

Package ‘FactoMineR’

January 27, 2012

Version 1.18

Date 2012-01-27

Title Multivariate Exploratory Data Analysis and Data Mining with R

Author Francois Husson, Julie Josse, Sebastien Le, Jeremy Mazet

Maintainer Francois Husson <francois.husson@agrocampus-ouest.fr>

Depends ellipse,lattice,cluster,scatterplot3d

Suggests missMDA

Description an R package for exploratory data analysis

License GPL (>= 2)

URL <http://factominer.free.fr>, <http://www.agrocampus-ouest.fr/math/>

Encoding latin1

Repository CRAN

Date/Publication 2012-01-27 15:13:09

R topics documented:

AFDM	3
CA	4
catdes	5
children	6
coeffRV	7
condes	8
coord.ellipse	9
decathlon	10
dimdesc	11
DMFA	12
estim_ncp	13

GPA	14
graph.var	16
HCPC	17
HMFA	19
JO	20
MCA	21
MFA	23
mortality	25
PCA	26
plot.AFDM	28
plot.CA	29
plot.catdes	30
plot.DMFA	31
plot.GPA	32
plot.HCPC	33
plot.HMFA	35
plot.MCA	36
plot.MFA	38
plot.PCA	40
plotellipses	41
plotGPApartial	43
plotMFApartial	44
poison	46
poison.text	46
poulet	47
prefpls	47
print.AFDM	48
print.CA	49
print.GPA	50
print.HMFA	50
print.MCA	51
print.MFA	52
print.PCA	52
reconst	53
simule	54
svd.triplet	55
tab.disjonctif	56
tab.disjonctif.prop	56
tea	57
textual	57
wine	58
write.infile	59

Description

Performs Multiple Factor Analysis with both quantitative and categorical data.

Usage

```
AFDM (base, ncp = 5, graph = TRUE, sup.var = NULL,
      ind.sup = NULL, axes = c(1,2), row.w = NULL)
```

Arguments

base	a data frame with n rows (individuals) and p columns
ncp	number of dimensions kept in the results (by default 5)
graph	boolean, if TRUE a graph is displayed
ind.sup	a vector indicating the indexes of the supplementary individuals
sup.var	a vector indicating the indexes of the supplementary variables
axes	a length 2 vector specifying the components to plot
row.w	an optional row weights (by default, uniform row weights)

Value

Returns a list including:

eig	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
group	a matrix with the variables (continuous and categorical) considered as a group
ind	a list of matrices with all the results for the individuals (coordinates, square cosine, contributions)
quali.var	a list of matrices with all the results for the categorical variables (coordinates, square cosine, contributions, v.test)
quanti.var	a list of matrices with all the results for the quantitative variables (coordinates, correlation, square cosine, contributions)
call	a list with some statistics

Returns the individuals factor map.

Author(s)

Francois Husson <Francois.Husson@agroparcampus-ouest.fr>, Jeremy Mazet

References

Pages J. (2004). Analyse factorielle de donnees mixtes. *Revue Statistique Appliquee*. LII (4). pp. 93-111.

See Also

[print.AFDM](#), [plot.AFDM](#)

Examples

```
## Not run:
data(wine)
res.afdm = AFDM(wine[,c(1,2,30,31)])

## End(Not run)
```

CA	<i>Correspondence Analysis (CA)</i>
----	-------------------------------------

Description

Performs Correspondence Analysis (CA) including supplementary row and/or column points.

Usage

```
CA(X, ncp = 5, row.sup = NULL, col.sup = NULL, graph = TRUE,
   axes = c(1,2), row.w = NULL)
```

Arguments

X	a data frame with n rows and p columns, i.e. a contingency table
ncp	number of dimensions kept in the results (by default 5)
row.sup	a vector indicating the indexes of the supplementary rows
col.sup	a vector indicating the indexes of the supplementary columns
graph	boolean, if TRUE a graph is displayed
axes	a length 2 vector specifying the components to plot
row.w	an optional row weights (by default, uniform row weights)

Value

Returns a list including:

eig	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
col	a list of matrices with all the results for the column variable (coordinates, square cosine, contributions, inertia)

row a list of matrices with all the results for the row variable (coordinates, square cosine, contributions, inertia)

col.sup a list of matrices containing all the results for the supplementary column points (coordinates, square cosine)

row.sup a list of matrices containing all the results for the supplementary row points (coordinates, square cosine)

call a list with some statistics

Returns the row and column points factor map.

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

References

Benzecri, J.-P. (1992) *Correspondence Analysis Handbook*, New-York : Dekker

Benzecri, J.-P. (1980) *L'analyse des données tome 2 : l'analyse des correspondances*, Paris : Bordas

Greenacre, M.J. (1993) *Correspondence Analysis in Practice*, London : Academic Press

Husson, F., Le, S. and Pages, J. (2010). *Exploratory Multivariate Analysis by Example Using R, Chapman and Hall*.

See Also

[print.CA](#), [plot.CA](#), [dimdesc](#)

Examples

```
data(children)
res.ca <- CA (children, col.sup = 6:8, row.sup = 15:18)
```

catdes	<i>Categories description</i>
--------	-------------------------------

Description

Description of the categories of one factor by categorical variables and/or by quantitative variables

Usage

```
catdes(donnee, num.var, proba = 0.05)
```

Arguments

donnee a data frame made up of at least one categorical variables and a set of quantitative variables and/or categorical variables

num.var the indice of the variable to characterized

proba the significance threshold considered to characterized the category (by default 0.05)

Value

Returns a list including:

<code>test.chi</code>	The categorical variables which characterized the factor are listed in ascending order (from the one which characterized the most the factor to the one which significantly characterized with the proba proba
<code>category</code>	description of each category of the num.var by each category of all the categorical variables
<code>quanti</code>	the description of each category of the num.var variable by the quantitative variables.

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

References

Husson, F., Le, S. and Pages, J. (2010). Exploratory Multivariate Analysis by Example Using R, *Chapman and Hall*. Lebart, L., Morineau, A. and Piron, M. (1995) Statistique exploratoire multidimensionnelle, *Dunod*.

See Also

[plot.catdes,condes](#)

Examples

```
data(wine)
catdes(wine, num.var=2)
```

children	<i>Children (data)</i>
----------	------------------------

Description

The data used here is a contingency table that summarizes the answers given by different categories of people to the following question : according to you, what are the reasons that can make hesitate a woman or a couple to have children?

Usage

```
data(children)
```

Format

A data frame with 18 rows and 8 columns. Rows represent the different reasons mentioned, columns represent the different categories (education, age) people belong to.

Source

Traitements Statistiques des Enquêtes (D. Grangé, L. Lebart, eds.) Dunod, 1993

Examples

```
data(children)
res.ca <- CA(children, col.sup = 6:8, row.sup = 15:18)
```

`coeffRV`*Calculate the RV coefficient and test its significance*

Description

Calculate the RV coefficient and test its significance.

Usage

```
coeffRV(X, Y)
```

Arguments

X a matrix with n rows (individuals) and p numerous columns (variables)
Y a matrix with n rows (individuals) and p numerous columns (variables)

Details

Calculates the RV coefficient between X and Y . It returns also the standardized RV, the expectation, the variance and the skewness under the permutation distribution. These moments are used to approximate the exact distribution of the RV statistic with the Pearson type III approximation and the p-value associated to this test is given.

Value

A list containing the following components:

RV	the RV coefficient between the two matrices
RVs	the standardized RV coefficients
mean	the mean of the RV permutation distribution
variance	the variance of the RV permutation distribution
skewness	the skewness of the RV permutation distribution
p.value	the p-value associated to the test of the significativity of the RV coefficient (with the Pearson type III approximation)

Author(s)

Julie Josse, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

References

- Escouffier, Y. (1973) *Le traitement des variables vectorielles*. *Biometrics* **29** 751–760.
- Josse, J., Husson, F., Pagès, J. (2007) *Testing the significance of the RV coefficient*. Aveiro, Portugal
- Kazi-Aoual, F., Hitier, S., Sabatier, R., Lebreton, J.-D., (1995) Refined approximations to permutations tests for multivariate inference. *Computational Statistics and Data Analysis*, **20**, 643–656

Examples

```
data(wine)
X <- wine[,3:7]
Y <- wine[,11:20]
coeffRV(X,Y)
```

condes

Continuous variable description

Description

Description continuous by quantitative variables and/or by categorical variables

Usage

```
condes(donnee, num.var, proba = 0.05)
```

Arguments

donnee	a data frame made up of at least one quantitative variable and a set of quantitative variables and/or categorical variables
num.var	the number of the variable to characterized
proba	the significance threshold considered to characterized the category (by default 0.05)

Value

Returns a list including:

quanti	the description of the num.var variable by the quantitative variables. The variables are sorted in ascending order (from the one which characterized the most to the one which significantly characterized with the proba proba)
quali	The categorical variables which characterized the continuous variables are listed in ascending order
category	description of the continuous variable num.var by each category of all the categorical variables

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[catdes](#)

Examples

```
data(decathlon)
condes(decathlon, num.var=3)
```

coord.ellipse	<i>Construct confidence ellipses</i>
---------------	--------------------------------------

Description

Construct confidence ellipses

Usage

```
coord.ellipse (coord.simul, centre = NULL, axes = c(1, 2),
              level.conf = 0.95, npoint = 100, bary = FALSE)
```

Arguments

coord.simul	a data frame containing the coordinates of the individuals for which the confidence ellipses are constructed. This data frame can contain more than 2 variables; the variables taken into account are chosen after. The first column must be a factor which allows to associate one row to an ellipse. The simule object of the result of the simule function correspond to a data frame.
centre	a data frame whose columns are the same than those of the coord.simul, and with the coordinates of the centre of each ellipse. This parameter is optional and NULL by default; in this case, the centre of the ellipses is calculated from the data
axes	a length 2 vector specifying the components of coord.simul that are taken into account
level.conf	confidence level used to construct the ellipses. By default, 0.95
npoint	number of points used to draw the ellipses
bary	boolean, if bary = TRUE, the coordinates of the ellipse around the barycentre of individuals are calculated

Value

res	a data frame with (npoint times the number of ellipses) rows and three columns. The first column is the factor of coord.simul, the two others columns give the coordinates of the ellipses on the two dimensions chosen.
call	the parameters of the function chosen

Author(s)

Jeremy Mazet

See Also

[simule](#)

decathlon

Performance in decathlon (data)

Description

The data used here refer to athletes' performance during two sporting events.

Usage

```
data(decathlon)
```

Format

A data frame with 41 rows and 13 columns: the first ten columns corresponds to the performance of the athletes for the 10 events of the decathlon. The columns 11 and 12 correspond respectively to the rank and the points obtained. The last column is a categorical variable corresponding to the sporting event (2004 Olympic Game or 2004 Decastar)

Source

Département de mathématiques appliquées, Agrocampus Rennes

Examples

```
data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup=13)
```

dimdesc	<i>Dimension description</i>
---------	------------------------------

Description

This function is designed to point out the variables and the categories that are the most characteristic according to each dimension obtained by a Factor Analysis.

Usage

```
dimdesc(res, axes = 1:3, proba = 0.05)
```

Arguments

res	an object of class PCA, MCA, CA, MFA or HMFA
axes	a vector with the dimensions to describe
proba	the significance threshold considered to characterized the dimension (by default 0.05)

Value

Returns a list including:

quanti	the description of the dimensions by the quantitative variables. The variables are sorted.
quali	the description of the dimensions by the categorical variables

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

References

Husson, F., Le, S. and Pages, J. (2010). Exploratory Multivariate Analysis by Example Using R, *Chapman and Hall*.

See Also

[PCA](#), [CA](#), [MCA](#), [MFA](#), [HMFA](#)

Examples

```
data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup=13, graph=FALSE)
dimdesc(res.pca)
```

DMFA

*Dual Multiple Factor Analysis (DMFA)***Description**

Performs Dual Multiple Factor Analysis (DMFA) with supplementary individuals, supplementary quantitative variables and supplementary categorical variables.

Usage

```
DMFA(don, num.fact = ncol(data), scale.unit = TRUE, ncp = 5,
     quanti.sup = NULL, quali.sup = NULL, graph = TRUE, axes=c(1,2))
```

Arguments

don	a data frame with n rows (individuals) and p columns (numeric variables)
num.fact	the number of the categorical variable which allows to make the group of individuals
scale.unit	a boolean, if TRUE (value set by default) then data are scaled to unit variance
ncp	number of dimensions kept in the results (by default 5)
quanti.sup	a vector indicating the indexes of the quantitative supplementary variables
quali.sup	a vector indicating the indexes of the categorical supplementary variables
graph	boolean, if TRUE a graph is displayed
axes	a length 2 vector specifying the components to plot

Value

Returns a list including:

eig	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
var	a list of matrices containing all the results for the active variables (coordinates, correlation between variables and axes, square cosine, contributions)
ind	a list of matrices containing all the results for the active individuals (coordinates, square cosine, contributions)
ind.sup	a list of matrices containing all the results for the supplementary individuals (coordinates, square cosine)
quanti.sup	a list of matrices containing all the results for the supplementary quantitative variables (coordinates, correlation between variables and axes)
quali.sup	a list of matrices containing all the results for the supplementary categorical variables (coordinates of each categories of each variables, and v.test which is a criterion with a Normal distribution)
svd	the result of the singular value decomposition

`var.partiel` a list with the partial coordinate of the variables for each group
`cor.dim.gr`
`Xc` a list with the data centered by group
`group` a list with the results for the groups (cordinate, normalized coordinates, cos2)
`Cov` a list with the covariance matrices for each group

Returns the individuals factor map and the variables factor map.

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[plot.DMFA](#), [dimdesc](#)

Examples

```
## Example with the famous Fisher's iris data
res.dmfa = DMFA ( iris, num.fact = 5)
```

estim_ncp

Estimate the number of components in Principal Component Analysis

Description

Estimate the number of components in PCA .

Usage

```
estim_ncp(X, ncp.min=0, ncp.max=NULL, scale=TRUE, method="GCV")
```

Arguments

`X` a data frame with continuous variables
`ncp.min` minimum number of dimensions to interpret, by default 0
`ncp.max` maximum number of dimensions to interpret, by default NULL which corresponds to the number of columns minus 2
`scale` a boolean, if TRUE (value set by default) then data are scaled to unit variance
`method` method used to estimate the number of components, "GCV" for the generalized cross-validation approximation or "Smooth" for the smoothing method (by default "GCV")

Value

Returns `ncp` the best number of dimensions to use (find the minimum or the first local minimum) and the mean error for each dimension tested

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>, Julie Josse <Julie.Josse@agrocampus-ouest.fr>

References

Josse, J. and Husson, F. (2010). Selecting the number of components in PCA using cross-validation approximations

See Also

[PCA](#)

Examples

```
data(decathlon)
nb.dim <- estim_ncp(decathlon[,1:10],scale=TRUE)
```

GPA

Generalised Procrustes Analysis

Description

Performs Generalised Procrustes Analysis (GPA) that takes into account missing values.

Usage

```
GPA(df, tolerance=10^-10, nbiteration=200, scale=TRUE,
    group, name.group = NULL, graph = TRUE, axes = c(1,2))
```

Arguments

<code>df</code>	a data frame with n rows (individuals) and p columns (quantitative variables)
<code>tolerance</code>	a threshold with respect to which the algorithm stops, i.e. when the difference between the GPA loss function at step n and $n+1$ is less than <code>tolerance</code>
<code>nbiteration</code>	the maximum number of iterations until the algorithm stops
<code>scale</code>	a boolean, if TRUE (which is the default value) scaling is required
<code>group</code>	a vector indicating the number of variables in each group
<code>name.group</code>	a vector indicating the name of the groups (the groups are successively named <code>group.1</code> , <code>group.2</code> and so on, by default)
<code>graph</code>	boolean, if TRUE a graph is displayed
<code>axes</code>	a length 2 vector specifying the components to plot

Details

Performs a Generalised Procrustes Analysis (GPA) that takes into account missing values: some data frames of `df` may have non described or non evaluated rows, i.e. rows with missing values only.

The algorithm used here is the one developed by Commandeur.

Value

A list containing the following components:

RV	a matrix of RV coefficients between partial configurations
RVs	a matrix of standardized RV coefficients between partial configurations
simi	a matrix of Procrustes similarity indexes between partial configurations
scaling	a vector of isotropic scaling factors
dep	an array of initial partial configurations
consensus	a matrix of consensus configuration
Xfin	an array of partial configurations after transformations
correlations	correlation matrix between initial partial configurations and consensus dimensions
PANOVA	a list of "Procrustes Analysis of Variance" tables, per assessor (config), per product(objet), per dimension (dimension)

Author(s)

Elisabeth Morand

References

- Commandeur, J.J.F (1991) *Matching configurations*. DSWO press, Leiden University.
- Dijksterhuis, G. & Punter, P. (1990) Interpreting generalized procrustes analysis "Analysis of Variance" tables, *Food Quality and Preference*, **2**, 255–265
- Gower, J.C (1975) Generalized Procrustes analysis, *Psychometrika*, **40**, 33–50
- Kazi-Aoual, F., Hitier, S., Sabatier, R., Lebreton, J.-D., (1995) Refined approximations to permutations tests for multivariate inference. *Computational Statistics and Data Analysis*, **20**, 643–656
- Qannari, E.M., MacFie, H.J.H, Courcoux, P. (1999) Performance indices and isotropic scaling factors in sensory profiling, *Food Quality and Preference*, **10**, 17–21

Examples

```
## Not run:
data(wine)
res.gpa <- GPA(wine[,-(1:2)], group=c(5,3,10,9,2),
              name.group=c("olf", "vis", "olfag", "gust", "ens"))

### If you want to construct the partial points for some individuals only
plotGPApartial (res.gpa)

## End(Not run)
```

graph.var

Make graph of variables

Description

Plot the graphs of the variables after a Factor Analysis.

Usage

```
graph.var(x, axes = c(1, 2),
         xlim = NULL, ylim = NULL, col.sup = "blue",
         col.var = "black", draw="all", label=draw, lim.cos2.var = 0.1,
         cex = 1, title = NULL, new.plot = TRUE, ...)
```

Arguments

x	an object of class PCA, MCA, MFA or HMFA
axes	a length 2 vector specifying the components to plot
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
col.sup	a color for the quantitative supplementary variables
col.var	a color for the variables
draw	a list of character for the variables which are drawn (by default, all the variables are drawn). You can draw all the active variables by putting "var" and/or all the supplementary variables by putting "quanti.sup" and/or a list with the names of the variables which should be drawn
label	a list of character for the variables which are labelled (by default, all the drawn variables are labelled). You can label all the active variables by putting "var" and/or all the supplementary variables by putting "quanti.sup" and/or a list with the names of the variables which should be labelled
lim.cos2.var	value of the square cosinus under the variables are not drawn
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
new.plot	boolean, if TRUE, a new graphical device is created
...	further arguments passed to or from other methods

Value

Returns the variables factor map.

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[PCA](#), [MFA](#), [MCA](#), [DMFA](#), [HMFA](#)

Examples

```
data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup = 13, graph = FALSE)
graph.var (res.pca, draw = c("var", "Points"),
          label = c("Long.jump", "Points"))
```

 HCPC

Hierarchical Clustering on Principle Components (HCPC)

Description

Performs an agglomerative hierarchical clustering on results from a factor analysis. It is possible to cut the tree by clicking at the suggested (or an other) level. Results include paragons, description of the clusters, graphics.

Usage

```
HCPC(res, nb.clust=0, consol=TRUE, iter.max=10, min=3,
      max=NULL, metric="euclidean", method="ward", order=TRUE,
      graph.scale="inertia", nb.par=5, graph=TRUE, proba=0.05,
      cluster.CA="rows",...)
```

Arguments

<code>res</code>	Either the result of a factor analysis, a dataframe, or a vector.
<code>nb.clust</code>	an integer. If 0, the tree is cut at the level the user clicks on. If -1, the tree is automatically cut at the suggested level (see details). If a (positive) integer, the tree is cut with <code>nb.clusters</code> clusters.
<code>consol</code>	a boolean. If TRUE, a k-means consolidation is performed.
<code>iter.max</code>	An integer. The maximum number of iterations for the consolidation.
<code>min</code>	an integer. The least possible number of clusters suggested.
<code>max</code>	an integer. The higher possible number of clusters suggested; by default the minimum between 10 and the number of individuals divided by 2.
<code>metric</code>	The metric used to built the tree. See agnes for details.
<code>method</code>	The method used to built the tree. See agnes for details.
<code>order</code>	A boolean. If TRUE, clusters are ordered following their center coordinate on the first axis.

<code>graph.scale</code>	A character string. By default "inertia" and the height of the tree corresponds to the inertia gain, else "sqrt-inertia" the square root of the inertia gain.
<code>nb.par</code>	An integer. The number of edited paragons.
<code>graph</code>	If TRUE, graphics are displayed. If FALSE, no graph are displayed.
<code>proba</code>	The probability used to select axes and variables in <code>catdes</code> (see catdes for details).
<code>cluster.CA</code>	A string equals to "rows" or "columns" for the clustering of Correspondence Analysis results.
<code>...</code>	Other arguments from other methods.

Details

The function first built the tree with `agnes`. Then the sum of the intra-cluster inertia are calculated for each partition. The suggested partition is the one with the higher relative loss of inertia ($i(\text{clusters } n+1)/i(\text{cluster } n)$).

The absolute loss of inertia ($i(\text{cluster } n)-i(\text{cluster } n+1)$) is plotted with the tree.

Value

Returns a list including:

<code>data.clust</code>	The original data with a supplementary row called <code>class</code> containing the partition.
<code>desc.axes</code>	The description of the classes by the factors (axes). See catdes for details.
<code>desc.fact</code>	The description of the classes by the variables. See catdes for details.
<code>call</code>	A list of parameters and internal objects.
<code>ind.desc</code>	The paragons (<code>para</code>) and the more typical individuals of each cluster. See details.

Returns the tree and a barplot of the inertia gains, the individual factor map with the tree (3D), the factor map with individuals colored by cluster (2D).

Author(s)

Guillaume Le Ray, Quentin Molto, Francois Husson <husson@agrocampus-ouest.fr>

See Also

[plot.HCPC](#), [catdes](#)

Examples

```
## Not run:
data(iris)
# Principal Component Analysis:
res.pca <- PCA(iris[,1:4], ncp=10, graph=FALSE)
# Clustering, auto nb of clusters:
res.hcpc=HCPC(res.pca, nb.clust=-1)

## End(Not run)
```

Description

Performs a hierarchical multiple factor analysis, using an object of class `list of data.frame`.

Usage

```
HMFA(X,H,type = rep("s", length(H[[1]])), ncp = 5, graph = TRUE,
      axes = c(1,2), name.group = NULL)
```

Arguments

<code>X</code>	a <code>data.frame</code>
<code>H</code>	a list with one vector for each hierarchical level; in each vector the number of variables or the number of group constituting the group
<code>type</code>	the type of variables in each group in the first partition; three possibilities: "c" or "s" for quantitative variables (the difference is that for "s", the variables are scaled in the program), "n" for categorical variables; by default, all the variables are quantitative and the variables are scaled unit
<code>ncp</code>	number of dimensions kept in the results (by default 5)
<code>graph</code>	boolean, if TRUE a graph is displayed
<code>axes</code>	a length 2 vector specifying the components to plot
<code>name.group</code>	a list of vector containing the name of the groups for each level of the hierarchy (by default, NULL and the group are named L1.G1, L1.G2 and so on)

Value

Returns a list including:

<code>eig</code>	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
<code>group</code>	a list of matrices with all the results for the groups (Lg and RV coefficients, coordinates, square cosine, contributions, distance to the origin, the correlations between each group and each factor)
<code>ind</code>	a list of matrices with all the results for the active individuals (coordinates, square cosine, contributions)
<code>quanti.var</code>	a list of matrices with all the results for the quantitative variables (coordinates, correlation between variables and axes)
<code>quali.var</code>	a list of matrices with all the results for the supplementary categorical variables (coordinates of each categories of each variables, and <code>v.test</code> which is a criterion with a Normal distribution)
<code>partial</code>	a list of arrays with the coordinates of the partial points for each partition

Author(s)

Sebastien Le, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

References

Le Dien, S. & Pagès, J. (2003) Hierarchical Multiple factor analysis: application to the comparison of sensory profiles, *Food Quality and Preferences*, **18 (6)**, 453-464.

See Also

[print.HMFA](#), [plot.HMFA](#), [dimdesc](#)

Examples

```
data(wine)
hierar <- list(c(2,5,3,10,9,2), c(4,2))
res.hmfa <- HMFA(wine, H = hierar, type=c("n",rep("s",5)))
```

JO

Number of medals in athletics during olympic games per country

Description

This data frame is a contingency table with the athletics events (in row) and the countries (in columns). Each cell gives the number of medals obtained during the 5 olympic games from 1992 to 2008 (Barcelona 1992, Atlanta 1996, Sydney 2000, Athens 2004, Beijing 2008).

Usage

```
data(JO)
```

Format

A data frame with the 24 events in athletics and in column the 58 countries who obtained at least one medal

Examples

```
## Not run:
data(JO)
res.ca <- CA(JO)
res.ca <- CA(JO, axes = 3:4)

## End(Not run)
```

Description

Performs Multiple Correspondence Analysis (MCA) with supplementary individuals, supplementary quantitative variables and supplementary categorical variables.

Missing values are treated as an additional level, categories which are rare can be ventilated

Usage

```
MCA(X, ncp = 5, ind.sup = NULL, quanti.sup = NULL,
     quali.sup = NULL, graph = TRUE, level.ventil = 0,
     axes = c(1,2), row.w = NULL, na.method="NA", tab.disj=NULL)
```

Arguments

<code>X</code>	a data frame with n rows (individuals) and p columns (categorical variables)
<code>ncp</code>	number of dimensions kept in the results (by default 5)
<code>ind.sup</code>	a vector indicating the indexes of the supplementary individuals
<code>quanti.sup</code>	a vector indicating the indexes of the quantitative supplementary variables
<code>quali.sup</code>	a vector indicating the indexes of the categorical supplementary variables
<code>graph</code>	boolean, if TRUE a graph is displayed
<code>level.ventil</code>	a number corresponding to the level under which the category is ventilated; by default, 0 and no ventilation is done
<code>axes</code>	a length 2 vector specifying the components to plot
<code>row.w</code>	an optional row weights (by default, a vector of 1 for uniform row weights)
<code>na.method</code>	a string corresponding to the name of the method used if there are missing values; available methods are "NA" or "Average" (by default, "NA")
<code>tab.disj</code>	optional data.frame corresponding to the disjunctive table used for the analysis; it corresponds to a disjunctive table obtained from imputation method (see package missMDA).

Value

Returns a list including:

<code>eig</code>	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
<code>var</code>	a list of matrices containing all the results for the active variables (coordinates, square cosine, contributions, v.test)
<code>ind</code>	a list of matrices containing all the results for the active individuals (coordinates, square cosine, contributions)

<code>ind.sup</code>	a list of matrices containing all the results for the supplementary individuals (coordinates, square cosine)
<code>quanti.sup</code>	a matrix containing the coordinates of the supplementary quantitative variables (the correlation between a variable and an axis is equal to the variable coordinate on the axis)
<code>quali.sup</code>	a list of matrices with all the results for the supplementary categorical variables (coordinates of each categories of each variables, square cosine and v.test which is a criterion with a Normal distribution)
<code>call</code>	a list with some statistics

Returns the graphs of the individuals and categories and the graph with the variables.

Author(s)

Francois Husson <husson@agrocampus-ouest.fr>, Julie Josse, Jeremy Mazet

References

Husson, F., Le, S. and Pages, J. (2010). *Exploratory Multivariate Analysis by Example Using R, Chapman and Hall.*

See Also

[plotellipses](#), [print.MCA](#), [plot.MCA](#), [dimdesc](#)

Examples

```
## Not run:
data(tea)
res.mca=MCA(tea,quanti.sup=19,quali.sup=20:36)
plot(res.mca,invisible=c("var","quali.sup","quanti.sup"),cex=0.7)
plot(res.mca,invisible=c("ind","quali.sup","quanti.sup"),cex=0.8)
plot(res.mca,invisible=c("quali.sup","quanti.sup"),cex=0.8)
dimdesc(res.mca)
plotellipses(res.mca,keepvar=1:4)
plotellipses(res.mca,keepvar="Tea")

## Example with missing values : use the missMDA package
require(missMDA)
data(vnf.example)
tab.disj.comp <- imputeMCA(vnf.example,ncp=2)
res.mca <- MCA(vnf.example,tab.disj=tab.disj.comp)

## End(Not run)
```

Description

Performs Multiple Factor Analysis in the sense of Escofier-Pages with supplementary individuals and supplementary groups of variables. Groups of variables can be quantitative, categorical or contingency tables.

Missing values in numeric variables are replaced by the column mean. Missing values in categorical variables are treated as an additional level.

Usage

```
MFA (base, group, type = rep("s",length(group)), ind.sup = NULL,
     ncp = 5, name.group = NULL, num.group.sup = NULL,
     graph = TRUE, weight.col.mfa = NULL, row.w = NULL,
     axes = c(1,2), tab.comp=NULL)
```

Arguments

base	a data frame with n rows (individuals) and p columns (variables)
group	a list indicating the number of variables in each group
type	the type of variables in each group; four possibilities: "c" or "s" for quantitative variables (the difference is that for "s" variables are scaled to unit variance), "n" for categorical variables and "f" for frequencies (from a contingency tables); by default, all variables are quantitative and scaled to unit variance
ind.sup	a vector indicating the indexes of the supplementary individuals
ncp	number of dimensions kept in the results (by default 5)
name.group	a vector containing the name of the groups (by default, NULL and the group are named group.1, group.2 and so on)
num.group.sup	the indexes of the illustrative groups (by default, NULL and no group are illustrative)
graph	boolean, if TRUE a graph is displayed
weight.col.mfa	vector of weights, useful for HMFA method (by default, NULL and an MFA is performed)
row.w	an optional row weights (by default, a vector of 1 for uniform row weights)
axes	a length 2 vector specifying the components to plot
tab.comp	object obtained from the imputeMFA function of the missMDA package that allows to handle missing values

Value

<code>summary.quali</code>	a summary of the results for the categorical variables
<code>summary.quant</code>	a summary of the results for the quantitative variables
<code>separate.analyses</code>	the results for the separate analyses
<code>eig</code>	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
<code>group</code>	a list of matrices containing all the results for the groups (Lg and RV coefficients, coordinates, square cosine, contributions, distance to the origin, the correlations between each group and each factor)
<code>rapport.inertie</code>	inertia ratio
<code>ind</code>	a list of matrices containing all the results for the active individuals (coordinates, square cosine, contributions)
<code>ind.sup</code>	a list of matrices containing all the results for the supplementary individuals (coordinates, square cosine)
<code>quant.var</code>	a list of matrices containing all the results for the quantitative variables (coordinates, correlation between variables and axes, contribution, cos2)
<code>quali.var</code>	a list of matrices containing all the results for categorical variables (coordinates of each categories of each variables, contribution and v.test which is a criterion with a Normal distribution)
<code>freq</code>	a list of matrices containing all the results for the frequencies (coordinates, contribution, cos2)
<code>quant.var.sup</code>	a list of matrices containing all the results for the supplementary quantitative variables (coordinates, correlation between variables and axes, cos2)
<code>quali.var.sup</code>	a list of matrices containing all the results for the supplementary categorical variables (coordinates of each categories of each variables, cos2 and v.test which is a criterion with a Normal distribution)
<code>freq.sup</code>	a list of matrices containing all the results for the supplementary frequencies (coordinates, cos2)
<code>partial.axes</code>	a list of matrices containing all the results for the partial axes (coordinates, correlation between variables and axes, correlation between partial axes)
<code>global.pca</code>	the result of the analysis when it is considered as a unique weighted PCA

Returns the individuals factor map, the variables factor map and the groups factor map.

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>, J. Mazet

References

- Escofier, B. and Pages, J. (1994) Multiple Factor Analysis (AFMULT package). *Computational Statistics and Data Analysis*, 18, 121-140.
- Becue-Bertaut, M. and Pages, J. (2008) Multiple factor analysis and clustering of a mixture of quantitative, categorical and frequency data. *Computational Statistics and Data Analysis*, 52, 3255-3268.

See Also

[print.MFA](#), [plot.MFA](#), [dimdesc](#)

Examples

```
data(wine)
res = MFA(wine, group=c(2,5,3,10,9,2), type=c("n",rep("s",5)),
  ncp=5, name.group=c("orig","olf","vis","olfag","gust","ens"),
  num.group.sup=c(1,6))
barplot(res$eig[,1],main="Eigenvalues",names.arg=1:nrow(res$eig))

## Not run:
#### Confidence ellipses around categories per variable
plotellipses(res)
plotellipses(res,keepvar="Label") ## for 1 variable

#### Interactive graph
liste = plotMFApartial(res)
plot(res,choix="ind",habillage = "Terroir")

###Example with groups of categorical variables
data (poison)
MFA(poison, group=c(2,2,5,6), type=c("s","n","n","n"),
  name.group=c("desc","desc2","symptom","eat"),
  num.group.sup=1:2)

###Example with groups of frequency tables
data(mortality)
res<-MFA(mortality,group=c(9,9),type=c("f","f"),
  name.group=c("1979","2006"))

## End(Not run)
```

mortality

The cause of mortality in France in 1979 and 2006

Description

The cause of mortality in France in 1979 and 2006.

Usage

```
data(mortality)
```

Format

A data frame with 62 rows (the different causes of death) and 18 columns. Each column corresponds to an age interval (15-24, 25-34, 35-44, 45-54, 55-64, 65-74, 75-84, 85-94, 95 and more) in a year. The 9 first columns correspond to data in 1979 and the 9 last columns to data in 2006. In each cell, the counts of deaths for a cause of death in an age interval (in a year) is given.

Source

Centre d'épidémiologie sur les causes médicales

Examples

```
data(mortality)

## Not run:
res<-MFA(mortality,group=c(9,9),type=c("f","f"),
         name.group=c("1979","2006"))

plot(res,choix="freq",invisible="ind",habillage="group")
lines(res$freq$coord[1:9,1],mfa$freq$coord[1:9,2],col="red")
lines(res$freq$coord[10:18,1],mfa$freq$coord[10:18,2],col="green")

## End(Not run)
```

 PCA

Principal Component Analysis (PCA)

Description

Performs Principal Component Analysis (PCA) with supplementary individuals, supplementary quantitative variables and supplementary categorical variables.
Missing values are replaced by the column mean.

Usage

```
PCA(X, scale.unit = TRUE, ncp = 5, ind.sup = NULL,
    quanti.sup = NULL, quali.sup = NULL, row.w = NULL,
    col.w = NULL, graph = TRUE, axes = c(1,2))
```

Arguments

<code>X</code>	a data frame with n rows (individuals) and p columns (numeric variables)
<code>ncp</code>	number of dimensions kept in the results (by default 5)
<code>scale.unit</code>	a boolean, if TRUE (value set by default) then data are scaled to unit variance
<code>ind.sup</code>	a vector indicating the indexes of the supplementary individuals
<code>quanti.sup</code>	a vector indicating the indexes of the quantitative supplementary variables
<code>quali.sup</code>	a vector indicating the indexes of the categorical supplementary variables
<code>row.w</code>	an optional row weights (by default, a vector of 1 for uniform row weights)
<code>col.w</code>	an optional column weights (by default, uniform column weights)
<code>graph</code>	boolean, if TRUE a graph is displayed
<code>axes</code>	a length 2 vector specifying the components to plot

Value

Returns a list including:

eig	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
var	a list of matrices containing all the results for the active variables (coordinates, correlation between variables and axes, square cosine, contributions)
ind	a list of matrices containing all the results for the active individuals (coordinates, square cosine, contributions)
ind.sup	a list of matrices containing all the results for the supplementary individuals (coordinates, square cosine)
quanti.sup	a list of matrices containing all the results for the supplementary quantitative variables (coordinates, correlation between variables and axes)
quali.sup	a list of matrices containing all the results for the supplementary categorical variables (coordinates of each categories of each variables, and v.test which is a criterion with a Normal distribution)

Returns the individuals factor map and the variables factor map.

Author(s)

Francois Husson <Francois.Husson@agroparcampus-ouest.fr>, Jeremy Mazet

References

Husson, F., Le, S. and Pages, J. (2010). Exploratory Multivariate Analysis by Example Using R, *Chapman and Hall*.

See Also

[print.PCA](#), [plot.PCA](#), [dimdesc](#)

Examples

```
data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup=13)
## plot of the eigenvalues
## barplot(res.pca$eig[,1],main="Eigenvalues",names.arg=1:nrow(res.pca$eig))
plot(res.pca,choix="ind",habillage=13)
dimdesc(res.pca, axes = 1:2)
## To draw ellipses around the categories of the 13th variable (which is categorical)
plotellipses(res.pca,13)

## Example with missing data
## use package missMDA
## Not run:
require(missMDA)
data(orange)
nb <- estim_ncpPCA(orange,ncp.min=0,ncp.max=5,method.cv="Kfold",nbsim=50)
```

```

imputed <- imputePCA(orange,ncp=nb$ncp)
res.pca <- PCA(imputed$completeObs)

## End(Not run)

```

plot.AFDM

Draw the Multiple Factor Analysis for Mixt Data graphs

Description

Draw the Multiple Factor Analysis for Mixt Data graphs.

Usage

```

## S3 method for class 'AFDM'
plot(x, choix = "group", axes = c(1, 2), lab.grpe = TRUE,
     lab.var = TRUE, lab.ind = TRUE, habillage = "none", col.lab = FALSE,
     col.hab = NULL, invisible = NULL, lim.cos2.var = 0., xlim = NULL,
     ylim = NULL, cex = 1, title = NULL, palette=NULL, new.plot = FALSE, ...)

```

Arguments

x	an object of class AFDM
choix	a string corresponding to the graph that you want to do ("ind" for the individual or categorical variables graph, "group" for all the variables (quantitative and categorical), "var" for the correlation circle)
axes	a length 2 vector specifying the components to plot
lab.grpe	boolean, if TRUE, the label of the groups are drawn
lab.var	boolean indicating if the labelled of the variables should be drawn on the map
lab.ind	boolean indicating if the labelled of the individuals should be drawn on the map
habillage	string corresponding to the color which are used. If "ind", one color is used for each individual else if it is the name or the position of a categorical variable, it colors according to the different categories of this variable
col.lab	boolean indicating if the labelled should be colored
col.hab	vector indicating the colors to use to labelled the rows or columns elements chosen in habillage
invisible	list of string; for choix ="ind", the individuals can be omit (invisible = "ind"), or supplementary individuals (invisible="ind.sup") or the centerg of gravity of the categorical variables (invisible= "quali"); if invisible = c("ind","ind.sup"), just the centers of gravity are drawn
lim.cos2.var	value of the square cosinus under the variables are not drawn
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'

cex	cf. function <code>par</code> in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : <code>palette=palette(c("black","red","blue"))</code> ; or you can use: <code>palette=palette(rainbow(30))</code> , or in black and white for example: <code>palette=palette(gray(seq(0,.9,len=25)))</code>
new.plot	boolean, if TRUE, a new graphical device is created
...	further arguments passed to or from other methods

Value

Returns the individuals factor map and the variables factor map.

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[AFDM](#)

plot.CA	<i>Draw the Correspondence Analysis (CA) graphs</i>
---------	---

Description

Draw the Correspondence Analysis (CA) graphs.

Usage

```
## S3 method for class 'CA'
plot(x, axes = c(1, 2),
      xlim = NULL, ylim = NULL, invisible = NULL, col.row = "blue",
      col.col = "red", col.row.sup = "darkblue",
      col.col.sup = "darkred", label = "all", cex = 1,
      title = NULL, palette = NULL, ...)
```

Arguments

x	an object of class CA
axes	a length 2 vector specifying the components to plot
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
invisible	string indicating if some points should be unlabelled ("row", "col", "row.sup", "col.sup")

col.row	a color for the rows points
col.col	a color for columns points
col.row.sup	a color for the supplementary rows points
col.col.sup	a color for supplementary columns points
label	a list of character for the elements which are labelled (by default, all the elements are labelled ("row", "row.sup", "col", "col.sup"))
cex	cf. function <code>par</code> in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : <code>palette=palette(c("black","red","blue"))</code> ; or you can use: <code>palette=palette(rainbow(30))</code> , or in black and white for example: <code>palette=palette(gray(seq(0,.9,len=25)))</code>
...	further arguments passed to or from other methods

Value

Returns the individuals factor map and the variables factor map.

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agroparc-ouest.fr>

See Also

[CA](#)

Examples

```
data(children)
res.ca <- CA (children, col.sup = 6:8, row.sup = 15:18)
```

plot.catdes *Plots for description of clusters (catdes)*

Description

Plots a graph from a catdes output.

Usage

```
## S3 method for class 'catdes'
plot(x,col="deepskyblue",show="all",numchar=10,...)
```

Arguments

x	A catdes object, see catdes for details.
col	The color of the bars.
show	a strig. If "quali", only the categorical variables are used. If "quanti", only the quantitative variables are used. If "all", both quali and quanti are used.
numchar	number of characters for the labels
...	further arguments passed to or from other methods

Value

Returns choosen plot.

Author(s)

Guillaume Le Ray, Francois Husson <husson@agrocampus-ouest.fr>

See Also

[catdes](#)

Examples

```
## Not run:  
data(wine)  
res.c=catdes(wine, num.var=2)  
plot.catdes(res.c)  
  
## End(Not run)
```

plot.DMFA

Draw the Duale Multiple Factor Analysis (DMFA) graphs

Description

Plot the graphs for a Principal Component Analysis (DMFA) with supplementary individuals, supplementary quantitative variables and supplementary categorical variables.

Usage

```
## S3 method for class 'DMFA'  
plot(x, axes = c(1, 2), choix = "ind", label="all",  
      lim.cos2.var = 0., xlim=NULL, ylim=NULL, title = NULL,  
      palette = NULL, new.plot = FALSE, ...)
```

Arguments

x	an object of class DMFA
axes	a length 2 vector specifying the components to plot
choix	the graph to plot ("ind" for the individuals, "var" for the variables)
label	a list of character for the elements which are labelled (by default, all the elements are labelled ("ind", ind.sup", "quali", "var", "quanti.sup"))
lim.cos2.var	value of the square cosinus under the variables are not drawn
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : palette=palette(c("black","red","blue")); or you can use: palette=palette(rainbow(30)), or in black and white for example: palette=palette(gray(seq(0,.9,len=25)))
new.plot	boolean, if TRUE, a new graphical device is created
...	further arguments passed to or from other methods

Value

Returns the individuals factor map and the variables factor map, the partial variables representation and the groups factor map.

Author(s)

Francois Husson <husson@agrocampus-ouest.fr>

See Also

[DMFA](#)

plot.GPA

Draw the General Procrustes Analysis (GPA) map

Description

Draw the General Procrustes Analysis (GPA) map.

Usage

```
## S3 method for class 'GPA'
plot(x, axes = c(1, 2),
     lab.ind.moy = TRUE, habillage = "ind",
     partial = "all", chrono = FALSE, xlim = NULL, ylim = NULL,
     cex = 1, title = NULL, palette = NULL, ...)
```

Arguments

x	an object of class GPA
axes	a length 2 vector specifying the components to plot
lab.ind.moy	boolean, if TRUE, the label of the mean points are drawn
habillage	string corresponding to the color which are used. If "ind", one color is used for each individual; if "group" the individuals are colored according to the group
partial	list of the individuals or of the center of gravity for which the partial points should be drawn (by default, partial = "none" and no partial points are drawn)
chrono	boolean, if TRUE, the partial points of a same point are linked (useful when groups correspond to different moment)
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : palette=palette(c("black","red","blue")); or you can use: palette=palette(rainbow(30)), or in black and white for example: palette=palette(gray(seq(0,.9,len=25)))
...	further arguments passed to or from other methods

Value

Returns the General Procrustes Analysis map.

Author(s)

Elisabeth Morand, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[GPA](#)

plot.HCPC

Plots for Hierarchical Classification on Principle Components (HCPC) results

Description

Plots graphs from a HCPC result: tree, barplot of inertia gains and first factor map with or without the tree, in 2 or 3 dimensions.

Usage

```
## S3 method for class 'HCPC'
plot(x, axes=c(1,2), choice="3D.map", rect=TRUE,
     draw.tree=TRUE, ind.names=TRUE, t.level="all", title=NULL,
     new.plot=FALSE, max.plot=15, tree.barplot=TRUE,
     centers.plot=FALSE, ...)
```

Arguments

x	A HCPC object, see HCPC for details.
axes	a two integers vector. Defines the axes of the factor map to plot.
choice	A string. "tree" plots the tree. "bar" plots bars of inertia gains. "map" plots a factor map, individuals colored by cluster. "3D.map" plots the same factor map, individuals colored by cluster, the tree above.
rect	a boolean. If TRUE, rectangles are drawn around clusters if choice ="tree".
tree.barplot	a boolean. If TRUE, the barplot of intra inertia losses is added on the tree graph.
draw.tree	A boolean. If TRUE, the tree is projected on the factor map if choice ="map".
ind.names	A boolean. If TRUE, the individuals names are added on the factor map when choice="3D.map"
t.level	Either a positive integer or a string. A positive integer indicates the starting level to plot the tree on the map when draw.tree=TRUE. If "all", the whole tree is plotted. If "centers", it draws the tree starting t the centers of the clusters.
title	a string. Title of the graph. NULL by default and a title is automatically defined
centers.plot	a boolean. If TRUE, the centers of clusters are drawn on the 3D factor maps.
new.plot	a boolean. If TRUE, the plot is done in a new window.
max.plot	The max for the bar plot
...	Other arguments from other methods.

Value

Returns choosen plot.

Author(s)

Guillaume Le Ray, Quentin Molto, Francois Husson <husson@agrocampus-ouest.fr>

See Also

[HCPC](#)

Examples

```
## Not run:
data(iris)
# Clustering, auto nb of clusters:
res.hcpc=HCPC(iris[1:4], nb.clust=3)
# 3D graph from a different point of view:
plot.HCPC(res.hcpc, choice="3D.map", angle=60)

## End(Not run)
```

plot.HMFA

*Draw the Hierarchical Multiple Factor Analysis (HMFA) graphs***Description**

Draw the Hierarchical Multiple Factor Analysis (HMFA) graphs

Usage

```
## S3 method for class 'HMFA'
plot(x, axes = c(1,2), num=6, choix = "ind",
     lab.grpe = TRUE, lab.var = TRUE, lab.ind.moy = TRUE,
     invisible = NULL, lim.cos2.var = 0.,
     xlim = NULL, ylim = NULL, cex = 1, title = NULL, new.plot = FALSE, ...)
```

Arguments

x	an object of class HMFA
axes	a length 2 vector specifying the components to plot
num	number of graphs in a same windows
choix	a string corresponding to the graph that you want to do ("ind" for the individual or categorical variables graph, "var" for the quantitative variables graph, "axes" for the graph of the partial axes, "group" for the groups representation)
lab.grpe	boolean, if TRUE, the label of the groups are drawn
lab.var	boolean, if TRUE, the label of the variables are drawn
lab.ind.moy	boolean, if TRUE, the label of the mean points are drawn
invisible	list of string; for choix ="ind", the individuals can be omit (invisible = "ind"), or the centers of gravity of the categorical variables (invisible= "quali")
lim.cos2.var	value of the square cosinus under with the points are not drawn
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
new.plot	boolean, if TRUE, a new graphical device is created
...	further arguments passed to or from other methods

Value

Returns the individuals factor map and the variables factor map.

Author(s)

Jérémy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[HMFA](#)

Examples

```
data(wine)
hierar <- list(c(2,5,3,10,9,2), c(4,2))
res.hmfa <- HMFA(wine, H = hierar, type=c("n",rep("s",5)), graph = FALSE)
plot(res.hmfa, invisible="quali")
plot(res.hmfa, invisible="ind")
```

plot.MCA

Draw the Multiple Correspondence Analysis (MCA) graphs

Description

Draw the Multiple Correspondence Analysis (MCA) graphs.

Usage

```
## S3 method for class 'MCA'
plot(x, axes = c(1, 2), choix="ind",
     xlim = NULL, ylim = NULL, invisible = NULL,
     col.ind = "blue", col.var = "red", col.quali.sup = "darkgreen",
     col.ind.sup = "darkblue", col.quanti.sup = "black",
     label = "all", cex = 1, title = NULL, habillage = "none",
     palette = NULL, new.plot = FALSE, ...)
```

Arguments

x	an object of class MCA
axes	a length 2 vector specifying the components to plot
choix	the graph to plot ("ind" for the individuals and the categories, "var" for the variables, "quanti.sup" for the supplementary quantitative variables)
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
invisible	string indicating if some points should not be drawn ("ind", "var", "ind.sup", "quali.sup", "quanti.sup")

col.ind	a color for the individuals, if color ="none" the label is not written
col.var	a color for the categories of categorical variables, if color ="none" the label is not written
col.quali.sup	a color for the categorical supplementary variables, if color ="none" the label is not written
col.ind.sup	a color for the supplementary individuals only if there is not habillage, if color ="none" the label is not written
col.quant.sup	a color for the supplementary quantitative variables, if color ="none" the label is not written
label	print the labels of the points
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
habillage	string corresponding to the color which are used. If "none", one color is used for the individual, another one for the categorical variables; if "quali", one color is used for each categorical variables; else if it is the position of a categorical variable, it colors according to the different categories of this variable
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : palette=palette(c("black","red","blue")); or you can use: palette=palette(rainbow(30)), or in black and white for example: palette=palette(gray(seq(0,.9,len=25)))
new.plot	boolean, if TRUE, a new graphical device is created
...	further arguments passed to or from other methods

Value

Returns the individuals factor map and the variables factor map.

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[MCA](#)

Examples

```
data(poison)
res.mca = MCA(poison, quali.sup = 3:4, quanti.sup = 1:2, graph=FALSE)
plot.MCA(res.mca,invisible=c("var","quali.sup"))
plot.MCA(res.mca,invisible="ind")
plot.MCA(res.mca,choix="var")
```

plot.MFA

*Draw the Multiple Factor Analysis (MFA) graphs***Description**

Draw the Multiple Factor Analysis (MFA) graphs.

Usage

```
## S3 method for class 'MFA'
plot(x, axes = c(1, 2), choix = "ind", ellipse = NULL, ellipse.par = NULL,
     lab.grpe = TRUE, lab.var = TRUE, lab.ind = TRUE,
     lab.par = FALSE, lab.col = TRUE, habillage = "ind", col.hab = NULL,
     invisible = NULL, partial = NULL, lim.cos2.var = 0.,
     chrono = FALSE, xlim = NULL, ylim = NULL,
     cex = 1, title = NULL, palette = NULL, new.plot = FALSE, ...)
```

Arguments

x	an object of class MFA
choix	a string corresponding to the graph that you want to do ("ind" for the individual or categorical variables graph, "var" for the quantitative variables graph, "freq" for the frequency or contingency tables, "axes" for the graph of the partial axes, "group" for the groups representation)
axes	a length 2 vector specifying the components to plot
ellipse	boolean (NULL by default), if not null, draw ellipses around the individuals, and use the results of coord.ellipse
ellipse.par	boolean (NULL by default), if not null, draw ellipses around the partial individuals, and use the results of coord.ellipse
lab.grpe	boolean, if TRUE, the labels of the groups are drawn
lab.var	boolean, if TRUE, the labels of the variables are drawn
lab.ind	boolean, if TRUE, the labels of the mean points are drawn
lab.par	boolean, if TRUE, the labels of the partial points are drawn
lab.col	boolean, if TRUE, the labels of the columns for the contingency tables are drawn
habillage	string corresponding to the color which are used. If "ind", one color is used for each individual; if "group" the individuals are colored according to the group; else if it is the name or the position of a categorical variable, it colors according to the different categories of this variable
col.hab	the colors to use. By default, colors are chosen
invisible	list of string; for choix = "ind", the individuals can be omitted (invisible = "ind"), or supplementary individuals (invisible = "ind.sup") or the center of gravity of the categorical variables (invisible = "quali"); if invisible = c("ind", "ind.sup"), just the centers of gravity are drawn

partial	list of the individuals or of the center of gravity for which the partial points should be drawn (by default, partial = NULL and no partial points are drawn)
lim.cos2.var	value of the square cosinus under which the points are not drawn
chrono	boolean, if TRUE, the partial points of a same point are linked (useful when groups correspond to different moments)
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : palette=palette(c("black","red","blue")); or you can use: palette=palette(rainbow(30)), or in black and white for example: palette=palette(gray(seq(0,.9,len=25)))
new.plot	boolean, if TRUE, a new graphical device is created
...	further arguments passed to or from other methods

Value

Returns the individuals factor map and the variables factor map.

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>, Jeremy Mazet

See Also

[MFA](#)

Examples

```
## Not run:
data(wine)
aa = MFA(wine,group=c(2,5,3,10,9,2),type=c("n",rep("s",5)),ncp=5,
        name.group=c("orig","olf","vis","olfag","gust","ens"),
        num.group.sup=c(1,6),graph=FALSE)
plot(aa, choix = "ind")
plot(aa, choix = "ind", partial="all")
plot(aa, choix = "Terroir")
plot(aa, choix = "var", habillage="group")
plot(aa, choix = "axes")

## End(Not run)
```

plot.PCA

Draw the Principal Component Analysis (PCA) graphs

Description

Plot the graphs for a Principal Component Analysis (PCA) with supplementary individuals, supplementary quantitative variables and supplementary categorical variables.

Usage

```
## S3 method for class 'PCA'
plot(x, axes = c(1, 2), choix = "ind",
     ellipse = NULL, xlim = NULL, ylim = NULL, habillage = "none",
     col.hab = NULL, col.ind = "black", col.ind.sup = "blue",
     col.quali = "magenta", col.quant.sup = "blue",
     col.var = "black", label = "all", invisible = NULL,
     lim.cos2.var = 0., cex = 1, title = NULL, palette=NULL,
     new.plot = FALSE, ...)
```

Arguments

x	an object of class PCA
axes	a length 2 vector specifying the components to plot
choix	the graph to plot ("ind" for the individuals, "var" for the variables)
ellipse	boolean (NULL by default), if not null, draw ellipses around the individuals, and use the results of coord.ellipse
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
habillage	give no color for the individuals ("none"), a color for each individual ("ind"), or color the individuals among a categorical variable (give the number of the categorical variable)
col.hab	a vector with the color to use for the individuals
col.ind	a color for the individuals if there only is not habillage
col.ind.sup	a color for the supplementary individuals only if there is not habillage
col.quali	a color for the categories of categorical variable only if there is not habillage
col.quant.sup	a color for the quantitative supplementary variables
col.var	a color for the variables
label	a list of character for the elements which are labelled (by default, all the elements are labelled ("ind", "ind.sup", "quali", "var", "quant.sup"))
invisible	string indicating if some points should not be drawn ("ind", "ind.sup" or "quali" for the individual graph and "var" or "quant.sup" for the correlation circle graph)

lim.cos2.var	value of the square cosinus under the variables are not drawn
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : palette=palette(c("black","red","blue")); or you can use: palette=palette(rainbow(30)), or in black and white for example: palette=palette(gray(seq(0,.9,len=25)))
new.plot	boolean, if TRUE, a new graphical device is created
...	further arguments passed to or from other methods

Value

Returns the individuals factor map and the variables factor map.

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agroparc-ouest.fr>

See Also

[PCA](#)

Examples

```
data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup = 13)
plot(res.pca, habillage = 13, col.hab=c("green","blue"))
## To automatically draw ellipses around the barycentres of all the categorical variables
plotellipses(res.pca)
## or another graph
aa=cbind.data.frame(decathlon[,13],res.pca$ind$coord)
bb=coord.ellipse(aa,bary=TRUE)
plot.PCA(res.pca,habillage=13,ellipse=bb)
```

plotellipses

Draw confidence ellipses around the categories

Description

Draw confidence ellipses around the categories.

Usage

```
plotellipses(model, keepvar = "all", axis = c(1, 2), means=TRUE, level = 0.95,
  magnify = 2, cex = 0.5, pch = 20, pch.means=15, type = c("g","p"),
  keepnames = TRUE, namescat = NULL, xlim=xlim, ylim=ylim, lwd=1,
  label="all", ...)
```

Arguments

model	an object of class MCA or PCA or MFA
keepvar	a boolean or numeric vector of indexes of variables or a character vector of names of variables. If keepvar is "all", "quali" or "quali.sup" variables which are plotted are all the categorical variables, only those which are used to compute the dimensions (active variables) or only the supplementary categorical variables. If keepvar is a numeric vector of indexes or a character vector of names of variables, only relevant variables are plotted.
axis	a length 2 vector specifying the components to plot
means	boolean which indicates if the confidence ellipses are for (the coordinates of) the means of the categories (the empirical variance is divided by the number of observations) or for (the coordinates of) the observations of the categories
level	the confidence level for the ellipses
magnify	numeric which control how the level names are magnified. A value of 2 means that the level names have character expansion equal to two times cex
cex	cf. function <code>par</code> in the graphics package
pch	plotting character for coordinates, cf. function <code>par</code> in the graphics package
pch.means	plotting character for means, cf. function <code>par</code> in the graphics package
type	cf. function <code>xyplot</code> in the lattice package
keepnames	a boolean or numeric vector of indexes of variables or a character vector of names of variables. If keepnames is TRUE, names of levels are taken from the (modified) dataset extracted from modele, if FALSE trimming names is done. When trimming, names of levels are taken from the (modified) dataset extracted from modele, then, the corresponding number of characters of names of original variables plus 1 is removed. If keepnames is a vector of indexes or names, trimming is done on all variables excepted whose in keepnames
namescat	a vector giving for each observation the value of categorical variable, each variable are stacked under each other. If NULL, names are taken from the (modified) dataset extracted from modele
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
lwd	The line width, a positive number, defaulting to 1
label	a list of character for the elements which are labelled (by default, "all", you can use "none", "ind", ind.sup"))
...	further arguments passed to or from other methods

Value

Return a graph with the ellipses. If only one variable is chosen, the graph is different.

Author(s)

Pierre-Andre Cornillon, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also[MCA,PCA](#)**Examples**

```
## Not run:
data(poison)
res.mca = MCA(poison, quali.sup = 3:4, quanti.sup = 1:2)
plotellipses(res.mca)
plotellipses(res.mca,keepvar=1:4)

## End(Not run)

data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup=13)
plotellipses(res.pca,keepvar=13)
```

plotGPApartial

*Draw an interactive General Procrustes Analysis (GPA) map***Description**

Draw an interactive General Procrustes Analysis (GPA) map. The graph is interactive and clicking on a point will draw the partial points, if you click on a point for which the partial points are yet drawn, the partial points are deleted. To stop the interactive plot, click on the title (or in the top of the graph)

Usage

```
plotGPApartial(x, axes = c(1, 2),
  lab.ind.moy = TRUE, habillage = "ind",
  chrono = FALSE, draw.partial = NULL,
  xlim = NULL, ylim = NULL, cex = 1, title = NULL, palette = NULL, ...)
```

Arguments

x	an object of class GPA
axes	a length 2 vector specifying the components to plot
lab.ind.moy	boolean, if TRUE, the label of the mean points are drawn
habillage	string corresponding to the color which are used. If "ind", one color is used for each individual; if "group" the individuals are colored according to the group
chrono	boolean, if TRUE, the partial points of a same point are linked (useful when groups correspond to different moment)
draw.partial	data frame of a boolean variable for all the individuals and all the centers of gravity and with for which the partial points should be drawn (by default, NULL and no partial points are drawn)

xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : palette=palette(c("black","red","blue")); or you can use: palette=palette(rainbow(30)), or in black and white for example: palette=palette(gray(seq(0,.9,len=25)))
...	further arguments passed to or from other methods

Value

Returns the General Procrustes Analysis map.

Author(s)

Elisabeth Morand, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[GPA](#)

plotMFApartial	<i>Plot an interactive Multiple Factor Analysis (MFA) graph</i>
----------------	---

Description

Draw an interactive Multiple Factor Analysis (MFA) graphs.

Usage

```
plotMFApartial(x, axes = c(1, 2),
  lab.ind.moy = TRUE, lab.par = FALSE, habillage = "ind",
  chrono = FALSE, col.hab = NULL, invisible = NULL,
  draw.partial = NULL, xlim = NULL, ylim = NULL,
  cex = 1, title = NULL, palette = NULL, ...)
```

Arguments

x	an object of class MFA
axes	a length 2 vector specifying the components to plot
lab.ind.moy	boolean, if TRUE, the label of the mean points are drawn
lab.par	boolean, if TRUE, the label of the partial points are drawn
habillage	string corresponding to the color which are used. If "ind", one color is used for each individual; if "quali" the individuals are colored according to one categorical variable; if "group" the individuals are colored according to the group

chrono	boolean, if TRUE, the partial points of a same point are linked (useful when groups correspond to different moment)
col.hab	the colors to use. By default, colors are chosen
invisible	list of string; for choix = "ind", the individuals can be omit (invisible = "ind"), or supplementary individuals (invisible="ind.sup") or the center of gravity of the categorical variables (invisible= "quali"); if invisible = c("ind","ind.sup"), just the centers of gravity are drawn
draw.partial	data frame of a boolean variable for all the individuals and all the centers of gravity and with for which the partial points should be drawn (by default, NULL and no partial points are drawn)
xlim	range for the plotted 'x' values, defaulting to the range of the finite values of 'x'
ylim	range for the plotted 'y' values, defaulting to the range of the finite values of 'y'
cex	cf. function par in the graphics package
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
palette	the color palette used to draw the points. By default colors are chosen. If you want to define the colors : palette=palette(c("black","red","blue")); or you can use: palette=palette(rainbow(30)), or in black and white for example: palette=palette(gray(seq(0,.9,len=25)))
...	further arguments passed to or from other methods

Value

Draw a graph with the individuals and the centers of gravity. The graph is interactive and clicking on a point will draw the partial points, if you click on a point for which the partial points are yet drawn, the partial points are deleted. To stop the interactive plot, click on the title (or in the top of the graph)

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agroparcampus-ouest.fr>

See Also

[MFA](#), [plot.MFA](#)

Examples

```
## Not run:
data(wine)
res.wine = MFA(wine,group=c(2,5,3,10,9,2),type=c("n",rep("s",5)),ncp=5,
  name.group=c("orig","olf","vis","olfag","gust","ens"),
  num.group.sup=c(1,6),graph=FALSE)
liste = plotMFApartial(res.wine)

## End(Not run)
```

poison	<i>Poison</i>
--------	---------------

Description

The data used here refer to a survey carried out on a sample of children of primary school who suffered from food poisoning. They were asked about their symptoms and about what they ate.

Usage

```
data(poison)
```

Format

A data frame with 55 rows and 15 columns.

Examples

```
data(poison)
res.mca <- MCA(poison, quanti.sup = 1:2, quali.sup=c(3,4))
```

poison.text	<i>Poison</i>
-------------	---------------

Description

The data used here refer to a survey carried out on a sample of children of primary school who suffered from food poisoning. They were asked about their symptoms and about what they ate.

Usage

```
data(poison)
```

Format

A data frame with 55 rows and 3 columns (the sex, if they are sick or not, and a textual variable with their symptom and what they eat).

Examples

```
data(poison.text)
res.text <- textual(poison.text, num.text = 3, contingency.by = c(1,2))
## Contingence table for the sex variable, the sich variable and the couple
## of variable sick-sex
res.text2 <- textual(poison.text, num.text = 3, contingency.by = list(1,2,c(1,2)))
```

poulet

Donnees genomiques sur les poulets

Description

Donnees genomiques sur les poulets

Usage

```
data(poulet)
```

Format

Un data frame avec 43 observations sur 7407 variables.

Régime a factor with levels J16 J16R16 J16R5 J48 J48R24 N

Autre variable variable quantitative correspondant a l'expression d'un gene

Examples

```
## Not run:
data(poulet)
res.pca = PCA(poulet,quali.sup=1, graph=FALSE)
plot(res.pca)
plot(res.pca,habillage=1,label="quali",palette=palette(c("black","red","blue","darkgreen","purple","orange")))
dimdesc(res.pca)
## Dessine des ellipses autour des centres de gravite
aa=cbind.data.frame(poulet[,1],res.pca$ind$coord)
bb=coord.ellipse(aa,bary=TRUE)
plot.PCA(res.pca,habillage=1,ellipse=bb)

## End(Not run)
```

prefpls

Scatter plot and additional variables with quality of representation contour lines

Description

This function is useful to interpret the usual graphs (x,y) with additional quantitative variables.

Usage

```
prefpls(donnee, var1 = 1, var2 = 2, firstvar = 3,
        lastvar = ncol(donnee), levels = c(0.2,0.4,0.6,0.7,0.8,0.9,1),
        asp = 1, nbchar = max(nchar(colnames(donnee))), title = NULL,
        choix="var")
```

Arguments

donnee	a data frame made up of quantitative variables
var1	the position of the variable corresponding to the x-axis
var2	the position of the variable corresponding to the y-axis
firstvar	the position of the first endogenous variable
lastvar	the position of the last endogenous variable (by default the last column of <code>donnee</code>)
levels	a list of the levels displayed in the graph of variables
asp	aspect ratio for the graph of the individuals
nbchar	the number of characters used for the labels of the variables
title	string corresponding to the title of the graph you draw (by default NULL and a title is chosen)
choix	the graph to plot ("ind" for the individuals, "var" for the variables)

Details

This function is very useful when there is a strong correlation between two variables x and y

Value

A scatter plot of the individuals
A graph with additional variables and the quality of representation contour lines.

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

References

Husson, F. & Pagès, J. (2005). Scatter plot and additional variables. *Journal of applied statistics*

Examples

```
data(decathlon)
prefpls(decathlon[,c(11,12,1:10)])
```

```
print.AFDM
```

Print the Multiple Factor Analysis of mixt Data (AFDM) results

Description

Print the Multiple Factor Analysis of mixt Data (AFDM) results.

Usage

```
## S3 method for class 'AFDM'
print(x, file = NULL, sep = ";", ...)
```

Arguments

x	an object of class AFDM
file	A connection, or a character string naming the file to print to. If NULL (the default), the results are not printed in a file
sep	character string to insert between the objects to print (if the argument file is not NULL)
...	further arguments passed to or from other methods

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[AFDM](#)

print.CA	<i>Print the Correspondance Analysis (CA) results</i>
----------	---

Description

Print the Correspondance Analysis (CA) results.

Usage

```
## S3 method for class 'CA'  
print(x, file = NULL, sep = ";", ...)
```

Arguments

x	an object of class CA
file	A connection, or a character string naming the file to print to. If NULL (the default), the results are not printed in a file
sep	character string to insert between the objects to print (if the argument file is not NULL)
...	further arguments passed to or from other methods

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[CA](#), [write.infile](#)

print.GPA

Print the Generalized Procrustes Analysis (GPA) results

Description

Print the Generalized Procrustes Analysis (GPA) results.

Usage

```
## S3 method for class 'GPA'  
print(x, file = NULL, sep = ";", ...)
```

Arguments

x	an object of class GPA
file	A connection, or a character string naming the file to print to. If NULL (the default), the results are not printed in a file
sep	character string to insert between the objects to print (if the argument file is not NULL)
...	further arguments passed to or from other methods

Author(s)

Elisabeth Morand, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[GPA](#), [write.infile](#)

print.HMFA

Print the Hierarchical Multiple Factor Analysis results

Description

Print the Hierarchical Multiple Factor Analysis results.

Usage

```
## S3 method for class 'HMFA'  
print(x, file = NULL, sep = ";", ...)
```

Arguments

x	an object of class HMFA
file	A connection, or a character string naming the file to print to. If NULL (the default), the results are not printed in a file
sep	character string to insert between the objects to print (if the argument file is not NULL)
...	further arguments passed to or from other methods

Author(s)

Sebastien Le, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[HMFA](#), [write.infile](#)

print.MCA

Print the Multiple Correspondance Analysis (MCA) results

Description

Print the Multiple Correspondance Analysis (MCA) results.

Usage

```
## S3 method for class 'MCA'  
print(x, file = NULL, sep = ";", ...)
```

Arguments

x	an object of class MCA
file	A connection, or a character string naming the file to print to. If NULL (the default), the results are not printed in a file
sep	character string to insert between the objects to print (if the argument file is not NULL)
...	further arguments passed to or from other methods

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[MCA](#), [write.infile](#)

print.MFA *Print the Multiple Factor Analysis results*

Description

Print the Multiple Factor Analysis results.

Usage

```
## S3 method for class 'MFA'  
print(x, file = NULL, sep = ";", ...)
```

Arguments

x	an object of class MFA
file	A connection, or a character string naming the file to print to. If NULL (the default), the results are not printed in a file
sep	character string to insert between the objects to print (if the argument file is not NULL)
...	further arguments passed to or from other methods

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[MFA](#), [write.infile](#)

print.PCA *Print the Principal Component Analysis (PCA) results*

Description

Print the Principal Component Analysis (PCA) results.

Usage

```
## S3 method for class 'PCA'  
print(x, file = NULL, sep = ";", ...)
```

Arguments

x	an object of class PCA
file	A connection, or a character string naming the file to print to. If NULL (the default), the results are not printed in a file
sep	character string to insert between the objects to print (if the argument file is not NULL)
...	further arguments passed to or from other methods

Author(s)

Jeremy Mazet, Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[PCA](#), [write.infile](#)

Examples

```
## Not run:
data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup = 13)
print(res.pca, file="c:/essai.csv", sep = ";")

## End(Not run)
```

reconst

Reconstruction of the data from the PCA or MFA results

Description

Reconstruct the data from the PCA or MFA results.

Usage

```
reconst(res, ncp=NULL)
```

Arguments

res	an object of class PCA or MFA
ncp	number of dimensions used to reconstitute the data (by default NULL and the number of dimensions calculated for the PCA or MFA is used)

Value

Returns a data frame with the number of individuals and the number of variables used for the PCA or MFA

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>, Julie Josse <Julie.Josse@agrocampus-ouest.fr>

See Also

[PCA](#), [MFA](#)

Examples

```
data(decathlon)
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup=13, graph=FALSE)
rec <- reconst(res.pca,ncp=2)
```

simule

Simulate by bootstrap

Description

Simulate by bootstrap

Usage

```
simule(data, nb.simul)
```

Arguments

data	A data frame from which the rows are the original data from which the simulated data are calculated (by the average of a bootstrap sample. The columns corresponds to the variables for which the simulation should be done. The first column must be a factor allowing to group the rows. A bootstrap simulation is done for each level of this factor.
nb.simul	The number of simulations.

Details

The simulation is independently done for each level of the factor. The number of rows can be different for each levels.

Value

mean	Data.frame with all the levels of the factor variable, and for each variable, the mean of the original data.
simul	Data.frame with all the levels of the factor variable, and for each variable, the nb.simul bootstrap simulations.
simul.mean	Data.frame with all the levels of the factor variable, and for each variable, the mean of the simulated data.

Author(s)

Jeremy Mazet

`svd.triplet`*Singular Value Decomposition of a Matrix*

Description

Compute the singular-value decomposition of a rectangular matrix with weights for rows and columns.

Usage

```
svd.triplet(X, row.w=NULL, col.w=NULL, ncp=Inf)
```

Arguments

<code>X</code>	a data matrix
<code>row.w</code>	vector with the weights of each row (NULL by default and the weights are uniform)
<code>col.w</code>	vector with the weights of each column (NULL by default and the weights are uniform)
<code>ncp</code>	the number of components kept for the outputs

Value

<code>d</code>	a vector containing the singular values of 'x';
<code>u</code>	a matrix whose columns contain the left singular vectors of 'x';
<code>v</code>	a matrix whose columns contain the right singular vectors of 'x'.

See Also[svd](#)

tab.disjonctif	<i>Make a disjonctif table</i>
----------------	--------------------------------

Description

Make a disjonctif table.

Usage

```
tab.disjonctif(tab)
```

Arguments

tab	a data frame with factors
-----	---------------------------

Value

The disjonctif table

tab.disjonctif.prop	<i>Make a disjunctive table when missing values are present</i>
---------------------	---

Description

Create a disjunctive table. The missing values are replaced by the proportion of the category.

Usage

```
tab.disjonctif.prop(tab, seed=NULL, row.w=NULL)
```

Arguments

tab	a data frame with factors
row.w	an optional row weights (by default, a vector of 1 for uniform row weights)
seed	a single value, interpreted as an integer for the set.seed function (if seed = NULL, missing values are initially imputed by the mean of each variable)

Value

The disjonctif table.prop

tea	<i>tea (data)</i>
-----	-------------------

Description

The data used here concern a questionnaire on tea. We asked to 300 individuals how they drink tea (18 questions), what are their product's perception (12 questions) and some personal details (4 questions).

Usage

```
data(tea)
```

Format

A data frame with 300 rows and 36 columns. Rows represent the individuals, columns represent the different questions. The first 18 questions are active ones, the 19th is a supplementary quantitative variable (the age) and the last variables are supplementary categorical variables.

Examples

```
data(tea)
res.mca=MCA(tea,quanti.sup=19,quali.sup=20:36)
plot(res.mca,invisible=c("var","quali.sup","quanti.sup"),cex=0.7)
plot(res.mca,invisible=c("ind","quali.sup","quanti.sup"),cex=0.8)
plot(res.mca,invisible=c("quali.sup","quanti.sup"),cex=0.8)
dimdesc(res.mca)
plotellipses(res.mca,keepvar=1:4)

## make a hierarchical clustering: click on the tree to define the number of clusters
## HCPC(res.mca)
```

textual	<i>Text mining</i>
---------	--------------------

Description

Calculates the number of occurrence of each words and a contingency table

Usage

```
textual(tab, num.text, contingency.by=1:ncol(tab),
        maj.in.min = TRUE, sep.word=NULL)
```

Arguments

<code>tab</code>	a data frame with one textual variable
<code>num.text</code>	indice of the textual variable
<code>contingence.by</code>	a list with the indices of the variables for which a contingency table is calculated by default a contingency table is calculated for all the variables (except the textual one). A contingency table can also be calculated for couple of variables. If <code>contingence.by</code> is equal to <code>num.text</code> , then the contingency table is calculated for each row of the data table
<code>maj.in.min</code>	boolean, if TRUE majuscule are transformed in minuscule
<code>sep.word</code>	a string with all the characters which correspond to separator of words (by default <code>sep.word = " ; () , ? . / : ' ! \$ % = + \ n ; { } < > [] @ - "</code>)

Value

Returns a list including:

<code>cont.table</code>	the contingency table with in rows the categories of the categorical variables (or the couple of categories), and in column the words, and in each cell the number of occurrence
<code>nb.words</code>	a data.frame with all the words and for each word, the number of lists in which it is present, and the number of occurrence

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

See Also

[CA](#)

Examples

```
data(poison.text)
res.text <- textual(poison.text, num.text = 3, contingence.by = c(1,2))
## Contingence table for the sex variable, the sich variable and the couple
## of variable sick-sex
res.text2 <- textual(poison.text, num.text = 3, contingence.by = list(1,2,c(1,2)))
```

wine

Wine

Description

The data used here refer to 21 wines of Val de Loire.

Usage

```
data(wine)
```

Format

A data frame with 21 rows (the number of wines) and 31 columns: the first column corresponds to the label of origin, the second column corresponds to the soil, and the others correspond to sensory descriptors.

Source

Centre de recherche INRA d'Angers

Examples

```
data(wine)

## Example of PCA
res.pca = PCA(wine,ncp=5, quali.sup = 1:2)

## Not run:
## Example of MCA
res.mca = MCA(wine,ncp=5, quanti.sup = 3:ncol(wine))

## Example of MFA
res.mfa = MFA(wine,group=c(2,5,3,10,9,2),type=c("n",rep("s",5)),ncp=5,
  name.group=c("orig","olf","vis","olfag","gust","ens"),
  num.group.sup=c(1,6),graph=FALSE)
plotellipses(res.mfa)
plotellipses(res.mfa,keepvar="Label") ## for 1 variable

## End(Not run)
```

write.infile

Print in a file

Description

Print in a file.

Usage

```
write.infile(X, file, sep=";", append = FALSE, nb.dec=4)
```

Arguments

X	an object of class list, data.frame, matrix, ...
file	A connection, or a character string naming the file to print to
sep	character string to insert between the objects to print (if the argument file is not NULL)
append	logical. If TRUE output will be appended to file; otherwise, it will overwrite the contents of file.
nb.dec	number of decimal printed, by default 4

Author(s)

Francois Husson <Francois.Husson@agrocampus-ouest.fr>

Examples

```
## Not run:  
data(decathlon)  
res.pca <- PCA(decathlon, quanti.sup = 11:12, quali.sup = 13)  
write.infile(res.pca, file="c:/essai.csv", sep = ";")  
  
## End(Not run)
```

Index

- *Topic **algebra**
 - svd.triplet, 55
 - tab.disjonctif, 56
 - tab.disjonctif.prop, 56
- *Topic **datasets**
 - children, 6
 - decathlon, 10
 - J0, 20
 - mortality, 25
 - poison, 46
 - poison.text, 46
 - poulet, 47
 - tea, 57
 - wine, 58
- *Topic **dplot**
 - coord.ellipse, 9
 - graph.var, 16
 - plot.AFDM, 28
 - plot.CA, 29
 - plot.catdes, 30
 - plot.DMFA, 31
 - plot.GPA, 32
 - plot.HCPC, 33
 - plot.HMFA, 35
 - plot.MCA, 36
 - plot.MFA, 38
 - plot.PCA, 40
 - plotGPApartial, 43
 - plotMFAPartial, 44
 - prefpls, 47
 - simule, 54
- *Topic **multivariate**
 - AFDM, 3
 - CA, 4
 - catdes, 5
 - coeffRV, 7
 - condes, 8
 - dimdesc, 11
 - DMFA, 12
 - estim_ncp, 13
 - GPA, 14
 - HCPC, 17
 - HMFA, 19
 - MCA, 21
 - MFA, 23
 - PCA, 26
 - plotellipses, 41
 - reconst, 53
 - textual, 57
- *Topic **print**
 - print.AFDM, 48
 - print.CA, 49
 - print.GPA, 50
 - print.HMFA, 50
 - print.MCA, 51
 - print.MFA, 52
 - print.PCA, 52
 - write.infile, 59
- AFDM, 3, 29, 49
- agnes, 17
- CA, 4, 11, 30, 49, 58
- catdes, 5, 9, 18, 31
- children, 6
- coeffRV, 7
- condes, 6, 8
- coord.ellipse, 9
- decathlon, 10
- dimdesc, 5, 11, 13, 20, 22, 25, 27
- DMFA, 12, 17, 32
- estim_ncp, 13
- GPA, 14, 33, 44, 50
- graph.var, 16
- HCPC, 17, 34
- HMFA, 11, 17, 19, 36, 51

JO, [20](#)

MCA, [11](#), [17](#), [21](#), [37](#), [43](#), [51](#)

MFA, [11](#), [17](#), [23](#), [39](#), [45](#), [52](#), [54](#)

mortality, [25](#)

par, [16](#), [29](#), [30](#), [33](#), [35](#), [37](#), [39](#), [41](#), [42](#), [44](#), [45](#)

PCA, [11](#), [14](#), [17](#), [26](#), [41](#), [43](#), [53](#), [54](#)

plot.AFDM, [4](#), [28](#)

plot.CA, [5](#), [29](#)

plot.catdes, [6](#), [30](#)

plot.DMFA, [13](#), [31](#)

plot.GPA, [32](#)

plot.HCPC, [18](#), [33](#)

plot.HMFA, [20](#), [35](#)

plot.MCA, [22](#), [36](#)

plot.MFA, [25](#), [38](#), [45](#)

plot.PCA, [27](#), [40](#)

plotellipses, [22](#), [41](#)

plotGPApartial, [43](#)

plotMFApartial, [44](#)

poison, [46](#)

poison.text, [46](#)

poulet, [47](#)

prefpls, [47](#)

print.AFDM, [4](#), [48](#)

print.CA, [5](#), [49](#)

print.GPA, [50](#)

print.HMFA, [20](#), [50](#)

print.MCA, [22](#), [51](#)

print.MFA, [25](#), [52](#)

print.PCA, [27](#), [52](#)

reconst, [53](#)

simule, [10](#), [54](#)

svd, [55](#)

svd.triplet, [55](#)

tab.disjonctif, [56](#)

tab.disjonctif.prop, [56](#)

tea, [57](#)

textual, [57](#)

wine, [58](#)

write.infile, [49–53](#), [59](#)

xyplot, [42](#)