Package ‘greed’

April 29, 2019

Type Package

Title greed: clustering and model selection for graphs and counts data

Version 1.0

Date 2019-02-27

Author Etienne Come

Maintainer Etienne Come <etienne.come@ifsttar.fr>

Description Greed enable the clustering of networks and counts data matrix such as document/term matrix with different type of generative models. Model selection and clustering is performed in combination by optimizing the Integrated Classification Likelihood (which is equivalent to minimizing the description length). Their are four models availables : SBM degree corected SBM, Mixture of Multinomials, Multivariate mixture of poisson. The optimization is performed thanks to a combination of greedy local search and a genetic algorithm.

License GPL

Depends R (>= 2.10)

Imports Rcpp (>= 1.0.0), Matrix, future, listenv, ggplot2, graphics, methods, stats,rARPACK

LinkingTo Rcpp, RcppArmadillo

Suggests testthat, knitr, rmarkdown, igraph

VignetteBuilder knitr

RoxygenNote 6.1.1

Collate 'RcppExports.R' 'models_classes.R' 'fit_classes.R' 'cleanpath.R' 'genetic_alg.R' 'hybrid_alg.R' 'alg_classes.R' 'dcsbm.R' 'generator.R' 'greed.R' 'misc.R' 'mm.R' 'mreg.R' 'multistart_alg.R' 'plot.R' 'sbm.R'

R topics documented:

`algs-classes` *Optimization algorithm classes*

Description

Optimization algorithm classes

alg An S4 class to represent an abstract optimisation algorithm.

gree An S4 class to represent a greedy algorithm extends `alg` class with multiple start.

seed An S4 class to represent a greedy algorithm extends `alg` class with initialization with spectral clustering and or k-means.

genetic An S4 class to represent a hybrid genetic/greedy algorithm extends `alg` class.

genetic An S4 class to represent a hybrid genetic/greedy algorithm extends `alg` class.

Slots

`name` Name of the algorithm

`nb_start` number of random starts (default to 10)

`pop_size` size of the solutions populations (default to 10)

`nb_max_gen` maximal number of generation to produce (default to 4)

`pop_size` size of the solutions populations (default to 10)

`nb_max_gen` maximal number of generation to produce (default to 4)

`cut,icl_path-method` *Cut cut a path to a desired number of cluster*

Description

Cut cut a path to a desired number of cluster

Usage

```
## S4 method for signature 'icl_path'
cut(x, K)
```

Arguments

`x`	A an `icl_path` solution
`K`	Desired number of cluster

Value

an icl_path obejct with the desired number of cluster

Methods (by class)

- `icl_path`: method to cut a fit to a desired number of cluster

`fits-classes` *Clustering solutions classes*

Description

Clustering solutions classes

icl_fit

An S4 abstract class to represent an icl fit of a clustering model.

- slots : `name,K,icl,cl,count`

icl_path

An S4 class to represent a hierachical path of solution.

- slots : `path,tree,ggtree,logalpha`

dcsbm_fit

An S4 class to represent a fit of a stochastick block model that extend `icl_fit`.

- slots : `name,K,icl,cl,obs_stats,model` dcsbm_path

An S4 class to represent a hierachical path of solutions for a DC-SBM model that extend `dcsbm_fit-class` and `icl_path-class`.

- slots : `name,K,icl,cl,obs_stats, model, path, tree, ggtree, logalpha`

mm_fit

An S4 class to represent an icl fit of a mixture of multinomials model that extend `icl_fit`.

- slots : `name,K,icl,cl,obs_stats,model`

mm_path

An S4 class to represent a hierachical path of solutions for a mixture of mutinomials model that extend `mm_fit-class` and `icl_path-class`.

- slots : `name,K,icl,cl,obs_stats, model, path, tree, ggtree, logalpha`

mreg_fit

An S4 class to represent an icl fit of a mixture of multinomials model that extend `icl_fit`.

- slots : `name,K,icl,cl,obs_stats,model`

mreg_path

An S4 class to represent a hierachical path of solutions for a Mixture of Regression model that extend `mreg_fit-class` and `icl_path-class`.

- slots : `name,K,icl,cl,obs_stats, model, path, tree, ggtree, logalpha`

sbm_fit

An S4 class to represent a fit of a stochastick block model that extend `icl_fit`.

- slots : `name,K,icl,cl,obs_stats,model`

sbm_path

An S4 class to represent a hierachical path of solutions for a SBM model that extend `sbm_fit-class` and `icl_path-class`.

- slots : `name,K,icl,cl,obs_stats, model, path, tree, ggtree, logalpha`

Slots

`name` of the fit

`K` a numeric vector of length 1 which correspond to the number of clusters

`icl` a numeric vector of length 1 which store the the icl value

`cl` a numeric vector of length N which store the clusters labels

`obs_stats` a list to store the observed statistics of the model needed to compute ICL.

`train_hist` a data.frame to store training history (format depends on the used algorithm used).

`path` a list of merge moves describing the hierachie of merge followed to complete totaly the merge path.

`tree` a tree representation of the merges.

`ggtree` a data.frame for easy ploting of the dendogram

`logalpha` a numeric value which corresponds to the starting value of log(alpha).

`model` an `icl_model` to store the model fitted

`model` an `icl_model` to store the model fitted

`fit_greed` *fit_greed_init*

Description

fit_greed_init

Usage

```
fit_greed(model, data, clt, type = "both", nb_max_pass = 50L,
  verbose = FALSE)
```

Arguments

`model`	icl_model
`data`	list with clustering data (fileds depend on model type)
`clt`	cluster labels 0,...,K-1
`type`	merge, swap, none, or both (default)
`nb_max_pass`	maximum number of pass for greedy swap
`verbose`	boolean for verbose mode default to false

Value

a model_fit object

`fit_greed_path` *fit_greed_path*

Description

fit_greed_path

Usage

```
fit_greed_path(data, init_fit)
```

Arguments

`data`	list with clustering data depnds on model type
`init_fit`	initial fit object

Value

a model_path object

`graphbalance` *graph_balance*

Description

graph_balance

Usage

```
graphbalance(x)
```

Arguments

`x`	a `sbm_fit-class` object to be plot

Value

a ggplot2 graph

`greed` *greed*

Description

Greed enable the clustering of networks and counts data such as document/term matrix with different model. Model selection and clustering is performed in combination by optimizing the Integrated Classification Likelihood (which is equivalent to minimizing the description length). Their are four models availables :

- Stochastic Block Model (directed)
- Degree corected Stochastic Block Model (directed)
- Mixture of Multinomials
- Multivariate mixture of poissons

The optimization is performed thanks to a combination of greedy local search and a genetic algorithm. The main entry point is the `greed` function to perfom the clustering.

Usage

```
greed(X, K = 20, model = find_model(X), alg = methods::new("hybrid"),
  verbose = FALSE)
```

Arguments

`X`	data to cluster sparseMatrix or matrix
`K`	Desired number of cluster
`model`	a dcsbm, sbm or mm model
`alg`	an optimisation algorithm hybrid, mutlistarts, seed or genetic
`verbose`	boolean for verbose mode

Value

an icl_path object

`greed_cond` *greed_cond*

Description

greed_cond

Usage

```
greed_cond(X, y, K = 20, model = find_model_cond(X, y),
  alg = methods::new("hybrid"), verbose = FALSE)
```

Arguments

`X`	covariable data
`y`	target variable
`K`	Desired number of cluster
`model`	an mreg model
`alg`	an optimisation algorithm hybrid, mutlistarts, seed or genetic
`verbose`	boolean for verbose mode

Value

an icl_path object

`lm_post` *lm_post*

Description

lm_post

Usage

```
lm_post(X, y, regu, a0, b0)
```

Arguments

`X`	data matrix of covariates Nxd
`y`	target Nx1
`regu`	prior precision parameter
`a0`	prior parameter
`b0`	prior parameter

`lm_post_add` *lm_post_add*

Description

lm_post_add

Usage

```
lm_post_add(current, X, y, regu, a0, b0)
```

Arguments

`current`	gaussian linear model to update
`X`	data matrix of covariates Ntxd
`y`	target Ntx1
`regu`	prior precision parameter
`a0`	prior parameter
`b0`	prior parameter

lm_post_del *lm_post_del*

Description

lm_post_del

Usage

```
lm_post_del(current, X, y, regu, a0, b0)
```

Arguments

current	gaussian linear model to update
X	data matrix of covariates Ntxd
y	target Ntx1
regu	prior precision parameter
a0	prior parameter
b0	prior parameter

lm_post_del1 *lm_post_del1*

Description

lm_post_del1

Usage

```
lm_post_del1(current, X, y, regu, a0, b0)
```

Arguments

current	gaussian linear model to update
X	data matrix of covariates 1xd
y	target 1x1
regu	prior precision parameter
a0	prior parameter
b0	prior parameter

`lm_post_merge` *lm_post_merge*

Description

lm_post_merge

Usage

```
lm_post_merge(current_k, current_l, regu, a0, b0)
```

Arguments

`current_k`	gaussian linear model to merge
`current_l`	gaussian linear model to merge
`regu`	prior precision parameter
`a0`	prior parameter
`b0`	prior parameter

`models-classes` *Clustering models classes*

Description

Clustering models classes

icl_model

An S4 class to represent an abstract clustering model

- slots : `name,alpha`

dcsbm

An S4 class to represent a stochastick block model that extends `icl_model` class.

- slots : `name,alpha,a0,b0`

mm

An S4 class to represent a mixture of multinomial also known has mixture of unigrams that extends `icl_model` class.

- slots : `name,alpha,beta`

mreg

An S4 class to represent a mixture of multinomial also known has mixture of unigrams that extends `icl_model` class.

- slots : `name,alpha,reg,a0,b0`

sbm

An S4 class to represent a stochastick block model that extends `icl_model` class.

- slots : `name,alpha,a0,b0`

Slots

`name` a character vector

`alpha` a numeric vector of length 1 which define the parameters of the dirichlet over the cluster proportions (default to 1)

`a0` a numeric vector of length 1 which define the parameters of the beta prior over the edges (default to 1)

`b0` a numeric vector of length 1 which define the parameters of the beta prior over the non-edges (default to 1)

`beta` a numeric vector of length 1 which define the parameters of the beta prior over the counts (default to 1)

`reg` a numeric vector of length 1 which define the variance parameter of the normal prior over the regression parameters (default to 0.1)

`a0` a numeric vector of length 1 which define the parameter a0 of the inverse gamma over the regression noise variance parameters (default to 1)

`b0` a numeric vector of length 1 which define the parameter b0 of the inverse gamma prior over the regression noise variance parameters (default to 1)

`a0` a numeric vector of length 1 which define the parameters of the beta prior over the edges (default to 1)

`b0` a numeric vector of length 1 which define the parameters of the beta prior over the non-edges (default to 1)

Examples

```
new("dcsbm")
new("mm")
new("mm",alpha=1,beta=1)
new("mreg")
new("mreg",alpha=1,reg=5,a0=0.5,b0=0.5)
new("sbm")
new("sbm",a0=0.5,b0=0.5,alpha=1)
```

nodelinklab *nodelinklab*

Description

nodelinklab

Usage

```
nodelinklab(sol, labels, s = 0)
```

Arguments

sol	mm_path-class object to be plot
labels	a vector of cluster labels
s	threeshold for links

Value

a ggplot2 graph

plot,dcsbm_fit,missing-method *Plot a clustering results*

Description

Main methods to explore clusterings results visualy.

Usage

```
## S4 method for signature 'dcsbm_fit,missing'
plot(x, type = "blocks")

## S4 method for signature 'dcsbm_path,missing'
plot(x, type = "blocks")

## S4 method for signature 'mm_fit,missing'
plot(x, type = "blocks")

## S4 method for signature 'mm_path,missing'
plot(x, type = "blocks")

## S4 method for signature 'mreg_fit,missing'
plot(x, type = "blocks")
```

```
## S4 method for signature 'mreg_path,missing'
plot(x, type = "blocks")

## S4 method for signature 'sbm_fit,missing'
plot(x, type = "blocks")

## S4 method for signature 'sbm_path,missing'
plot(x, type = "blocks")
```

Arguments

`x`	`icl_fit-class` object to be ploted
`type`	type of desired graphics : tree,pathy, blocks, nodelink, front

Value

a ggplot2 object to visualize the results

`post_probs` *post_probs*

Description

post_probs

Usage

```
post_probs(model, data, clt)
```

Arguments

`model`	icl_model
`data`	list with clustering data (fileds depend on model type)
`clt`	cluster labels in 1,..,K

print,icl_path-method *print print an icl_path object*

Description

print print an icl_path object

Usage

```
## S4 method for signature 'icl_path'
print(x)
```

Arguments

x	icl_path-class object to print

rdcsbm *Generate graph adjacency matrix using a degree corrected SBM*

Description

rmm returns a count matrix and the cluster labels generated randomly unsig a Mixture of Multinomial model.

Usage

```
rdcsbm(N, pi, mu, betain, betaout)
```

Arguments

N	A numeric value the size of the graph to generate
pi	A numeric vector of length K with clusters proportions. Must sum up to 1.
mu	A numeric matrix of dim K x K with the connectivity pattern to generate, elements in [0,1].
betain	A numeric vector of length N which specify the in-degree correction will be normalized per cluster during the generation.
betaout	A numeric vector of length N which specify the out-degree correction will be normalized per cluster during the generation.

Details

It take the sample size, cluster proportions and emission matrix, and as input and sample a graph accordingly together with the clusters labels.

Value

A list with fields:

- x: the count matrix as a `dgCMatrix`
- K: number of generated clusters
- N: number of vertex
- cl: vector of clusters labels
- pi: clusters proportions
- mu: connectivity matrix
- betain: normalized in-degree parameters
- betaout: normalized out-degree parameters

`rmm` *Generate graph adjacency matrix using a Multinomial Mixture*

Description

`rmm` returns a count matrix and the cluster labels generated randomly unsig a Mixture of Multinomial model.

Usage

```
rmm(N, pi, mu, lambda)
```

Arguments

`N`	A numeric value the size of the graph to generate
`pi`	A numeric vector of length K with clusters proportions. Must sum up to 1.
`mu`	A numeric matrix of dim k x D with the clusters patterns to generate, all elements in [0,1].
`lambda`	A numeric value which specify the expectation for the row sums.

Details

It take the sample size, cluster proportions and emission matrix, and as input and sample a graph accordingly together with the clusters labels.

Value

A list with fields:

- x: the count matrix as a `dgCMatrix`
- K: number of generated clusters
- N: number of vertex
- cl: vector of clusters labels
- pi: clusters proportions
- mu: connectivity matrix
- lambda: expectation of row sums

`rmreg` *Generate X and y with a mixture of regression model*

Description

`rmreg` returns an X matrix, a y vector and the cluster labels generated randomly unsig a Mixture of regression model.

Usage

```
rmreg(N, pi, mu, sigma, X = cbind(matrix(stats::rnorm(N * (nrow(mu) -
  1)), N, nrow(mu) - 1), rep(1, N)))
```

Arguments

`N`	A numeric value the size of the graph to generate
`pi`	A numeric vector of length K with clusters proportions (must sum up to 1)
`mu`	A numeric matrix of dim K x d with the regression parameters
`sigma`	A numeric of length 1 with the target conditional variance
`X`	A matrix of covariate

Details

It take the sample size, cluster proportions and regression parameters matrix and variance as input accordingly

Value

A list with fields:

- X: the covariate matrix
- y: the target feature
- K: number of generated clusters
- N: sample size
- cl: vector of clusters labels
- pi: clusters proportions
- mu: regression parameters
- sigma: conditional variance

`rsbm` *Generate graph adjacency matrix using a SBM*

Description

`rsbm` returns the adjacency matrix and the cluster labels generated randomly unsing a Stochastick Block Model.

Usage

```
rsbm(N, pi, mu)
```

Arguments

`N`	A numeric value the size of the graph to generate
`pi`	A numeric vector of length K with clusters proportions. Must sum up to 1.
`mu`	A numeric matrix of dim K x K with the connectivity pattern to generate. elements in [0,1].

Details

This function take graph size, cluster proportions and connectivity matrix as input and sample a graph accordingly together with the clusters labels.

Value

A list with fields:

- x: the graph adjacency matrix as a `dgCMatrix`
- K: number of generated clusters
- N: number of vertex
- cl: vector of clusters labels
- pi: clusters proportions
- mu: connectivuty matrix

Examples

```
simu = rsbm(100,rep(1/5,5),diag(rep(0.1,5))+0.001)
x  = simu$x
xl = simu$cl
```

show,icl_path-method *show show an icl_path object*

Description

show show an icl_path object

Usage

```
## S4 method for signature 'icl_path'
show(object)
```

Arguments

`object`	`icl_path-class` object to print

spectral *spectral Regularized spectral clustering nips paper 2013*

Description

spectral Regularized spectral clustering nips paper 2013

Usage

```
spectral(X, K)
```

Arguments

`X`	An adjacency matrix in sparse format
`K`	Desired number of cluster

Value

cl Vector of clsuter labels