# Package 'clustrd'

July 17, 2022

**Type** Package

**Title** Methods for Joint Dimension Reduction and Clustering

**Description** A class of methods that combine dimension reduction and clustering of continuous, categorical or mixed-
type data (Markos, Iodice D'Enza and van de Velden 2019; <DOI:10.18637/jss.v091.i10>). 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. For mixed-
type data, it provides mixed Reduced K-means and mixed Factorial K-
means (van de Velden, Iodice D'Enza and Markos 2019; <DOI:10.1002/wics.1456>), which combine PCA for mixed-type data with K-means.

**Version** 1.4.0

**Date** 2022-07-16

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

**Maintainer** Angelos Markos <amarkos@gmail.com>

**Depends** ggplot2, grid

**Imports** rARPACK, tibble, corpcor, GGally, fpc, cluster, dplyr, plyr,
ggrepel, ca, stats

**License** GPL-3

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-07-16 23:20:06 UTC

## R topics documented:

1

| clustrd-package | *Methods for Joint Dimension Reduction and Clustering* |
|---|---|

## Description

A class of methods that combine dimension reduction and clustering of continuous, categorical or mixed-type data (Markos, Iodice D'Enza and van de Velden 2019; <DOI:10.18637/jss.v091.i10>). 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. For mixed-type data, it provides mixed Reduced K-means and mixed Factorial K-means (van de Velden, Iodice D'Enza and Markos 2019; <DOI:10.1002/wics.1456>), which combine PCA for mixed-type data with K-means.

## Details

| | |
|---|---|
| Package: | clustrd |
| Type: | Package |
| Version: | 1.3.6-2 |
| Date: | 2019-10-28 |
| License: | GPL-3 |

**Author(s)**

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

**References**

Markos, A., Iodice D'Enza, A., & van de Velden, M. (2019). Beyond Tandem Analysis: Joint Dimension Reduction and Clustering in R. *Journal of Statistical Software*, *91*(10), 1–24. doi:10.18637/jss.v091.i10.

---

| bribery | *Bribery cases in Russia* |
|---|---|

---

**Description**

The data set refers to a collection of 55 articles on bribery cases from central Russian newspapers 1999-2000 (Mirkin, 2005). The variables reflect the following five-fold structure of bribery situations: two interacting sides - the office and the client, their interaction, the corrupt service rendered, and the environment in which it all occurs. These structural aspects can be characterized by 11 variables that have been manually recovered from the newspaper articles.

**Usage**

```
data("bribery")
```

**Format**

A data frame with 55 observations on 11 categorical variables.

Of  Type of Office

Cl  Level of Client

Serv  Type of service: obstruction of justice, favours, cover-up, change of category, extortion of money for rendering free services

Occ  Frequency of occurrence

Init  Who initiated the bribery act

Brib  Bribe Level in $

Typ  Type of corruption

Net  Corruption network

Con  Condition of corruption

Bran  Branch at which the corrupt service occurred

Pun  Punishment

**References**

Mirkin, B. (2005). *Clustering for data mining: a data recovery approach*. Chapman and Hall/CRC.

**Examples**

```
data(bribery)
```

---

clusmca                    *Joint dimension reduction and clustering of categorical data.*

---

## 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,
inboot = FALSE, seed = NULL)

## 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 set.seed() for offsetting the random number generator when smartStart = NULL. The default value is NULL. |
| inboot | Used internally in the bootstrap functions to perform bootstrapping on the indicator matrix. |

| 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 | Attribute 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 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

[cluspca](), [cluspcamix](), [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, seed = 1234)
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, seed = 1234)
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 = NULL)

## 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

| | |
|---|---|
| data | Dataset with metric variables |
| nclus | Number of clusters (nclus = 1 returns the PCA solution |
| ndim | Dimensionality of the solution |
| method | Specifies the method. Options are RKM for reduced K-means and FKM for factorial K-means (default = "RKM") |
| alpha | Adjusts for the relative importance of RKM and FKM in the objective function; alpha = 0.5 leads to reduced K-means, alpha = 0 to factorial K-means, and alpha = 1 reduces to the tandem approach (PCA followed by K-means) |
| 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 set.seed() for offsetting the random number generator when smartStart = NULL. The default value is NULL. |
| 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 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.

When nclus = 1 the function returns the PCA solution and plot(object) shows the corresponding biplot.

## 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](), [cluspcamix](), [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, seed = 1234)
plot(outTandem)

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

```
#outPCA = cluspca(macro, 1, 2)
#outPCA
#Scatterplot (dimensions 1 and 2)
#plot(outPCA)
```

---

cluspcamix                    *Joint dimension reduction and clustering of mixed-type data.*

---

### Description

This function implements clustering and dimension reduction for mixed-type variables, i.e., categorical and metric (see, Yamamoto & Hwang, 2014; van de Velden, Iodice D'Enza, & Markos 2019; Vichi, Vicari, & Kiers, 2019). This framework includes Mixed Reduced K-means and Mixed Factorial K-means, as well as a compromise of these two methods. The methods combine Principal Component Analysis of mixed-data for dimension reduction with K-means for clustering.

### Usage

```
cluspcamix(data, nclus, ndim, method=c("mixedRKM", "mixedFKM"),
center = TRUE, scale = TRUE, alpha=NULL, rotation="none",
nstart = 100, smartStart=NULL, seed=NULL, inboot = FALSE)

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

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

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

### Arguments

| | |
|---|---|
| data | Dataset with categorical and metric variables |
| nclus | Number of clusters (nclus = 1 returns the PCAMIX solution) |
| ndim | Dimensionality of the solution |
| method | Specifies the method. Options are mixedRKM for mixed reduced K-means and mixedFKM for mixed factorial K-means (default = "mixedRKM") |
| 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) |
| alpha | Adjusts for the relative importance of Mixed RKM and Mixed FKM in the objective function; alpha = 0.5 leads to mixed reduced K-means, alpha = 0 to mixed factorial K-means, and alpha = 1 reduces to the tandem approach (PCAMIX followed by K-means) |

| 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 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 |
| seed | An integer that is used as argument by set.seed() for offsetting the random number generator when smartStart = NULL. The default value is NULL. |
| inboot | Used internally in the bootstrap functions to perform bootstrapping on the indicator matrix. |
| x | For the print method, a class of cluspcamix |
| object | For the summary method, a class of cluspcamix |
| 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 cluspcamix.

The hidden fitted method returns cluster fitted values. If method is "classes", this is a vector of cluster membership (the cluster component of the "cluspcamix" 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 solution of PCAMIX and plot(object) shows the corresponding 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 |
| 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

van de Velden, M., Iodice D'Enza, A., & Markos, A. (2019). Distance-based clustering of mixed data. *Wiley Interdisciplinary Reviews: Computational Statistics*, e1456.

Vichi, M., Vicari, D., & Kiers, H.A.L. (2019). Clustering and dimension reduction for mixed variables. *Behaviormetrika*. doi:10.1007/s41237-018-0068-6.

Yamamoto, M., & Hwang, H. (2014). A general formulation of cluster analysis with dimension reduction and subspace separation. *Behaviormetrika*, *41*, 115-129.

## See Also

cluspca, clusmca, tuneclus

## Examples

```
data(diamond)
#Mixed Reduced K-means solution with 3 clusters in 2 dimensions
#after 10 random starts
outmixedRKM = cluspcamix(diamond, 3, 2, method = "mixedRKM", nstart = 10, seed = 1234)
outmixedRKM
#A graph with the categories and a biplot of the continuous variables (dimensions 1 and 2)
plot(outmixedRKM)

#Tandem analysis: PCAMIX or FAMD followed by K-means solution
#with 3 clusters in 2 dimensions after 10 random starts
outTandem = cluspcamix(diamond, 3, 2, alpha = 1, nstart = 10, seed = 1234)
outTandem
#Scatterplot (dimensions 1 and 2)
plot(outTandem)

#nclus = 1 just gives the PCAMIX or FAMD solution
#outPCAMIX = cluspcamix(diamond, 1, 2)
#outPCAMIX
#Biplot (dimensions 1 and 2)
#plot(outPCAMIX)
```

---

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)
```

---

diamond                          *Diamond Stone Pricing*

---

**Description**

Data on 308 diamond stones sold in Singapore. The main attributes are diamond weight, colour, clarity, certification body and price in Singapore $. The weight of a diamond stone is indicated in terms of carat units. Since stones may be divided into 3 clusters due to their size, namely small (less than 0.5 carats), medium (0.5 to less than 1 carat) and large (1 carat and over), following Chu (2001), three binary variables have been built representing the three caratage ranges, and three quantitative variables (denoted Small, Medium, Large) have been derived by multiplying such binary variables by carats. So, the "Small" variable has nonzero values (i.e., the carat values) only for the smallest diamonds (less than 0.5 carats), and likewise for the other two variables. Thus, these variables are

weighted binary variables. The colour of a diamond is graded from D (completely colourless), E, F, G, ..., to I (almost colorless). Clarity refers to the diamond's internal and external imperfections. Clarity is graded on a scale from IF (internally flawless), to very very slightly imperfect (VVS1 or VVS2), and very slightly imperfect, VS1 or VS2. Three certification bodies were used: New York based Gemmological Institute of America (GIA), Antwerp based International Gemmological Institute (IGI) and Hoge Raad Voor Diamant (HRD).

## Usage

```
data(diamond)
```

## Format

A data frame with 308 observations on the following 7 variables.

Small weighted binary variable with nonzero values (i.e., the carat values) for diamonds with less than 0.5 carats.

Medium weighted binary variable with nonzero values (i.e., the carat values) for diamonds from 0.5 to less than 1 carat.

Large weighted binary variable with nonzero values (i.e., the carat values) for diamonds from 1 carat and over.

Colour the color of the diamond with a factor with levels (D, E, F, G, H, I).

Clarity the clarity of the diamond with a factor with levels (IF, VVS1, VVS2, VS1, VS2).

Certification the certification body with a factor with levels (GIA, IGI, HRD).

Price the price of a diamond in Singapore $.

## References

Chu, S. (2001). Pricing the C's of Diamond Stones, *Journal of Statistics Education*, *9*(2).

---

| global_bootclus | *Global stabiliy assessment of Joint Dimension Reduction and Clustering methods by bootstrapping.* |
|---|---|

---

## Description

Runs joint dimension and clustering algorithms repeatedly for different numbers of clusters on bootstrap replica of the original data and returns corresponding cluster assignments, and cluster agreement indices comparing pairs of partitions.

## Usage

```
global_bootclus(data, nclusrange = 3:4, ndim = NULL,
method = c("RKM","FKM","mixedRKM","mixedFKM","clusCA","MCAk","iFCB"),
nboot = 10, alpha = NULL, alphak = NULL, center = TRUE,
scale = TRUE, nstart = 100, smartStart = NULL, seed = NULL)
```

**Arguments**

| | |
|---|---|
| data | Continuous, Categorical ot Mixed data set |
| nclusrange | An integer or an integer vector with the number of clusters or a range of numbers of clusters (should be greater than one) |
| ndim | Dimensionality of the solution; if NULL it is set to nclus - 1 |
| method | Specifies the method. Options are RKM for Reduced K-means, FKM for Factorial K-means, mixedRKM for Mixed Reduced K-means, mixedFKM for Mixed Factorial K-means, MCAk for MCA K-means, iFCB for Iterative Factorial Clustering of Binary variables and clusCA for Cluster Correspondence Analysis. |
| nboot | Number of bootstrap pairs of partitions |
| alpha | Adjusts for the relative importance of (mixed) RKM and FKM in the objective function; alpha = 1 reduces to PCA/PCAMIX, alpha = 0.5 to (mixed) reduced K-means, and alpha = 0 to (mixed) 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 metric variables should be shifted to be zero centered (default = TRUE) |
| scale | A logical value indicating whether the metric variables should be scaled to have unit variance before the analysis takes place (default = TRUE) |
| 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 |
| seed | An integer that is used as argument by set.seed() for offsetting the random number generator when smartStart = NULL. The default value is NULL. |

**Details**

The algorithm for assessing global cluster stability is similar to that in Dolnicar and Leisch (2010) and can be summarized in three steps:

*Step 1. Resampling:* Draw bootstrap samples $S_i$ and $T_i$ of size *n* from the data and use the original data, X, as evaluation set $E_i = X$. Apply the clustering method of choice to $S_i$ and $T_i$ and obtain $C^S_i$ and $C^T_i$.

*Step 2. Mapping:* Assign each observation $x_i$ to the closest centers of $C^S_i$ and $C^T_i$ using Euclidean distance, resulting in partitions $C^{XS}_i$ and $C^{XT}_i$, where $C^{XS}_i$ is the partition of the original data, X, predicted from clustering bootstrap sample $S_i$ (same for $T_i$ and $C^{XT}_i$).

*Step 3. Evaluation:* Use the Adjusted Rand Index (ARI, Hubert & Arabie, 1985) or the Measure of Concordance (MOC, Pfitzner 2008) as measure of agreement and stability.

Inspect the distributions of ARI/MOC to assess the global reproducibility of the clustering solutions.

While nboot = 100 is recommended, smaller run numbers could give quite informative results as well, if computation times become too high.

Note that the stability of a clustering solution is assessed, but stability is not the only important validity criterion - clustering solutions obtained by very inflexible clustering methods may be stable but not valid, as discussed in Hennig (2007).

## Value

| | |
|---|---|
| `nclusrange` | An integer or an integer vector with the number of clusters or a range of numbers of clusters |
| `clust1` | Partitions, C^XS_i of the original data, X, predicted from clustering bootstrap sample S_i (see Details) |
| `clust2` | Partitions, C^XT_i of the original data, X, predicted from clustering bootstrap sample T_i (see Details) |
| `index1` | Indices of the original data rows in bootstrap sample S_i |
| `index2` | Indices of the original data rows in bootstrap sample T_i |
| `rand` | Adjusted Rand Index values |
| `moc` | Measure of Concordance values |

## References

Hennig, C. (2007). Cluster-wise assessment of cluster stability. *Computational Statistics and Data Analysis*, *52*, 258-271.

Pfitzner, D., Leibbrandt, R., & Powers, D. (2009). Characterization and evaluation of similarity measures for pairs of clusterings. *Knowledge and Information Systems*, *19*(3), 361-394.

Dolnicar, S., & Leisch, F. (2010). Evaluation of structure and reproducibility of cluster solutions using the bootstrap. *Marketing Letters*, *21*(1), 83-101.

## See Also

[local_bootclus](local_bootclus)

## Examples

```
## 3 bootstrap replicates and nstart = 1 for speed in example,
## use at least 20 replicates for real applications
data(diamond)
boot_mixedRKM = global_bootclus(diamond[,-7], nclusrange = 3:4,
method = "mixedRKM", nboot = 3, nstart = 1, seed = 1234)

boxplot(boot_mixedRKM$rand, xlab = "Number of clusters", ylab =
"adjusted Rand Index")

## 5 bootstrap replicates and nstart = 10 for speed in example,
## use more for real applications
#data(macro)
#boot_RKM = global_bootclus(macro, nclusrange = 2:5,
#method = "RKM", nboot = 5, nstart = 10, seed = 1234)

#boxplot(boot_RKM$rand, xlab = "Number of clusters", ylab =
#"adjusted Rand Index")

## 5 bootstrap replicates and nstart = 1 for speed in example,
## use more for real applications
#data(bribery)
```

```
#boot_cluCA = global_bootclus(bribery, nclusrange = 2:5,
#method = "clusCA", nboot = 5, nstart = 1, seed = 1234)

#boxplot(boot_cluCA$rand, xlab = "Number of clusters", ylab =
#"adjusted Rand Index")
```

---

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 (items with prefix AF), defined as the benign uses of humor to enhance one's relationships with others; self-enhancing (SE), indicating uses of humor to enhance the self; aggressive (AG), the use of humor to enhance the self at the expense of others; self-defeating (SD), 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. The number of respondents is 993.

### Usage

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

### Format

A data frame with 993 observations on 32 Likert-type variables (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

SE2 If I am feeling depressed, I can usually cheer myself up with humor

AG3 If someone makes a mistake, I will often tease them about it

SD4 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

SE6 Even when I'm by myself, I'm often amused by the absurdities of life

AG7 People are never offended or hurt by my sense of humor

SD8 I will often get carried away in putting myself down if it makes my family or friends laugh

AF9 I rarely make other people laugh by telling funny stories about myself

SE10 If I am feeling upset or unhappy I usually try to think of something funny about the situation to make myself feel better

AG11 When telling jokes or saying funny things, I am usually not very concerned about how other people are taking it

SD12 I often try to make people like or accept me more by saying something funny about my own weaknesses, blunders, or faults

AF13  I laugh and joke a lot with my closest friends

SE14  My humorous outlook on life keeps me from getting overly upset or depressed about things

AG15  I do not like it when people use humor as a way of criticizing or putting someone down

SD16  I don't often say funny things to put myself down

AF17  I usually don't like to tell jokes or amuse people

SE18  If I'm by myself and I'm feeling unhappy, I make an effort to think of something funny to cheer myself up

AG19  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

SD20  I often go overboard in putting myself down when I am making jokes or trying to be funny

AF21  I enjoy making people laugh

SE22  If I am feeling sad or upset, I usually lose my sense of humor

AG23  I never participate in laughing at others even if all my friends are doing it

SD24  When I am with friends or family, I often seem to be the one that other people make fun of or joke about

AF25  I don't often joke around with my friends

SE26  It is my experience that thinking about some amusing aspect of a situation is often a very effective way of coping with problems

AG27  If I don't like someone, I often use humor or teasing to put them down

SD28  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

AF29  I usually can't think of witty things to say when I'm with other people

SE30  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

AG31  Even if something is really funny to me, I will not laugh or joke about it if someone will be offended

SD32  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)
```

---

local_bootclus                     *Cluster-wise stability assessment of Joint Dimension Reduction and*
                                   *Clustering methods by bootstrapping.*

---

**Description**

Assessment of the cluster-wise stability of a joint dimension and clustering method. The data is resampled using bootstrapping and the Jaccard similarities of the original clusters to the most similar clusters in the resampled data are computed. The mean over these similarities is used as an index of the stability of a cluster. The method is similar to the one described in Hennig (2007).

**Usage**

```
local_bootclus(data, nclus, ndim = NULL,
method = c("RKM","FKM","mixedRKM","mixedFKM","clusCA","MCAk","iFCB"),
scale = TRUE, center= TRUE, alpha = NULL, nstart=100,
nboot=10, alphak = .5, seed = NULL)
```

**Arguments**

| | |
|---|---|
| data | Continuous, Categorical ot Mixed data set |
| nclus | Number of clusters |
| ndim | Dimensionality of the solution |
| method | Specifies the method. Options are RKM for Reduced K-means, FKM for Factorial K-means, mixedRKM for Mixed Reduced K-means, mixedFKM for Mixed Factorial K-means, MCAk for MCA K-means, iFCB for Iterative Factorial Clustering of Binary variables and clusCA for Cluster Correspondence Analysis. |
| scale | A logical value indicating whether the metric variables should be scaled to have unit variance before the analysis takes place (default = TRUE) |
| center | A logical value indicating whether the metric variables should be shifted to be zero centered (default = TRUE) |
| alpha | Adjusts for the relative importance of (mixed) RKM and FKM in the objective function; alpha = 1 reduces to PCA/PCAMIX, alpha = 0.5 to (mixed) reduced K-means, and alpha = 0 to (mixed) factorial K-means |
| nstart | Number of random starts (default = 100) |
| nboot | Number of bootstrap pairs of partitions |
| 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" |
| seed | An integer that is used as argument by set.seed() for offsetting the random number generator when smartStart = NULL. The default value is NULL. |

**Details**

The algorithm for assessing local cluster stability is similar to that in Hennig (2007) and can be summarized in three steps:

*Step 1. Resampling:* Draw bootstrap samples S_i and T_i of size n from the data and use the original data as evaluation set E_i = X. Apply a joint dimension reduction and clustering method to S_i and T_i and obtain C^S_i and C^T_i.

*Step 2. Mapping*: Assign each observation x_i to the closest centers of C^S_i and C^T_i using Euclidean distance, resulting in partitions C^XS_i and C^XT_i.

*Step 3. Evaluation*: Obtain the maximum Jaccard agreement between each original cluster C_k and each one of the two bootstrap clusters, C_^k'XS_i and C_^k'XT_i as measure of agreement and stability, and take the average of each pair.

Inspect the distributions of the maximum Jaccard coefficients to assess the cluster level (local) stability of the solution.

Here are some guidelines for interpretation. Generally, a valid, stable cluster should yield a mean Jaccard similarity value of 0.75 or more. Between 0.6 and 0.75, clusters may be considered as indicating patterns in the data, but which points exactly should belong to these clusters is highly doubtful. Below average Jaccard values of 0.6, clusters should not be trusted. "Highly stable" clusters should yield average Jaccard similarities of 0.85 and above.

While B = 100 is recommended, smaller run numbers could give quite informative results as well, if computation times become too high.

Note that the stability of a cluster is assessed, but stability is not the only important validity criterion - clusters obtained by very inflexible clustering methods may be stable but not valid, as discussed in Hennig (2007).

**Value**

| | |
|---|---|
| nclus | An integer with the number of clusters |
| clust1 | Partitions, C^XS_i of the original data, X, predicted from clustering bootstrap sample S_i (see Details) |
| clust2 | Partitions, C^XT_i of the original data, X, predicted from clustering bootstrap sample T_i (see Details) |
| index1 | Indices of the original data rows in bootstrap sample S_i |
| index2 | Indices of the original data rows in bootstrap sample T_i |
| Jaccard | Mean Jaccard similarity values |

**References**

Hennig, C. (2007). Cluster-wise assessment of cluster stability. *Computational Statistics and Data Analysis*, *52*, 258-271.

**See Also**

[global_bootclus](global_bootclus)

## Examples

```
## 5 bootstrap replicates and nstart = 10 for speed in example,
## use more for real applications
data(iris)
bootres = local_bootclus(iris[,-5], nclus = 3, ndim = 2,
method = "RKM", nboot = 5, nstart = 1, seed = 1234)

boxplot(bootres$Jaccard, xlab = "cluster number", ylab =
"Jaccard similarity")

## 5 bootstrap replicates and nstart = 5 for speed in example,
## use more for real applications
#data(diamond)
#bootres = local_bootclus(diamond[,-7], nclus = 4, ndim = 3,
#method = "mixedRKM", nboot = 5, nstart = 10, seed = 1234)

#boxplot(bootres$Jaccard, xlab = "cluster number", ylab =
#"Jaccard similarity")

## 5 bootstrap replicates and nstart = 1 for speed in example,
## use more for real applications
#data(bribery)
#bootres = local_bootclus(bribery, nclus = 5, ndim = 4,
#method = "clusCA", nboot = 10, nstart = 1, seed = 1234)

#boxplot(bootres$Jaccard, xlab = "cluster number", ylab =
#"Jaccard similarity")
```

---

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.

---

mybond                          *James Bond films*

---

## Description

The data set refers to 26 James Bond films produced up to 2021, based on 10 film characteristics: 7 continuous (year of release, production budget, box office gross in the USA and worldwide, running time, IMDB average rating, Rotten Tomatoes rating) and 3 categorical (Bond actor, native country of the actor playing the villain, native country of the actor playing the Bond girl). All figures in USD are adjusted for inflation. Most of the data was compiled from the Wikipedia page: https://en.wikipedia.org/wiki/List_of_James_Bond_films.

## Usage

```
data("mybond")
```

## Format

A data frame with 26 observations on the following 10 variables.

year  Year of release

budget  Official production budget (in million USD)

grossusa  Box office gross in the USA (in million USD)

grosswrld  Box office gross worldwide (in million USD)

rtime  Running time in minutes

IMDB  IMDB rating

rottentomatoes  Rotten Tomatoes rating

actor  Bond actor

villaincnt  Native country of the actor playing the villain

bondgirlcnt  Native country of the actor playing the Bond girl

## Examples

```
data(mybond)
```

---

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, objlabs = FALSE, attlabs = NULL,
subplot = FALSE, max.overlaps=10, ...)
```

### 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 |
| objlabs | A logical value indicating whether object labels will be plotted; if TRUE row names of the data matrix are used (default = FALSE). Warning: when TRUE, execution time of the plotting function will increase dramatically as the number of objects gets larger |
| 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 |
| max.overlaps | Maximum number of text labels allowed to overlap. Defaults to 10 |
| ... | 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, plot.cluspcamix

## Examples

```
data("mybond")
#Cluster Correspondence Analysis with 3 clusters in 2 dimensions after 10 random starts
outclusCA = clusmca(mybond[,8:10], 3, 2, nstart = 100, seed = 234)
#Save the ggplot2 scatterplot
map = plot(outclusCA, max.overlaps = 40)$map
#Customization (adding titles)
map + ggtitle(paste("Cluster CA plot of the James bond categorical data: 3 clusters of sizes ",
                    paste(outclusCA$size, collapse = ", "),sep = "")) +
    xlab("Dim. 1") + ylab("Dim. 2") +
    theme(plot.title = element_text(size = 10, face = "bold", hjust = 0.5))

data("mybond")
#i-FCB with 3 clusters in 2 dimensions after 10 random starts
outclusCA = clusmca(mybond[,8:10], 3, 2, method = "iFCB", nstart= 10)
#Scatterlot with the observations only (dimensions 1 and 2)
#and cluster description plots showing the 20 largest std. residuals
#(with the full distribution showing in subplots)
plot(outclusCA, dim = c(1,2), 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), attlabs, max.overlaps=10, ...)
```

## 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) |
| attlabs | Vector of custom attribute labels; if not provided, default labeling is applied |
| max.overlaps | Maximum number of text labels allowed to overlap. Defaults to 10 |
| ... | 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](), [plot.cluspcamix]()

## 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$cluster,iris[,5])
#Save the ggplot2 scatterplot
map = plot(outclusPCA)$map
#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))
```

---

plot.cluspcamix            *Plotting function for* cluspcamix() *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, for metric variables, it returns a parallel coordinate plot showing cluster means and for categorical variables, a series of barplots showing the standardized residuals per attribute for each cluster.

## Usage

```
## S3 method for class 'cluspcamix'
plot(x, dims = c(1, 2), cludesc = FALSE,
topstdres = 20, objlabs = FALSE, attlabs = NULL, attcatlabs = NULL,
subplot = FALSE, what = c(TRUE,TRUE), max.overlaps = 10, ...)
```

## Arguments

| | |
|---|---|
| x | Object returned by cluspcamix() |
| 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) |
| topstdres | Number of largest standardized residuals used to describe each cluster (default = 20). Works only in combination with cludesc = TRUE |

| 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 |
|---|---|
| objlabs | A logical value indicating whether object labels will be plotted; if TRUE row names of the data matrix are used (default = FALSE). Warning: when TRUE, execution time of the plotting function will increase dramatically as the number of objects gets larger |
| attlabs | Vector of custom labels of continuous attributes; if not provided, default labeling is applied |
| attcatlabs | Vector of custom labels of categorical attributes (categories); if not provided, default labeling is applied |
| max.overlaps | Maximum number of text labels allowed to overlap. Defaults to 10 |
| ... | Further arguments to be transferred to cluspcamix() |

## Value

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

## References

van de Velden, M., Iodice D'Enza, A., & Markos, A. (2019). Distance-based clustering of mixed data. *Wiley Interdisciplinary Reviews: Computational Statistics*, e1456.

Vichi, M., Vicari, D., & Kiers, H. A. L. (2019). Clustering and dimension reduction for mixed variables. *Behaviormetrika*. doi:10.1007/s41237-018-0068-6.

## See Also

[plot.clusmca](), [plot.cluspca]()

## Examples

```
data(diamond)
#Mixed Reduced K-means solution with 3 clusters in 2 dimensions
#after 10 random starts
outmixedRKM = cluspcamix(diamond, 3, 2, method = "mixedRKM", nstart = 10)
#Scatterplot (dimensions 1 and 2)
plot(outmixedRKM, cludesc = TRUE)
```

---

| 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","mixedRKM","mixedFKM","clusCA","iFCB","MCAk"),
criterion = "asw", dst = "full", alpha = NULL, alphak = NULL,
center = TRUE, scale = TRUE, rotation = "none", nstart = 100,
smartStart = NULL, seed = NULL)

## 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, Categorical ot Mixed data set |
| 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, mixedRKM for mixed reduced K-means, mixedFKM for mixed 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 (mixed) RKM and FKM in the objective function; alpha = 1 reduces to PCA, alpha = 0.5 to (mixed) reduced K-means, and alpha = 0 to (mixed) 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 set.seed() for offsetting the random number generator when smartStart = NULL. The default value is NULL. |
| 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 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

| clusobjbest | The output of the optimal run of cluspca() or clusmca() |
|---|---|
| nclusbest | The optimal number of clusters |
| ndimbest | The optimal number of dimensions |
| critbest | The optimal criterion value for nclusbest clusters and ndimbest dimensions |
| critgrid | Matrix of size nclusrange x ndimrange 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) |

| criterion | "asw" for average Silhouette width or "ch" for "Calinski-Harabasz" |
|-----------|--------------------------------------------------------------------|
| cluasw    | Average Silhouette width values of each cluster, when criterion = "asw" |

### 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

global_bootclus, local_bootclus

### 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
# nstart = 1 for speed in example
# use more for real applications
bestRKM = tuneclus(macro, 3:4, 2:3, method = ”RKM”,
criterion = ”asw”, dst = ”low”, nstart = 1, seed = 1234)
bestRKM
#plot(bestRKM)

# Cluster Correspondence Analysis for a range of clusters and dimensions
data(bribery)
# Cluster quality assessment based on the Callinski-Harabasz index in the full dimensional space
bestclusCA = tuneclus(bribery, 4:5, 3:4, method = ”clusCA”,
criterion = ”ch”, nstart = 20, seed = 1234)
bestclusCA
#plot(bestclusCA, cludesc = TRUE)

# Mixed reduced K-means for a range of clusters and dimensions
data(diamond)
# Cluster quality assessment based on the average silhouette width in the low dimensional space
# nstart = 5 for speed in example
# use more for real applications
bestmixedRKM = tuneclus(diamond[,-7], 3:4, 2:3,
method = ”mixedRKM”, criterion = ”asw”, dst = ”low”,
nstart = 5, seed = 1234)
bestmixedRKM
#plot(bestmixedRKM)
```

# Index