Package ‘trimcluster’

July 17, 2025

Title Cluster Analysis with Trimming

Version 0.2-0

VersionNote Released 0.1-6 on 2025-06-28 on CRAN

Depends R (>= 1.9.0)

Imports tclust

Suggests fpc

Description Trimmed k-means clustering. The method is described in Cuesta-Albertos et al. (1997) <doi:10.1214/aos/1031833664>.

Maintainer Valentin Todorov <`valentin@todorov.at`>

License GPL (>= 3)

URL `https://github.com/valentint/trimcluster`

BugReports `https://github.com/valentint/trimcluster/issues`

Repository CRAN

Date/Publication 2025-07-17 08:40:01 UTC

NeedsCompilation no

Author Christian Hennig [aut],
Valentin Todorov [cre] (ORCID: <`https://orcid.org/0000-0003-4215-0245`>)

Contents

`trimkmeans` *Trimmed k-means clustering*

Description

The trimmed k-means clustering method by Cuesta-Albertos, Gordaliza and Matran (1997). This optimizes the k-means criterion under trimming a portion of the points.

Usage

```
  trimkmeans(data,k,trim=0.1, scaling=FALSE,
        runs=500, niter1=3, niter2=20, nkeep=5, points=NULL,
        countmode, printcrit, maxit,
        parallel=FALSE, n.cores=-1, trace=0, ...)

  ## S3 method for class 'tkm'
print(x, ...)
  ## S3 method for class 'tkm'
plot(x, data, ...)
```

Arguments

`data`	matrix or data.frame with raw data
`k`	integer. Number of clusters.
`trim`	numeric between 0 and 1. Proportion of points to be trimmed.
`scaling`	logical. If `TRUE`, the variables are centered at their means and scaled to unit variance before execution.
`runs`	The number of random initializations to be performed.
`niter1`	The number of concentration steps to be performed for the nstart initializations.
`niter2`	The maximum number of concentration steps to be performed for the `nkeep` solutions kept for further iteration. The concentration steps are stopped, whenever two consecutive steps lead to the same data partition.
`nkeep`	The number of iterated initializations (after niter1 concentration steps) with the best values in the target function that are kept for further iterations
`points`	`NULL` or a matrix with k vectors used as means to initialize the algorithm. If initial mean vectors are specified, `runs` should be 1 (otherwise the same initial means are used for all runs).
`countmode`	(deprecated) optional positive integer. Every `countmode` algorithm runs `trimkmeans` shows a message.
`printcrit`	(deprecated) logical. If `TRUE`, all criterion values (mean squares) of the algorithm runs are printed.
`maxit`	(deprecated, use the combination `nkeep, niter1` and `niter2`) The maximum number of concentration steps to be performed. The concentration steps are stopped, whenever two consecutive steps lead to the same data partition.

`parallel`	A logical value, specifying whether the nstart initializations should be done in parallel.
`n.cores`	The number of cores to use when paralellizing, only taken into account if parallel=TRUE.
`trace`	Defines the tracing level, which is set to 0 by default. Tracing level 1 gives additional information on the stage of the iterative process.
`x`	object of class `tkm`.
`...`	further arguments to be transferred to `plot` or `plotcluster`.

Details

The function `trimkmeans()` now calls the function `tkmeans()` from the package `tclust`. This makes the procedure much faster since (a) `tkmeans()` is implemented in C++, (b) a new random initialization is introduced (see the parameters `niter1`, `niter2` and `nkeep` which replace the previous `maxit` and (c) it is posible to run the initialization in parallel (see the argument `parallel` and `ncores`.

`plot.tkm` calls `plotcluster` if the dimensionality of the data `p` is 1, shows a scatterplot with non-trimmed regions if `p=2` and discriminant coordinates computed from the clusters (ignoring the trimmed points) if `p>2`.

Value

An object of class 'tkm' which is a LIST with components

`classification`	integer vector coding cluster membership with trimmed observations coded as `k+1`.
`means`	numerical matrix giving the mean vectors of the k classes.
`disttom`	vector of squared Euclidean distances of all points to the closest mean.
`ropt`	maximum value of `disttom` so that the corresponding point is not trimmed.
`k`	see above.
`trim`	see above.
`runs`	see above.
`scaling`	see above.

Author(s)

Christian Hennig <`chrish@stats.ucl.ac.uk`> `http://www.homepages.ucl.ac.uk/~ucakche/`

References

Cuesta-Albertos, J. A., Gordaliza, A., and Matran, C. (1997) Trimmed k-Means: An Attempt to Robustify Quantizers, Annals of Statistics, 25, 553-576.

See Also

`plotcluster`

Examples

```
  set.seed(10001)
  n1 <-60
  n2 <-60
  n3 <-70
  n0 <-10
  nn <- n1+n2+n3+n0
  pp <- 2
  X <- matrix(rep(0,nn*pp),nrow=nn)
  ii <-0
  for (i in 1:n1){
    ii <-ii+1
    X[ii,] <- c(5,-5)+rnorm(2)
  }
  for (i in 1:n2){
    ii <- ii+1
    X[ii,] <- c(5,5)+rnorm(2)*0.75
  }
  for (i in 1:n3){
    ii <- ii+1
    X[ii,] <- c(-5,-5)+rnorm(2)*0.75
  }
  for (i in 1:n0){
    ii <- ii+1
    X[ii,] <- rnorm(2)*8
  }
  tkm1 <- trimkmeans(X, k=3, trim=0.1, runs=5)
## runs=5 is used to save computing time; runs must be >= nkeep

  print(tkm1)
  plot(tkm1,X)
```

Index