Package 'CLARA.seq'

June 24, 2021

Type Package

Title CLARA Clustering

Version 1.1.1

Author Louis Tochon (with Matthias Studer)

Maintainer Louis Tochon <louis.tochon@etu.unige.ch>

Description
Clustering algorithm (CLARA) to handle big datasets and index to evaluate the quality of the partition. This package works with the help of TraMineR's package for distance calulations.

Imports TraMineR, cluster, dplyr, doParallel, parallel, foreach

License GPL

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

R topics documented:

`clarans_clust` *CLARANS clustering*

Description

With the help of TraMineR package, CLARANS clustering provide a clustering of big dataset. The main objective is to cluster state sequences with the "LCS" distance calculation method to find the best partition in N clusters.

WARNING : this function is less efficient than cLARA.

Usage

```
clarans_clust(
  data,
  nb_cluster,
  distargs = list(method = "LCS"),
  maxneighbours,
  numlocal,
  plot = FALSE,
  cores = detectCores() - 1
)
```

Arguments

data	The dataset to use. In case of sequences, use seqdef (from TraMineR package) to create such an object.
nb_cluster	The number of medoids
distargs	List with method parameters to apply. (See the function seqdist in TraMineR package)
maxneighbours	Number of neighbours to explore to find a better clustering
numlocal	Number of initialisation of the starting medoids
plot	Boolean variable to plot the research convergence
cores	Number of cores to use for parallelism

Value

An object with the data, the medoids id (name of the line), the clustering and the distance matrix

Examples

```
#creating sequences
library(TraMineR)
data(mvad)
mvad.labels <- c("employment", "further education", "higher education","joblessness", "school", "training")
mvad.scode <- c("EM", "FE", "HE", "JL", "SC", "TR")
mvad.seq <- seqdef(mvad, 17:86, states = mvad.scode,abels = mvad.labels, xtstep = 6)

#CLARANS Clustering
my_cluster <- clarans_clust(mvad.seq,nb_cluster = 4, maxneighbours = 20, numlocal = 4, plot = TRUE)
```

---

clara_clust *CLARA clustering*

Description

With the help of TraMineR package, CLARA clustering provide a clustering of big dataset.
The main objective is to cluster state sequences with the "LCS" distance calculation method to find the best partition in N clusters.

Usage

```
clara_clust(
  data,
  nb_sample = 100,
  size_sample = 40 + 2 * nb_cluster,
  nb_cluster = 4,
  distargs = list(method = "LCS"),
  plot = FALSE,
  find_best_method = "Distance",
  with.diss = TRUE,
  cores = detectCores() - 1
)
```

Arguments

data	The dataset to use. In case of sequences, use seqdef (from TraMineR package) to create such an object.
nb_sample	The number of subsets to test.
size_sample	The size of each subset
nb_cluster	The number of medoids
distargs	List with method parameters to apply. (See the function seqdist in TraMineR package)
plot	Boolean variable to plot the result of clustering
find_best_method	Method to select the best subset. "Distance" is for the mean distance and "DB" is for Davies-Bouldin value.
with.diss	Boolean if the distance matrix should be returned
cores	Number of cores to use for parallelism

Value

An object with the data, the medoids id (name of the line), the clustering and the distance matrix

Examples

```
#creating sequences
library(TraMineR)
data(mvad)
mvad.labels <- c("employment", "further education", "higher education","joblessness", "school", "training")
mvad.scode <- c("EM", "FE", "HE", "JL", "SC", "TR")
mvad.seq <- seqdef(mvad, 17:86, states = mvad.scode,abels = mvad.labels, xtstep = 6)

#CLARA Clustering
my_cluster <- clara_clust(mvad.seq,nb_cluster = 4, nb_sample = 10, size_sample = 20, with.diss = TRUE)

#CLARA Clustering with Davies-Bouldin Method
my_cluster <- clara_clust(mvad.seq,nb_cluster = 4, nb_sample = 10, size_sample = 20, with.diss = TRUE, find_bes
```

davies_bouldin *Davies Bouldin Index*

Description

Implementation of Davies-Bouldin index to evaluate the quality of CLARA Clustering.

Usage

```
davies_bouldin(
  seq_obj,
  distargs = list(method = "LCS"),
  diss = TRUE,
  plot = TRUE,
  cores = detectCores() - 1
)
```

Arguments

seq_obj	The object generated with CLARA Clustering
distargs	List with method parameters to apply. (See the function seqdist in TraMineR package)
diss	Boolean to express if the parameter diss from CLARA.seq clustering has been returned (Matrix size must be k columns and n rows - see refseq function from TraMineR package)
plot	Boolean variable to plot the research convergence
cores	Number of cores to use for parallelism

Value

The value of the index

Examples

```
## Not run:
my_index <- davies_bouldin(my_cluster)

## End(Not run)
```

fuzzy_clust *FUZZY-CLARA clustering*

Description

With the help of TraMineR package, FUZZY-CLARA clustering provide a clustering of big dataset. The main objective is to cluster state sequences with the "LCS" distance calculation method to find the best partition in N clusters.
This function is a mix between CLARA and the ROBUST FUZZY C-MEDOIDS.

WARNING : This function is not finished yet !

Usage

```
fuzzy_clust(
  data,
  nb_sample = 100,
  size_sample = 40 + 2 * nb_cluster,
  nb_cluster = 4,
  distargs = list(method = "LCS"),
  fuzzyfier = 2,
  p = 5,
  threshold = 10,
  max_iter = 10,
  noise = 0.5,
  plot = FALSE,
  cores = detectCores() - 1
)
```

Arguments

data	The dataset to use. In case of sequences, use seqdef (from TraMineR package) to create such an object.
nb_sample	The number of subsets to test.
size_sample	The size of each subset
nb_cluster	The number of medoids
distargs	List with method parameters to apply. (See the function seqdist in TraMineR package)
fuzzyfier	Value of the fuzzifier (default is 2, which is the traditionnal value)
p	Number of candidate to test to be a better medoid
threshold	Variable to exclude outliers, whose values are greater than threshold
max_iter	Number of maximal iteration to do to find the set of medoids
noise	Small value to avoid divisions by 0 error
plot	Boolean variable to plot the research convergence
cores	Number of cores to use for parallelism

Value

An object with the data, the medoids id (name of the line), the clustering and the distance matrix

Examples

```
#creating sequences
library(TraMineR)
data(mvad)
mvad.labels <- c("employment", "further education", "higher education","joblessness", "school", "training")
mvad.scode <- c("EM", "FE", "HE", "JL", "SC", "TR")
mvad.seq <- seqdef(mvad, 17:86, states = mvad.scode,abels = mvad.labels, xtstep = 6)

#CLARA-FUzZY Clustering
my_cluster <- fuzzy_clust(mvad.seq,nb_sample = 14, size_sample = 50, plot = TRUE, threshold = 7, max_iter = 10,
```

Index