

Package ‘trimcluster’

April 19, 2009

Title Cluster analysis with trimming

Version 0.1-2

Date 2007-11-05

Author Christian Hennig <chrish@stats.ucl.ac.uk>

Depends R (>= 1.9.0)

Suggests fpc

Description Trimmed k-means clustering.

Maintainer Christian Hennig <chrish@stats.ucl.ac.uk>

License GPL

URL <http://www.homepages.ucl.ac.uk/~ucakche/>

Repository CRAN

Date/Publication 2007-11-05 19:22:18

R topics documented:

| | |
|----------------------|----------|
| trimkmeans | 1 |
| Index | 5 |

| | |
|------------|-----------------------------------|
| trimkmeans | <i>Trimmed k-means clustering</i> |
|------------|-----------------------------------|

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=100, points=NULL,
           countmode=runs+1, printcrit=FALSE,
           maxit=2*nrow(as.matrix(data)))

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

Arguments

| | |
|------------------------|--|
| <code>data</code> | matrix or data.frame with raw data |
| <code>k</code> | integer. Number of clusters. |
| <code>trim</code> | numeric between 0 and 1. Proportion of points to be trimmed. |
| <code>scaling</code> | logical. If TRUE, the variables are centered at their means and scaled to unit variance before execution. |
| <code>runs</code> | integer. Number of algorithm runs from initial means (randomly chosen from the data points). |
| <code>points</code> | NULL or a matrix with k vectors used as means to initialize the algorithm. If initial mean vectors are specified, <code>runs</code> should be 1 (otherwise the same initial means are used for all runs). |
| <code>countmode</code> | optional positive integer. Every <code>countmode</code> algorithm runs <code>trimkmeans</code> shows a message. |
| <code>printcrit</code> | logical. If TRUE, all criterion values (mean squares) of the algorithm runs are printed. |
| <code>maxit</code> | integer. Maximum number of iterations within an algorithm run. Each iteration determines all points which are closer to a different cluster center than the one to which they are currently assigned. The algorithm terminates if no more points have to be reassigned, or if <code>maxit</code> is reached. |
| <code>x</code> | object of class <code>tkm</code> . |
| <code>...</code> | further arguments to be transferred to <code>plot</code> or <code>plotcluster</code> . |

Details

`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

| | |
|-----------------------------|---|
| <code>classification</code> | integer vector coding cluster membership with trimmed observations coded as $k+1$. |
| <code>means</code> | numerical matrix giving the mean vectors of the k classes. |
| <code>disttom</code> | vector of squared Euclidean distances of all points to the closest mean. |
| <code>ropt</code> | maximum value of <code>disttom</code> so that the corresponding point is not trimmed. |
| <code>k</code> | see above. |
| <code>trim</code> | see above. |
| <code>runs</code> | see above. |
| <code>scaling</code> | 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
}
tkml <- trimkmeans(X,k=3,trim=0.1,runs=3)
# runs=3 is used to save computing time.
print(tkml)
plot(tkml,X)
```

Index

*Topic **cluster**

trimkmeans, 1

*Topic **multivariate**

trimkmeans, 1

plot.tkm(*trimkmeans*), 1

plotcluster, 2, 3

print.tkm(*trimkmeans*), 1

trimkmeans, 1