

Package ‘clustrd’

April 16, 2018

Type Package

Title Methods for Joint Dimension Reduction and Clustering

Description A class of methods that combine dimension reduction and clustering of continuous or categorical data. For continuous data, the package contains implementations of factorial K-means (Vichi and Kiers 2001; <DOI:10.1016/S0167-9473(00)00064-5>) and reduced K-means (De Soete and Carroll 1994; <DOI:10.1007/978-3-642-51175-2_24>); both methods that combine principal component analysis with K-means clustering. For categorical data, the package provides MCA K-means (Hwang, Dillon and Takane 2006; <DOI:10.1007/s11336-004-1173-x>), i-FCB (Iodice D'Enza and Palumbo 2013, <DOI:10.1007/s00180-012-0329-x>) and Cluster Correspondence Analysis (van de Velden, Iodice D'Enza and Palumbo 2017; <DOI:10.1007/s11336-016-9514-0>), which combine multiple correspondence analysis with K-means.

Version 1.2.1

Date 2018-04-16

Author Angelos Markos [aut, cre], Alfonso Iodice D'Enza [aut], Michel van de Velden [ctb]

Maintainer Angelos Markos <amarkos@gmail.com>

Depends ggplot2, dummies, grid

Imports corpcor, GGally, fpc, cluster, dplyr, plyr, ggrepel, ca, stats

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2018-04-16 16:50:38 UTC

R topics documented:

clusmca	2
cluspca	4
cmc	7
hsq	8
macro	10
plot.clusmca	10
plot.cluspca	12
tuneclus	13

clusmca	<i>Joint dimension reduction and clustering of categorical data.</i>
---------	--

Description

This function implements MCA K-means (Hwang, Dillon and Takane, 2006), i-FCB (Iodice D'Enza and Palumbo, 2013) and Cluster Correspondence Analysis (van de Velden, Iodice D'Enza and Palumbo, 2017). The methods combine variants of Correspondence Analysis for dimension reduction with K-means for clustering.

Usage

```
clusmca(data, nclus, ndim, method=c("clusCA","iFCB","MCAk"),
        alphak = .5, nstart = 100, smartStart = NULL, gamma = TRUE,
        seed = 1234)

## S3 method for class 'clusmca'
print(x, ...)

## S3 method for class 'clusmca'
summary(object, ...)

## S3 method for class 'clusmca'
fitted(object, mth = c("centers", "classes"), ...)
```

Arguments

data	Dataset with categorical variables
nclus	Number of clusters (nclus = 1 returns the MCA solution; see Details)
ndim	Dimensionality of the solution
method	Specifies the method. Options are MCAk for MCA K-means, iFCB for Iterative Factorial Clustering of Binary variables and clusCA for Cluster Correspondence Analysis (default = "clusCA")
alphak	Non-negative scalar to adjust for the relative importance of MCA (alphak = 1) and K-means (alphak = 0) in the solution (default = .5). Works only in combination with method = "MCAk"
nstart	Number of random starts (default = 100)
smartStart	If NULL then a random cluster membership vector is generated. Alternatively, a cluster membership vector can be provided as a starting solution
gamma	Scaling parameter that leads to similar spread in the object and variable scores (default = TRUE)
seed	An integer that is used as argument by <code>set.seed()</code> for offsetting the random number generator when <code>smartStart = NULL</code> . The default value is 1234

x	For the print method, a class of clusmca
object	For the summary method, a class of clusmca
mth	For the fitted method, a character string that specifies the type of fitted value to return: "centers" for the observations center vector, or "class" for the observations cluster membership value
...	Not used

Details

For the K-means part, the algorithm of Hartigan-Wong is used by default.

The hidden print and summary methods print out some key components of an object of class clusmca.

The hidden fitted method returns cluster fitted values. If method is "classes", this is a vector of cluster membership (the cluster component of the "clusmca" object). If method is "centers", this is a matrix where each row is the cluster center for the observation. The rownames of the matrix are the cluster membership values.

When nclus = 1 the function returns the MCA solution with objects in principal and variables in standard coordinates (plot(object) shows the corresponding asymmetric biplot).

Value

obscoord	Object scores
attcoord	Variable scores
centroid	Cluster centroids
cluster	Cluster membership
criterion	Optimal value of the objective criterion
size	The number of objects in each cluster
nstart	A copy of nstart in the return object
odata	A copy of data in the return object

References

Hwang, H., Dillon, W. R., and Takane, Y. (2006). An extension of multiple correspondence analysis for identifying heterogeneous subgroups of respondents. *Psychometrika*, 71, 161-171.

Iodice D'Enza, A., and Palumbo, F. (2013). Iterative factor clustering of binary data. *Computational Statistics*, 28(2), 789-807.

van de Velden M., Iodice D'Enza, A., and Palumbo, F. (2017). Cluster correspondence analysis. *Psychometrika*, 82(1), 158-185.

See Also

[cluspca](#), [tuneclus](#)

Examples

```

data(cmc)
# Preprocessing: values of wife's age and number of children were categorized
# into three groups based on quartiles
cmc$W_AGE = ordered(cut(cmc$W_AGE, c(16,26,39,49), include.lowest = TRUE))
levels(cmc$W_AGE) = c("16-26", "27-39", "40-49")
cmc$NCHILD = ordered(cut(cmc$NCHILD, c(0,1,4,17), right = FALSE))
levels(cmc$NCHILD) = c("0", "1-4", "5 and above")

#Cluster Correspondence Analysis solution with 3 clusters in 2 dimensions
#after 10 random starts
outclusCA = clusmca(cmc, 3, 2, method = "clusCA", nstart = 10)
outclusCA
#Scatterplot (dimensions 1 and 2)
plot(outclusCA)

#MCA K-means solution with 3 clusters in 2 dimensions after 10 random starts
outMCAk = clusmca(cmc, 3, 2, method = "MCAk", nstart = 10)
outMCAk
#Scatterplot (dimensions 1 and 2)
plot(outMCAk)

#nclus = 1 just gives the MCA solution
#outMCA = clusmca(cmc, 1, 2)
#outMCA
#Scatterplot (dimensions 1 and 2)
#asymmetric biplot with scaling gamma = TRUE
#plot(outMCA)

```

cluspca

Joint dimension reduction and clustering of continuous data.

Description

This function implements Factorial K-means (Vichi and Kiers, 2001) and Reduced K-means (De Soete and Carroll, 1994), as well as a compromise version of these two methods. The methods combine Principal Component Analysis for dimension reduction with K-means for clustering.

Usage

```

cluspca(data, nclus, ndim, alpha = NULL, method = c("RKM", "FKM"),
center = TRUE, scale = TRUE, rotation = "none", nstart = 100,
smartStart = NULL, seed = 1234)

## S3 method for class 'cluspca'
print(x, ...)

## S3 method for class 'cluspca'
summary(object, ...)

```

```
## S3 method for class 'cluspca'
fitted(object, mth = c("centers", "classes"), ...)
```

Arguments

<code>data</code>	Dataset with metric variables
<code>nclus</code>	Number of clusters (<code>nclus = 1</code> returns the PCA solution; see Details)
<code>ndim</code>	Dimensionality of the solution
<code>method</code>	Specifies the method. Options are RKM for reduced K-means and FKM for factorial K-means (default = "RKM")
<code>alpha</code>	Adjusts for the relative importance of RKM and FKM in the objective function; <code>alpha = 0.5</code> leads to reduced K-means, <code>alpha = 0</code> to factorial K-means, and <code>alpha = 1</code> reduces to the tandem approach
<code>center</code>	A logical value indicating whether the variables should be shifted to be zero centered (default = TRUE)
<code>scale</code>	A logical value indicating whether the variables should be scaled to have unit variance before the analysis takes place (default = TRUE)
<code>rotation</code>	Specifies the method used to rotate the factors. Options are <code>none</code> for no rotation, <code>varimax</code> for varimax rotation with Kaiser normalization and <code>promax</code> for promax rotation (default = "none")
<code>nstart</code>	Number of starts (default = 100)
<code>smartStart</code>	If NULL then a random cluster membership vector is generated. Alternatively, a cluster membership vector can be provided as a starting solution
<code>seed</code>	An integer that is used as argument by <code>set.seed()</code> for offsetting the random number generator when <code>smartStart = NULL</code> . The default value is 1234
<code>x</code>	For the print method, a class of <code>clusmca</code>
<code>object</code>	For the summary method, a class of <code>clusmca</code>
<code>mth</code>	For the fitted method, a character string that specifies the type of fitted value to return: "centers" for the observations center vector, or "class" for the observations cluster membership value
<code>...</code>	Not used

Details

For the K-means part, the algorithm of Hartigan-Wong is used by default.

The hidden `print` and `summary` methods print out some key components of an object of class `cluspca`.

The hidden `fitted` method returns cluster fitted values. If `method` is "classes", this is a vector of cluster membership (the cluster component of the "cluspca" object). If `method` is "centers", this is a matrix where each row is the cluster center for the observation. The rownames of the matrix are the cluster membership values.

Value

obscoord	Object scores
attcoord	Variable scores
centroid	Cluster centroids
cluster	Cluster membership
criterion	Optimal value of the objective function
size	The number of objects in each cluster
scale	A copy of scale in the return object
center	A copy of center in the return object
nstart	A copy of nstart in the return object
odata	A copy of data in the return object

References

De Soete, G., and Carroll, J. D. (1994). K-means clustering in a low-dimensional Euclidean space. In Diday E. et al. (Eds.), *New Approaches in Classification and Data Analysis*, Heidelberg: Springer, 212-219.

Vichi, M., and Kiers, H.A.L. (2001). Factorial K-means analysis for two-way data. *Computational Statistics and Data Analysis*, 37, 49-64.

See Also

[clusmca](#), [tuneclus](#)

Examples

```
#Reduced K-means with 3 clusters in 2 dimensions after 10 random starts
data(macro)
outRKM = cluspca(macro, 3, 2, method = "RKM", rotation = "varimax", scale = FALSE, nstart = 10)
summary(outRKM)
#Scatterplot (dimensions 1 and 2) and cluster description plot
plot(outRKM, cludesc = TRUE)

#Factorial K-means with 3 clusters in 2 dimensions
#with a Reduced K-means starting solution
data(macro)
outFKM = cluspca(macro, 3, 2, method = "FKM", rotation = "varimax",
scale = FALSE, smartStart = outRKM$cluster)
outFKM
#Scatterplot (dimensions 1 and 2) and cluster description plot
plot(outFKM, cludesc = TRUE)

#To get the Tandem approach (PCA(SVD) + K-means)
outTandem = cluspca(macro, 3, 2, alpha = 1)
plot(outTandem)

#nclus = 1 just gives the PCA solution
```

```
#outPCA = cluspca(macro, 1, 2)
#outPCA
#Scatterplot (dimensions 1 and 2) with scaling gamma = TRUE
#plot(outPCA)
```

cmc

Contraceptive Choice in Indonesia

Description

Data of married women in Indonesia who were not pregnant (or did not know they were pregnant) at the time of the survey. The dataset contains demographic and socio-economic characteristics of the women along with their preferred method of contraception (no use, long-term methods, short-term methods).

Usage

```
data(cmc)
```

Format

A data frame containing 1,437 observations on the following 10 variables.

W_AGE wife's age in years.

W_EDU ordered factor indicating wife's education, with levels "low", "2", "3" and "high".

H_EDU ordered factor indicating wife's education, with levels "low", "2", "3" and "high".

NCHILD number of children.

W_REL factor indicating wife's religion, with levels "non-Islam" and "Islam".

W_WORK factor indicating if the wife is working.

H_OCC factor indicating husband's occupation, with levels "1", "2", "3" and "4". The labels are not known.

SOL ordered factor indicating the standard of living index with levels "low", "2", "3" and "high".

MEDEXP factor indicating media exposure, with levels "good" and "not good".

CM factor indicating the contraceptive method used, with levels "no-use", "long-term" and "short-term".

Source

This dataset is part of the 1987 National Indonesia Contraceptive Prevalence Survey and was created by Tjen-Sien Lim. It has been taken from the UCI Machine Learning Repository at <http://archive.ics.uci.edu/ml/>.

References

Lim, T.-S., Loh, W.-Y. & Shih, Y.-S. (1999). A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-three Old and New Classification Algorithms. *Machine Learning*, 40(3), 203-228.

Examples

```
data(cmc)
```

```
hsq
```

```
Humor Styles
```

Description

The dataset was collected with an interactive online version of the Humor Styles Questionnaire (HSQ) which assesses four independent ways in which people express and appreciate humor (Martin et al. 2003): affiliative, defined as the benign uses of humor to enhance one's relationships with others; self-enhancing, indicating uses of humor to enhance the self; aggressive, the use of humor to enhance the self at the expense of others; self-defeating the use of humor to enhance relationships at the expense of oneself. The main part of the questionnaire consisted of 32 statements rated from 1 to 5 according to the respondents' level of agreement. Three more questions were included (age, gender and self-reported accuracy of answer). The number of respondents is 993, after removing the cases with missing values in the 32 statements.

Usage

```
data("hsq")
```

Format

A data frame with 993 observations on 35 variables. The first 32 variables are Likert-type statements with 5 response categories, ranging from 1 (strong agreement) to 5 (strong disagreement).

- AF1 I usually don't laugh or joke around much with other people
- AF2 If I am feeling depressed, I can usually cheer myself up with humor
- AF3 If someone makes a mistake, I will often tease them about it
- AF4 I let people laugh at me or make fun at my expense more than I should
- AF5 I don't have to work very hard at making other people laugh - I seem to be a naturally humorous person
- AF6 Even when I'm by myself, I'm often amused by the absurdities of life
- AF7 People are never offended or hurt by my sense of humor
- AF8 I will often get carried away in putting myself down if it makes my family or friends laugh
- SE1 I rarely make other people laugh by telling funny stories about myself
- SE2 If I am feeling upset or unhappy I usually try to think of something funny about the situation to make myself feel better
- SE3 When telling jokes or saying funny things, I am usually not very concerned about how other people are taking it
- SE4 I often try to make people like or accept me more by saying something funny about my own weaknesses, blunders, or faults

- SE5 I laugh and joke a lot with my closest friends
- SE6 My humorous outlook on life keeps me from getting overly upset or depressed about things
- SE7 I do not like it when people use humor as a way of criticizing or putting someone down
- SE8 I don't often say funny things to put myself down
- AG1 I usually don't like to tell jokes or amuse people
- AG2 If I'm by myself and I'm feeling unhappy, I make an effort to think of something funny to cheer myself up
- AG3 Sometimes I think of something that is so funny that I can't stop myself from saying it, even if it is not appropriate for the situation
- AG4 I often go overboard in putting myself down when I am making jokes or trying to be funny
- AG5 I enjoy making people laugh
- AG6 If I am feeling sad or upset, I usually lose my sense of humor
- AG7 I never participate in laughing at others even if all my friends are doing it
- AG8 When I am with friends or family, I often seem to be the one that other people make fun of or joke about
- SD1 I don't often joke around with my friends
- SD2 It is my experience that thinking about some amusing aspect of a situation is often a very effective way of coping with problems
- SD3 If I don't like someone, I often use humor or teasing to put them down
- SD4 If I am having problems or feeling unhappy, I often cover it up by joking around, so that even my closest friends don't know how I really feel
- SD5 I usually can't think of witty things to say when I'm with other people
- SD6 I don't need to be with other people to feel amused - I can usually find things to laugh about even when I'm by myself
- SD7 Even if something is really funny to me, I will not laugh or joke about it if someone will be offended
- SD8 Letting others laugh at me is my way of keeping my friends and family in good spirits

References

Martin, R. A., Puhlik-Doris, P., Larsen, G., Gray, J., & Weir, K. (2003). Individual differences in uses of humor and their relation to psychological well-being: Development of the Humor Styles Questionnaire. *Journal of Research in Personality, 37*(1), 48-75.

Examples

```
data(hsq)
```

macro

Economic Indicators of 20 OECD countries for 1999

Description

Data on the macroeconomic performance of national economies of 20 countries, members of the OECD (September 1999). The performance of the economies reflects the interaction of six main economic indicators (percentage change from the previous year): gross domestic product (GDP), leading indicator (LI), unemployment rate (UR), interest rate (IR), trade balance (TB), net national savings (NNS).

Usage

```
data(macro)
```

Format

A data frame with 20 observations on the following 6 variables.

GDP numeric

LI numeric

UR numeric

IR numeric

TB numeric

NNS numeric

References

Vichi, M. & Kiers, H. A. (2001). Factorial k-means analysis for two-way data. *Computational Statistics & Data Analysis*, 37(1), 49-64.

plot.clusmca

Plotting function for clusmca() output.

Description

Plotting function that creates a scatterplot of the object scores and/or the attribute scores and the cluster centroids. Optionally, the function returns a series of barplots showing the standardized residuals per attribute for each cluster.

Usage

```
## S3 method for class 'clusmca'  
plot(x, dims = c(1,2), what = c(TRUE,TRUE),  
      cludesc = FALSE, topstdres = 20, attlabs = NULL, binary = FALSE, subplot = FALSE, ...)
```

Arguments

x	Object returned by clusmca()
dims	Numerical vector of length 2 indicating the dimensions to plot on horizontal and vertical axes respectively; default is first dimension horizontal and second dimension vertical
what	Vector of two logical values specifying the contents of the plots. First entry indicates whether a scatterplot of the objects is displayed in principal coordinates. Second entry indicates whether a scatterplot of the attribute categories is displayed in principal coordinates. Cluster centroids are always displayed. The default is c(TRUE, TRUE) and the resultant plot is a biplot of both objects and attribute categories with gamma-based scaling (see van de Velden et al., 2017)
cludesc	A logical value indicating whether a series of barplots is produced showing the largest (in absolute value) standardized residuals per attribute for each cluster (default = FALSE)
topstdres	Number of largest standardized residuals used to describe each cluster (default = 20). Works only in combination with cludesc = TRUE
attlabs	Vector of custom attribute labels; if not provided, default labeling is applied
subplot	A logical value indicating whether a subplot with the full distribution of the standardized residuals will appear at the bottom left corner of the corresponding plots. Works only in combination with cludesc = TRUE
binary	A logical value indicating whether the visualization refers to a dataset of binary variables
...	Further arguments to be transferred to clusmca()

Value

The function returns a ggplot2 scatterplot of the solution obtained via clusmca() that can be further customized using the **ggplot2** package. When cludesc = TRUE the function also returns a series of ggplot2 barplots showing the largest (or all) standardized residuals per attribute for each cluster.

References

- Hwang, H., Dillon, W. R., and Takane, Y. (2006). An extension of multiple correspondence analysis for identifying heterogenous subgroups of respondents. *Psychometrika*, 71, 161-171.
- Iodice D'Enza, A., and Palumbo, F. (2013). Iterative factor clustering of binary data. *Computational Statistics*, 28(2), 789-807.
- van de Velden M., Iodice D'Enza, A., and Palumbo, F. (2017). Cluster correspondence analysis. *Psychometrika*, 82(1), 158-185.

See Also

[plot.cluspca](#)

Examples

```

data("hsq")
#Cluster Correspondence Analysis with 3 clusters in 2 dimensions after 10 random starts
outclusMCA = clusmca(hsq[,1:8], 3, 2, nstart = 10)
#Save the ggplot2 scatterplot
map = plot(outclusMCA)
#Customization (adding titles)
map + ggtitle(paste("Cluster CA plot of the hsq data: 3 clusters of sizes ",
paste(outclusMCA$size, collapse = ", "), sep = "")) +
xlab("Dim. 1") + ylab("Dim. 2") +
theme(plot.title = element_text(size = 10, face = "bold", hjust = 0.5))

data("hsq")
#i-FCB with 4 clusters in 3 dimensions after 10 random starts
outclusMCA = clusmca(hsq[,1:8], 4, 3, method = "iFCB", nstart= 10)
#Scatterplot with the observations only (dimensions 1 and 3)
#and cluster description plots showing the 20 largest std. residuals
#(with the full distribution showing in subplots)
plot(outclusMCA, dim = c(1,3), what = c(TRUE, FALSE), cludesc = TRUE,
subplot = TRUE)

```

plot.cluspca

Plotting function for cluspca() output.

Description

Plotting function that creates a scatterplot of the objects, a correlation circle of the variables or a biplot of both objects and variables. Optionally, it returns a parallel coordinate plot showing cluster means.

Usage

```

## S3 method for class 'cluspca'
plot(x, dims = c(1, 2), cludesc = FALSE, what = c(TRUE,TRUE), atlabs, ...)

```

Arguments

x	Object returned by cluspca()
dims	Numerical vector of length 2 indicating the dimensions to plot on horizontal and vertical axes respectively; default is first dimension horizontal and second dimension vertical
what	Vector of two logical values specifying the contents of the plots. First entry indicates whether a scatterplot of the objects and cluster centroids is displayed and the second entry whether a correlation circle of the variables is displayed. The default is c(TRUE, TRUE) and the resultant plot is a biplot of both objects and variables
cludesc	A logical value indicating if a parallel coordinate plot showing cluster means is produced (default = FALSE)

attnlabs Vector of custom attribute labels; if not provided, default labeling is applied
 ... Further arguments to be transferred to cluspca()

Value

The function returns a ggplot2 scatterplot of the solution obtained via cluspca() that can be further customized using the **ggplot2** package. When cludesc = TRUE the function also returns a ggplot2 parallel coordinate plot.

References

De Soete, G., and Carroll, J. D. (1994). K-means clustering in a low-dimensional Euclidean space. In Diday E. et al. (Eds.), *New Approaches in Classification and Data Analysis*, Heidelberg: Springer, 212-219.

Vichi, M., and Kiers, H.A.L. (2001). Factorial K-means analysis for two-way data. *Computational Statistics and Data Analysis*, 37, 49-64.

See Also

[plot.clusmca](#)

Examples

```
data("macro")
#Factorial K-means (3 clusters in 2 dimensions) after 100 random starts
outFKM = cluspca(macro, 3, 2, method = "FKM", rotation = "varimax")
#Scatterplot (dimensions 1 and 2) and cluster description plot
plot(outFKM, cludesc = TRUE)

data("iris", package = "datasets")
#Compromise solution between PCA and Reduced K-means
#on the iris dataset (3 clusters in 2 dimensions) after 100 random starts
outclusPCA = cluspca(iris[,-5], 3, 2, alpha = 0.3, rotation = "varimax")
table(outclusPCA$cluster, iris[,5])
#Save the ggplot2 scatterplot
map = plot(outclusPCA)
#Customization (adding titles)
map + ggtitle(paste("A compromise solution between RKM and FKM on the iris:
3 clusters of sizes ", paste(outclusPCA$size,
collapse = ", "), sep = "")) + xlab("Dimension 1") + ylab("Dimension 2") +
theme(plot.title = element_text(size = 10, face = "bold", hjust = 0.5))
```

tuneclus

Cluster quality assessment for a range of clusters and dimensions.

Description

This function facilitates the selection of the appropriate number of clusters and dimensions for joint dimension reduction and clustering methods.

Usage

```
tuneclus(data, nclusrange = 3:4, ndimrange = 2:3,
method = c("RKM", "FKM", "clusCA", "iFCB", "MCAk"),
criterion = "asw", dst = "full", alpha = NULL, alphak = NULL,
center = TRUE, scale = TRUE, rotation = "none", nstart = 100,
smartStart = NULL, seed = 1234)

## S3 method for class 'tuneclus'
print(x, ...)

## S3 method for class 'tuneclus'
summary(object, ...)

## S3 method for class 'tuneclus'
fitted(object, mth = c("centers", "classes"), ...)
```

Arguments

data	Continuous or Categorical dataset
nclusrange	An integer vector with the range of numbers of clusters which are to be compared by the cluster validity criteria. Note: the number of clusters should be greater than one
ndimrange	An integer vector with the range of dimensions which are to be compared by the cluster validity criteria
method	Specifies the method. Options are RKM for reduced K-means, FKM for factorial K-means, MCAk for MCA K-means, iFCB for Iterative Factorial Clustering of Binary variables and clusCA for Cluster Correspondence Analysis
criterion	One of asw, ch or crit. Determines whether average silhouette width, Calinski-Harabasz index or objective value of the selected method is used (default = "asw")
dst	Specifies the data used to compute the distances between objects. Options are full for the original data (after possible scaling) and low for the object scores in the low-dimensional space (default = "full")
alpha	Adjusts for the relative importance of RKM and FKM in the objective function; alpha = 1 reduces to PCA, alpha = 0.5 to reduced K-means, and alpha = 0 to factorial K-means
alphak	Non-negative scalar to adjust for the relative importance of MCA (alphak = 1) and K-means (alphak = 0) in the solution (default = .5). Works only in combination with method = "MCAk"
center	A logical value indicating whether the variables should be shifted to be zero centered (default = TRUE)
scale	A logical value indicating whether the variables should be scaled to have unit variance before the analysis takes place (default = TRUE)

rotation	Specifies the method used to rotate the factors. Options are none for no rotation, varimax for varimax rotation with Kaiser normalization and promax for promax rotation (default = "none")
nstart	Number of starts (default = 100)
smartStart	If NULL then a random cluster membership vector is generated. Alternatively, a cluster membership vector can be provided as a starting solution
seed	An integer that is used as argument by <code>set.seed()</code> for offsetting the random number generator when <code>smartStart = NULL</code> . The default value is 1234
x	For the print method, a class of <code>clusmca</code>
object	For the summary method, a class of <code>clusmca</code>
mth	For the fitted method, a character string that specifies the type of fitted value to return: "centers" for the observations center vector, or "class" for the observations cluster membership value
...	Not used

Details

For the K-means part, the algorithm of Hartigan-Wong is used by default.

The hidden print and summary methods print out some key components of an object of class `tuneclus`.

The hidden fitted method returns cluster fitted values. If method is "classes", this is a vector of cluster membership (the cluster component of the "tuneclus" object). If method is "centers", this is a matrix where each row is the cluster center for the observation. The rownames of the matrix are the cluster membership values.

Value

<code>clusobjbest</code>	The output of the optimal run of <code>cluspca()</code> or <code>clusmca()</code>
<code>nclusbest</code>	The optimal number of clusters
<code>ndimbest</code>	The optimal number of dimensions
<code>critbest</code>	The optimal criterion value for <code>nclusbest</code> clusters and <code>ndimbest</code> dimensions
<code>critgrid</code>	Matrix of size <code>nclusrange</code> x <code>ndimrange</code> with the criterion values for the specified ranges of clusters and dimensions (values are calculated only when the number of clusters is greater than the number of dimensions; otherwise values in the grid are left blank)

References

Calinski, R.B., and Harabasz, J., (1974). A dendrite method for cluster analysis. *Communications in Statistics*, 3, 1-27.

Kaufman, L., and Rousseeuw, P.J., (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York.

See Also

[cluspca](#), [clusmca](#)

Examples

```
# Reduced K-means for a range of clusters and dimensions
data(macro)
# Cluster quality assessment based on the average silhouette width # in the low dimensional space
bestRKM = tuneclus(macro, 3:4, 2:3, method = "RKM", criterion = "asw", dst = "low", nstart = 10)
bestRKM
plot(bestRKM)

# Cluster Correspondence Analysis for a range of clusters and dimensions
data(hsq)
# Cluster quality assessment based on the average silhouette width # in the full dimensional space
bestclusCA = tuneclus(hsq[,1:4], 3:4, 2:3, method = "clusCA", criterion = "asw", nstart = 10)
bestclusCA
plot(bestclusCA)
```


Index

*Topic **datasets**

cmc, [7](#)

hsq, [8](#)

macro, [10](#)

clusmca, [2](#), [6](#), [15](#)

cluspca, [3](#), [4](#), [15](#)

cmc, [7](#)

fitted.clusmca (clusmca), [2](#)

fitted.cluspca (cluspca), [4](#)

fitted.tuneclus (tuneclus), [13](#)

hsq, [8](#)

macro, [10](#)

plot.clusmca, [10](#), [13](#)

plot.cluspca, [11](#), [12](#)

print.clusmca (clusmca), [2](#)

print.cluspca (cluspca), [4](#)

print.tuneclus (tuneclus), [13](#)

summary.clusmca (clusmca), [2](#)

summary.cluspca (cluspca), [4](#)

summary.tuneclus (tuneclus), [13](#)

tuneclus, [3](#), [6](#), [13](#)