Package 'bootcluster'

November 12, 2024

Type Package

Title Bootstrapping Estimates of Clustering Stability

Version 0.4.1

Author Han Yu [aut],
Mingmei Tian [aut],
Tianmou Liu [aut, cre]

Maintainer Tianmou Liu <tianmouliu@outlook.com>

Description Implementation of the bootstrapping approach for the estimation of clustering stability and its application in estimating the number of clusters, as introduced by Yu et al (2016)<doi:10.1142/9789814749411_0007>. Implementation of the non-parametric bootstrap approach to assessing the stability of module detection in a graph, the extension for the selection of a parameter set that defines a graph from data in a way that optimizes stability and the corresponding visualization functions, as introduced by Tian et al (2021) <doi:10.1002/sam.11495>. Implemented out-of-bag stability estimation function and k-select Smin-based k-selection function as introduced by Liu et al (2022) <doi:10.1002/sam.11593>. Implemented ensemble clustering method based-on k-means clustering method, spectral clustering method and hierarchical clustering method.

Depends R (>= 3.5.1)

Imports cluster, mclust, flexclust, fpc, plyr, dplyr, doParallel, foreach, igraph (>= 1.2.0), compiler, stats, parallel, grid, ggplot2, gridExtra, intergraph, GGally, network, kernlab, sna

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2024-11-12 09:50:02 UTC

Contents

`agreement` *Calculate agreement between two clustering results*

Description

Calculate agreement between two clustering results

Usage

```
agreement(clst1, clst2)
```

Arguments

`clst1`	First clustering result
`clst2`	Second clustering result

Value

Vector of agreement values

`agreement_nk` *Calculate agreement between two clustering results with known number of clusters*

Description

Calculate agreement between two clustering results with known number of clusters

Usage

```
agreement_nk(clst1, clst2, nk)
```

Arguments

`clst1`	First clustering result
`clst2`	Second clustering result
`nk`	Number of clusters

Value

Vector of agreement values

`ensemble.cluster.multi` *Multi-Method Ensemble Clustering with Graph-based Consensus*

Description

Implements ensemble clustering by combining multiple clustering methods (k-means, hierarchical, and spectral clustering) using a graph-based consensus approach.

Usage

```
ensemble.cluster.multi(
  x,
  k_km,
  k_hc,
  k_sc,
  n_ref = 3,
  B = 100,
  hc.method = "ward.D",
  dist_method = "euclidean"
)
```

Arguments

x	data.frame or matrix where rows are observations and columns are features
k_km	number of clusters for k-means clustering
k_hc	number of clusters for hierarchical clustering
k_sc	number of clusters for spectral clustering
n_ref	number of reference distributions for stability assessment (default: 3)
B	number of bootstrap samples for stability estimation (default: 100)
hc.method	hierarchical clustering method (default: "ward.D")
dist_method	distance method for spectral clustering (default: "euclidean")

Details

This function implements a multi-method ensemble clustering approach that: 1. Applies multiple clustering methods (k-means, hierarchical, spectral) 2. Assesses stability of each clustering through bootstrapping 3. Constructs a weighted bipartite graph representing all clusterings 4. Uses fast greedy community detection for final consensus

Value

A list containing:

membership Final cluster assignments from ensemble consensus

k_consensus Number of clusters found in consensus

individual_results List of results from individual clustering methods

stability_measures Stability measures for each method

graph igraph object of the ensemble graph

Examples

```
data(iris)
df <- iris[,1:4]
result <- ensemble.cluster.multi(df, k_km=3, k_hc=3, k_sc=3)
plot(df[,1:2], col=result$membership, pch=16)
```

`esmbl.stability` *Estimate the stability of a clustering based on non-parametric bootstrap out-of-bag scheme, with option for subsampling scheme*

Description

Estimate the stability of a clustering based on non-parametric bootstrap out-of-bag scheme, with option for subsampling scheme

Usage

```
esmbl.stability(
  x,
  k,
  scheme = "kmeans",
  B = 100,
  hc.method = "ward.D",
  cut_ratio = 0.5,
  dist_method = "euclidean"
)
```

Arguments

`x`	`data.frame` of the data set where rows are observations and columns are features
`k`	number of clusters for which to estimate the stability
`scheme`	clustering method to use ("kmeans", "hc", or "spectral")
`B`	number of bootstrap re-samples
`hc.method`	hierarchical clustering method (default: "ward.D")
`cut_ratio`	ratio for subsampling (default: 0.5)
`dist_method`	distance method for spectral clustering (default: "euclidean")

Details

This function estimates the stability through out-of-bag observations It estimate the stability at the (1) observation level, (2) cluster level, and (3) overall.

Value

membership vector of membership for each observation from the reference clustering

obs_wise vector of estimated observation-wise stability

clust_wise vector of estimated cluster-wise stability

overall numeric estimated overall stability

Smin numeric estimated Smin through out-of-bag scheme

Author(s)

Tianmou Liu

Examples

```
set.seed(123)
data(iris)
df <- iris[,1:4]
result <- esmbl.stability(df, k=3, scheme="kmeans")
```

`k.select` *Estimate number of clusters*

Description

Estimate number of clusters by bootstrapping stability

Usage

```
k.select(x, range = 2:7, B = 20, r = 5, threshold = 0.8, scheme_2 = TRUE)
```

Arguments

`x`	a `data.frame` of the data set
`range`	a `vector` of `integer` values, of the possible numbers of clusters k
`B`	number of bootstrap re-samplings
`r`	number of runs of k-means
`threshold`	the threshold for determining k
`scheme_2`	`logical` `TRUE` if scheme 2 is used, `FASLE` if scheme 1 is used

Details

This function estimates the number of clusters through a bootstrapping approach, and a measure Smin, which is based on an observation-wise similarity among clusterings. The number of clusters k is selected as the largest number of clusters, for which the Smin is greater than a threshold. The threshold is often selected between 0.8 ~ 0.9. Two schemes are provided. Scheme 1 uses the clustering of the original data as the reference for stability calculations. Scheme 2 searches acrossthe clustering samples that gives the most stable clustering.

Value

`profile` a `vector` of Smin measures for determining k

`k` `integer` estimated number of clusters

Author(s)

Han Yu

References

Bootstrapping estimates of stability for clusters, observations and model selection. Han Yu, Brian Chapman, Arianna DiFlorio, Ellen Eischen, David Gotz, Matthews Jacob and Rachael Hageman Blair.

Examples

```
set.seed(1)
data(wine)
x0 <- wine[,2:14]
x <- scale(x0)
k.select(x, range = 2:10, B=20, r=5, scheme_2 = TRUE)
```

---

k.select_ref *Estimate number of clusters*

Description

Estimate number of clusters by bootstrapping stability

Usage

```
k.select_ref(df, k_range = 2:7, n_ref = 5, B = 100, B_ref = 50, r = 5)
```

Arguments

df	data.frame of the input dataset
k_range	integer valued vector of the numbers of clusters k to be tested upon
n_ref	number of reference distribution to be generated
B	number of bootstrap re-samples
B_ref	number of bootstrap resamples for the reference distributions
r	number of runs of k-means

Details

This function uses the out-of-bag scheme to estimate the number of clusters in a dataset. The function calculate the Smin of the dataset and at the same time, generate a reference dataset with the same range as the original dataset in each dimension and calculate the Smin_ref. The differences between Smin and Smin_ref at each k,Smin_diff(k), is taken into consideration as well as the standard deviation of the differences. We choose the k to be the argmax of (Smin_diff(k) - (Smin_diff(k+1) + (Smin_diff(k+1)))). If Smin_diff(k) less than 0.1 for all k in k_range, we say k = 1

Value

`profile vector` of (Smin_diff(k) - (Smin_diff(k+1) + se(Smin_diff(k+1)))) measures for researchers's inspection

`k` estimated number of clusters

Author(s)

Tianmou Liu

References

Bootstrapping estimates of stability for clusters, observations and model selection. Han Yu, Brian Chapman, Arianna DiFlorio, Ellen Eischen, David Gotz, Matthews Jacob and Rachael Hageman Blair.

Examples

```
set.seed(1)
data(iris)
df <- data.frame(iris[,1:4])
df <- scale(df)
k.select_ref(df, k_range = 2:7, n_ref = 5, B=500, B_ref = 500, r=5)
```

`min_agreement` *Calculate minimum agreement across clusters*

Description

Calculates the minimum average agreement value across all clusters

Usage

```
min_agreement(clst, agrmt)
```

Arguments

`clst`	clustering result vector
`agrmt`	agreement values vector

Value

minimum average agreement value across clusters

network.stability *Estimate of detect module stability*

Description

Estimate of detect module stability

Usage

```
network.stability(
  data.input,
  threshold,
  B = 20,
  cor.method,
  large.size,
  PermuNo,
  scheme_2 = FALSE
)
```

Arguments

data.input	a data.frame of the data set where the rows are observations and columns are covariates
threshold	a numeric number of threshold for correlation matrix
B	number of bootstrap re-samplings
cor.method	the correlation method applied to the data set,three method are available: "pearson", "kendall", "spearman".
large.size	the smallest set of modules, the large.size=0 is recommended to use right now.
PermuNo	number of random graphs for null
scheme_2	logical TRUE if scheme 2 is used, FASLE if scheme 1 is used. Right now, only FASLE is recommended.

Details

This function estimates the modules' stability through bootstrapping approach for the given threshold. The approach to stability estimation is to compare the module composition of the reference correlation graph to the various bootstrapped correlation graphs, and to assess the stability at the (1) node-level, (2) module-level, and (3) overall.

Value

stabilityresult a list of result for nodes-wise stability

modularityresult list of modularity information with the given threshold

`jaccardresult list` estimated unconditional observed stability and the estimates of expected stability under the null

`originalinformation list` information for original data, igraph object and adjacency matrix constructed with the given threshold

Author(s)

Mingmei Tian

References

A framework for stability-based module detection in correlation graphs. Mingmei Tian,Rachael Hageman Blair,Lina Mu, Matthew Bonner, Richard Browne and Han Yu.

Examples

```
set.seed(1)
data(wine)
x0 <- wine[1:50,]

mytest<-network.stability(data.input=x0,threshold=0.7, B=20,
cor.method='pearson',large.size=0,
PermuNo = 10,
scheme_2 = FALSE)
```

---

network.stability.output

Plot method for objests from threshold.select

Description

Plot method for objests from threshold.select

Usage

```
network.stability.output(input, optimal.only = FALSE)
```

Arguments

`input`	a `list` of results from function `threshold.select`
`optimal.only`	a `logical` value indicating whether only plot the network with optimal threshold or not. The default is False, generating all network figures with a large number of nodes could take some time.

Details

network.stability.output is used to generate a series of network plots based on the given threshold.seq,where the nodes are colored by the level of stability. The network with optimal threshold value selected by function threshold.select is colored as red.

Value

Plot of network figures

Author(s)

Mingmei Tian

References

A framework for stability-based module detection in correlation graphs. Mingmei Tian,Rachael Hageman Blair,Lina Mu, Matthew Bonner, Richard Browne and Han Yu.

Examples

```
set.seed(1)
data(wine)
x0 <- wine[1:50,]

mytest<-threshold.select(data.input=x0,threshold.seq=seq(0.1,0.5,by=0.05), B=20,
cor.method='pearson',large.size=0,
PermuNo = 10,
no_cores=1,
scheme_2 = FALSE)
network.stability.output(mytest)
```

---

ob.stability *Estimate the stability of a clustering based on non-parametric bootstrap out-of-bag scheme, with option for subsampling scheme*

Description

Estimate the stability of a clustering based on non-parametric bootstrap out-of-bag scheme, with option for subsampling scheme

Usage

```
ob.stability(x, k, B = 500, r = 5, subsample = FALSE, cut_ratio = 0.5)
```

Arguments

x	data.frame of the data set where the rows as observations and columns as dimensions of features
k	number of clusters for which to estimate the stability
B	number of bootstrap re-samples
r	integer parameter in the kmeansCBI() funtion
subsample	logical parameter to use the subsampling scheme option in the resampling process (instead of bootstrap)
cut_ratio	numeric parameter between 0 and 1 for subsampling scheme training set ratio

Details

This function estimates the stability through out-of-bag observations It estimate the stability at the (1) observation level, (2) cluster level, and (3) overall.

Value

membership vector of membership for each observation from the reference clustering

obs_wise vector of estimated observation-wise stability

clust_wise vector of estimated cluster-wise stability

overall numeric estimated overall stability

Smin numeric estimated Smin through out-of-bag scheme

Author(s)

Tianmou Liu

References

Bootstrapping estimates of stability for clusters, observations and model selection. Han Yu, Brian Chapman, Arianna DiFlorio, Ellen Eischen, David Gotz, Matthews Jacob and Rachael Hageman Blair.

Examples

```
set.seed(123)
data(iris)
df <- data.frame(iris[,1:4])
# You can choose to scale df before clustering by
# df <- scale(df)
ob.stability(df, k = 2, B=500, r=5)
```

ref_dist *Generate reference distribution for stability assessment*

Description

Generates a reference distribution by sampling from uniform distributions with ranges determined by the original data.

Usage

```
ref_dist(df)
```

Arguments

df	data.frame or matrix of the original dataset

Details

Generate Reference Distribution

Value

A scaled matrix containing the reference distribution

Examples

```
data(iris)
df <- iris[,1:4]
ref <- ref_dist(df)
```

ref_dist_bin *Generate reference distribution for binary data*

Description

Generates a reference distribution by randomly permuting each column of the original binary dataset.

Usage

```
ref_dist_bin(df)
```

Arguments

df	data.frame or matrix of the original binary dataset

Details

Generate Binary Reference Distribution

Value

A matrix containing the permuted binary reference distribution

Examples

```
binary_data <- matrix(sample(0:1, 100, replace=TRUE), ncol=5)
ref <- ref_dist_bin(binary_data)
```

---

ref_dist_pca *Generate PCA-based reference distribution*

Description

Generates a reference distribution in PCA space by sampling from uniform distributions with ranges determined by the PCA-transformed data.

Usage

```
ref_dist_pca(df)
```

Arguments

df	data.frame or matrix of the original dataset

Details

Generate Reference Distribution using PCA

Value

A scaled matrix containing the reference distribution in PCA space

Examples

```
data(iris)
df <- iris[,1:4]
ref <- ref_dist_pca(df)
```

`stability` *Estimate clustering stability of k-means*

Description

Estimate of k-means bootstrapping stability

Usage

```
stability(x, k, B = 20, r = 5, scheme_2 = TRUE)
```

Arguments

`x`	a `data.frame` of the data set
`k`	a `integer` number of clusters
`B`	number of bootstrap re-samplings
`r`	number of runs of k-means
`scheme_2`	`logical` `TRUE` if scheme 2 is used, `FASLE` if scheme 1 is used

Details

This function estimates the clustering stability through bootstrapping approach. Two schemes are provided. Scheme 1 uses the clustering of the original data as the reference for stability calculations. Scheme 2 searches acrossthe clustering samples that gives the most stable clustering.

Value

`membership` a `vector` of membership for each observation from the reference clustering

`obs_wise` `vector` of estimated observation-wise stability

`overall` `numeric` estimated overall stability

Author(s)

Han Yu

References

Bootstrapping estimates of stability for clusters, observations and model selection. Han Yu, Brian Chapman, Arianna DiFlorio, Ellen Eischen, David Gotz, Matthews Jacob and Rachael Hageman Blair.

Examples

```
set.seed(1)
data(wine)
x0 <- wine[,2:14]
x <- scale(x0)
stability(x, k = 3, B=20, r=5, scheme_2 = TRUE)
```

`threshold.select` *Estimate of the overall Jaccard stability*

Description

Estimate of the overall Jaccard stability

Arguments

`data.input`	a `data.frame` of the data set where the rows are observations and columns are covariates
`threshold.seq`	a `numeric` sequence of candidate threshold
`B`	number of bootstrap re-samplings
`cor.method`	the correlation method applied to the data set,three method are available: `"pearson"`, `"kendall"`, `"spearman"`.
`large.size`	the smallest set of modules, the `large.size=0` is recommended to use right now.
`PermuNo`	number of random graphs for the estimation of expected stability
`no_cores`	a `interger` number of CPU cores on the current host (This function can't not be used yet).

Details

`threshold.select` is used to estimate of the overall Jaccard stability from a sequence of given threshold candidates, `threshold.seq`.

Value

`stabilityresult` a `list` of result for nodes-wise stability

`modularityresult` a `list` of modularity information with each candidate threshold

`jaccardresult` a `list` estimated unconditional observed stability and the estimates of expected stability under the nul

`originalinformation` a `list` information for original data, igraph object and adjacency matrix constructed with each candidate threshold

`threshold.seq` a `list` of candicate threshold given to the function

Author(s)

Mingmei Tian

References

A framework for stability-based module detection in correlation graphs. Mingmei Tian,Rachael Hageman Blair,Lina Mu, Matthew Bonner, Richard Browne and Han Yu.

Examples

```
set.seed(1)
data(wine)
x0 <- wine[1:50,]

mytest<-threshold.select(data.input=x0,threshold.seq=seq(0.5,0.8,by=0.05), B=20,
cor.method='pearson',large.size=0,
PermuNo = 10,
no_cores=1,
scheme_2 = FALSE)
```

---

wine *Wine Data Set*

Description

These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines.

Usage

```
data(wine)
```

Format

The data set `wine` contains a `data.frame` of 14 variables. The first variable is the types of wines. The other 13 variables are quantities of the constituents.

References

https://archive.ics.uci.edu/ml/datasets/wine

Index