (lambda1 and lambda2) for estimating the probabilities of connection in a Generalized Stochastic Block Model.

Usage

```
crossval(
  A,
  epsilon = 0.1,
  nb.boot = 5,
  thresh = 1e-05,
  maxit = 100,
  lambda1.max = NULL,
  lambda2.max = NULL,
  lambda1.min = NULL,
  lambda2.min = NULL,
  length = 10,
  S0 = NULL,
  L0 = NULL,
  trace.it = FALSE
)
```

Arguments

A	nxn adjacency matrix
epsilon	regularization parameter for the L^2-norm penalty (positive number, if NULL, default method is applied)
nb.boot	number of folds for cross validation (integer)
thresh	convergence tolerance (positive number)
maxit	maximum number of iterations (positive integer)
lambda1.max	maximum regularization parameter for nuclear norm penalty (positive number)
lambda2.max	maximum regularization parameter for 2,1-norm norm penalty (positive number)
lambda1.min	minimum regularization parameter for nuclear norm penalty (positive number)
lambda2.min	minimum regularization parameter for 2,1-norm norm penalty (positive number)
length	size of cross-validation grid (integer)
S0	initial value for the sparse component
L0	initial value for the low-rank component
trace.it	whether messages about convergence should be printed (boolean)

Value

The values selected by cross-validation for the regularization parameters lambda1 and lambda2. The return value is a list of components

- lambda1 selected value for the parameter of the nuclear norm penalization.
- lambda2 selected value for the parameter of the 2,1-norm penalization.
- estim.cv result of the gsbm_mcgd function for the parameters selected.
- error a table containing the errors for all pairs of parameters on the grid.
- lambda1.grid grid of value for the parameter lambda1.
- lambda2.grid grid of value for the parameter lambda2.

Examples

```
# Draw a 50x50 adjacency matrix
# Generalized SBM with 2 communities and 2 outliers
# Create low-rank matrix L
L <- matrix(0,50,50) # low-rank component
L[1:25, 1:25] <- 0.6 # connection probabilities within community 1
L[1:25, 26:48] <- 0.1 # connection probabilities between communities 1 and 2
L[26:48, 1:25] <- 0.1 # connection probabilities between communities 1 and 2
L[26:48, 26:48] <- 0.6 # connection probabilities within community 2

# Create column-sparse matrix S
S <- matrix(0,50,50) # column sparse component
S[49:50,1:48] <- 0.6 # connection probabilities between outliers and inliers

# Draw connections and create the adjacency matrix
undir <- rbinom(n=50*(50-1)/2, size=1, prob=(L+S+t(S))[upper.tri(L+S+t(S))]) # draw edges
A <- matrix(0,50,50)
A[upper.tri(A)] <- undir
A <- (A+t(A))
```

---

gsbm_mcgd *Fit a Generalized Stochastic Block Model*

Description

Given an adjacency matrix with missing observations, the function gsbm_mgcd robustly estimates the probabilities of connections between nodes.

Usage

```
gsbm_mcgd(
  A,
  lambda1,
  lambda2,
  epsilon = 0.1,
  U = NULL,
  maxit = 100,
  thresh = 1e-06,
  S0 = NULL,
  L0 = NULL,
  R0 = NULL,
  trace.it = FALSE
)
```

Arguments

A	nxn adjacency matrix
lambda1	regularization parameter for nuclear norm penalty (positive number)

lambda2	regularization parameter for 2,1-norm penalty (positive number)
epsilon	regularization parameter for the L2-norm penalty (positive number, if NULL, default method is applied)
U	lower bound on nuclear norm (positive number, if NULL, default method is applied)
maxit	maximum number of iterations (positive integer, if NULL, default method is applied)
thresh	convergence tolerance (positive number, if NULL, default method is applied)
S0	initial value for the sparse component (if NULL, default method is applied))
L0	initial value for the low-rank component (if NULL, default method is applied)
R0	lower bound on nuclear norm of L0 (positive number, if NULL, default method is applied)
trace.it	whether messages about convergence should be printed (boolean, if NULL, default is FALSE)

Value

The estimate for the nxn matrix of probabilities of connections between nodes. It is given as the sum of a low-rank nxn matrix L, corresponding to connections between inlier nodes, and a column sparse nxn matrix S, corresponding to connections between outlier nodes and the rest of the network. The matrices L and S are such that

E(A) = L - diag(L) + S + S'

where E(A) is the expectation of the adjacency matrix, diag(L) is a nxn diagonal matrix with diagonal entries equal to those of L, and S' means S transposed.

The return value is a list of components

- A the adjacency matrix.
- L estimate for the low-rank component.
- S estimate for the column-sparse component.
- objective the value of the objective function.
- R a bound on the nuclear norm of the low-rank component.
- iter number of iterations between convergence of the objective function.

Examples

```
# Draw a 50x50 adjacency matrix
# Generalized SBM with 2 communities and 2 outliers
# Create low-rank matrix L
L <- matrix(0,50,50) # low-rank component
L[1:25, 1:25] <- 0.6 # connection probabilities within community 1
L[1:25, 26:48] <- 0.1 # connection probabilities between communities 1 and 2
L[26:48, 1:25] <- 0.1 # connection probabilities between communities 1 and 2
L[26:48, 26:48] <- 0.6 # connection probabilities within community 2

# Create column-sparse matrix S
```

```
S <- matrix(0,50,50) # column sparse component
S[49:50,1:48] <- 0.6 # connection probabilties between outliers and inliers

# Draw connections and create the adjacency matrix
undir <- rbinom(n=50*(50-1)/2, size=1, prob=(L+S+t(S))[upper.tri(L+S+t(S))]) # draw edges
A <- matrix(0,50,50)
A[upper.tri(A)] <- undir
A <- (A+t(A))

# Estimate the probabilities of connection
lambda1 <- 7
lambda2 <- 7
res <- gsbm_mcgd(A, lambda1, lambda2)
```

`gsbm_mcgd_parallel` *Fit a Generalized Stochastic Block Model*

Description

Given an adjacency matrix with missing observations, the function `gsbm_mgcd` robustly estimates the probabilities of connections between nodes.

Usage

```
gsbm_mcgd_parallel(
  A,
  lambda1,
  lambda2,
  epsilon = 0.1,
  maxit = 100,
  step_L = 0.01,
  step_S = 0.1,
  trace.it = FALSE,
  n_cores = detectCores(),
  save = FALSE,
  file = NULL
)
```

Arguments

`A`	nxn adjacency matrix
`lambda1`	regularization parameter for nuclear norm penalty (positive number)
`lambda2`	regularization parameter for 2,1-norm penalty (positive number)
`epsilon`	regularization parameter for the L2-norm penalty (positive number, if NULL, default method is applied)
`maxit`	maximum number of iterations (positive integer, if NULL, default method is applied)

step_L	step size for the gradient step of L parameter (positive number)
step_S	step size for the gradient step of S parameter (positive number)
trace.it	whether messages about convergence should be printed (boolean, if NULL, default is FALSE)
n_cores	number of cores to parallellize on (integer number, default is set with detectCores())
save	whether or not value of current estimates should be saved at each iteration (boolean)
file	if save is set to TRUE, name of the folder where current estimates should be saved (character string, file saved in file/L_iter.txt at iteration iter)

Value

The estimate for the nxn matrix of probabilities of connections between nodes. It is given as the sum of a low-rank nxn matrix L, corresponding to connections between inlier nodes, and a column sparse nxn matrix S, corresponding to connections between outlier nodes and the rest of the network. The matrices L and S are such that

E(A) = L - diag(L) + S + S'

where E(A) is the expectation of the adjacency matrix, diag(L) is a nxn diagonal matrix with diagonal entries equal to those of L, and S' means S transposed.

The return value is a list of components

- A the adjacency matrix.
- L estimate for the low-rank component.
- S estimate for the column-sparse component.
- objective the value of the objective function.
- R a bound on the nuclear norm of the low-rank component.
- iter number of iterations between convergence of the objective function.

---

les_miserables *Character network from "Les miserables" novel*

Description

A dataset containing Les Misérables characters network, encoding interactions between characters of Victor Hugo's novel. Two characters are connected whenever they appear in the same chapter. This dataset was first created by Donald Knuth as part of the Stanford Graph Base. (https://people.sc.fsu.edu/~jburkardt/datasets/sgb/sgb.html). It contains 77 nodes corresponding to characters of the novel, and 254

Usage

```
data(les_miserables)
```

Format

A list with 2 attributes:

A adjacency matrix of the graph. A binary matrix encoding 254 connections between 77 nodes

names a vector giving the names of the characters corresponding to the rows and columns of the adjacency matrix

Source

https://people.sc.fsu.edu/~jburkardt/datasets/sgb/sgb.html

`PrimarySchool` *Network of interactions within a primary school in the course of a day*

Description

This network, collected and analyzed by J. Stehle et al. in "High-resolution measurements of face-to-face contact patterns in a primary school", records physical interactions between 226 children and 10 teachers within a primary school over the course of a day. The network data was collected using a system of sensors worn by the participants. This system records the duration of interactions between two individuals facing each other at a maximum distance of one and a half meters.

Usage

```
data(PrimarySchool)
```

Format

A list with 2 attributes:

A adjacency matrix of the graph. A binary matrix encoding 2490 undirected connections between 236 nodes, with 7054 missing entries

class vector indicating the class of the node if the corresponding individual is a child, and otherwise that it belongs to the group of teachers.

Source

https://doi.org/10.1371/journal.pone.0023176.s003

Index