Package 'FactoMineR'

April 26, 2009

Version 1.12

Date 2009-20-03

Title Factor 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

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 2009-04-26 14:21:05

R topics documented:

AFDM 2
CA 4
catdes 5
children 6
coeffRV 7
condes 8
coord.ellipse 9
decathlon 10
dimdesc 10
DMFA 11
GPA 12
graph.var 14
HCPC 15

HMFA . . . 17
JO . . . 18
MCA . . . 19
MFA . . . 20
PCA . . . 23
plot.AFDM . . . 24
plot.CA . . . 26
plot.catdes . . . 27
plot.DMFA . . . 28
plot.GPA . . . 29
plot.GPApartial . . . 30
plot.HCPC . . . 31
plot.HMFA . . . 32
plot.MCA . . . 33
plot.MFA . . . 35
plot.MFApartial . . . 37
plot.PCA . . . 38
plotellipses . . . 40
poison . . . 41
poison.text . . . 42
poulet . . . 42
prefpls . . . 43
print.AFDM . . . 44
print.CA . . . 45
print.GPA . . . 45
print.HMFA . . . 46
print.MCA . . . 47
print.MFA . . . 47
print.PCA . . . 48
simule . . . 49
svd.triplet . . . 50
tab.disjonctif . . . 50
tea . . . 51
textual . . . 52
wine . . . 53
write.infile . . . 54

Index **55**

`AFDM` *Multiple Factor Analysis for Mixed Data*

Description

Performs Multiple Factor Analysis with both quantitative and qualitative 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
link.group	
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 qualitative 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@agrocampus-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 *Correspondence Analysis (CA)*

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

See Also

`print.CA`, `plot.CA`, `dimdesc`

Examples

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

---

catdes *Categories description*

Description

Description of the categories of one factor by qualitative 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 qualitative variables and a set of quantitative variables and/or qualitative variables
`num.var`	the indice of the variable to caracterized
`proba`	the significance threshold considered to caracterized the category (by default 0.05)

Value

Returns a list including:

`test.chi`	The qualitative 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`
`category`	description of each category of the `num.var` by each category of all the qualitative variables
`quanti`	the description of each category of the `num.var` variable by the quantitative variables.

Author(s)

Francois Husson ⟨Francois.Husson@agrocampus-ouest.fr⟩

References

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 *Children (data)*

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 qualitative 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 qualitative variables
num.var	the indice of the variable to caracterized
proba	the significance threshold considered to caracterized 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 qualitative 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 qualitative variables

Author(s)

Francois Husson ⟨Francois.Husson@agrocampus-ouest.fr⟩

See Also

catdes

Examples

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

`coord.ellipse` *Construct confidence ellipses*

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 qualitative 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 *Dimension description*

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 caracterized 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 qualitative variables

Author(s)

Francois Husson ⟨Francois.Husson@agrocampus-ouest.fr⟩

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 qualitative 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 qualitative 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 qualitative 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 qualitative 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@agrocampus-ouest.fr⟩

See Also

`plot.DMFA`, `dimdesc`

Examples

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

`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

df	a data frame with *n* rows (individuals) and *p* columns (quantitative varaibles)
tolerance	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 tolerance
nbiteration	the maximum number of iterations until the algorithm stops
scale	a boolean, if TRUE (which is the default value) scaling is required
group	a vector indicating the number of variables in each group
name.group	a vector indicating the name of the groups (the groups are successively named group.1, group.2 and so on, by default)
graph	boolean, if TRUE a graph is displayed
axes	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 assesor (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
plot.GPApartial (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 Classification on Principle Components (HCPC)*

Description

Performs an unsupervised hierarchical classification on results from a factor analysis. It is possible to cut the tree by clicking at the suggested (or an other) level. Results inlude 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, ...)
```

Arguments

res	Either the result of a factor analysis, a dataframe, or a vector.
nb.clust	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 nb.cluters clusters.
consol	a boolean. If TRUE, a k-means consolidation is performed.
iter.max	An integer. The maximum number of iterations for the consolidation.
min	an integer. The least possible number of clusters suggested.
max	an integer. The higher possible number of clusters suggested.
metric	The metric used to built the tree. See agnes for details.
method	The method used to built the tree. See agnes for details.
order	A boolean. If TRUE, clusters are ordered following their center coordinate on the first axis.
graph.scale	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.
nb.par	An integer. The number of edited paragons.
graph	If TRUE, graphics are displayed. If FALSE, no graph are displayed.
proba	The probability used to select axes and variables in catdes (see catdes for details.
...	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(clusters n+1)/i(cluster n)).

The absolut loss of inertia (i(cluster n)-i(cluster n+1)) is ploted with the tree.

Value

Returns a list including:

data.clust	The original data with a supplementary row called class containing the partition.
desc.fact	The description of the classes by factors (axes) or variables (var). See catdes for details.
call	A list or parameters and internal objects.
ind.desc	The paragons (para) 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)

## To choose an other partition on the tree:
res.HCPC=HCPC(res.hcpc)
## End(Not run)
```

HMFA *Hierarchical Multiple Factor Analysis*

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

`X`	a `data.frame`
`H`	a list with one vector for each hierarchical level; in each vector the number of variables or the number of group constituting the group
`type`	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 qualitative variables; by default, all the variables are quantitative and the variables are scaled unit
`ncp`	number of dimensions kept in the results (by default 5)
`graph`	boolean, if TRUE a graph is displayed
`axes`	a length 2 vector specifying the components to plot
`name.group`	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:

`eig`	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
`group`	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)
`ind`	a list of matrices with all the results for the active individuals (coordinates, square cosine, contributions)
`quanti.var`	a list of matrices with all the results for the quantitative variables (coordinates, correlation between variables and axes)
`quali.var`	a list of matrices with all the results for the supplementary qualitative variables (coordinates of each categories of each variables, and v.test which is a criterion with a Normal distribution)
`partial`	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 *Jeu de donnes sur nombre de medailles par pays*

Description

Le jeu de donnees est un tableau de contingence croisant en lignes les epreuves d'athletisme et en colonnes les differents pays. Chaque cellule contient le nombre total de medailles (or, argent et bronze) obtenues lors des olympiades de 1992 à 2008 (Barcelone 1992, Atlanta 1996, Sydney 2000, Athenes 2004, Pekin 2008).

Usage

```
data(JO)
```

Format

Un data frame avec en lignes les 24 disciplines d'athletisme et en colonne les 58 pays ayant obtenu au moins une medaille

Examples

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

MCA *Multiple Correspondence Analysis (MCA)*

Description

Performs Multiple Correspondence Analysis (MCA) with supplementary individuals, supplementary quantitative variables and supplementary qualitative 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)
```

Arguments

X	a data frame with *n* rows (individuals) and *p* columns (categorical variables)
ncp	number of dimensions kept in the results (by default 5)
ind.sup	a vector indicating the indexes of the supplementary individuals
quanti.sup	a vector indicating the indexes of the quantitative supplementary variables
quali.sup	a vector indicating the indexes of the qualitative supplementary variables
graph	boolean, if TRUE a graph is displayed
level.ventil	level under which the category is ventilated; by default, 0 and no nventilation is done
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
`var`	a list of matrices containing all the results for the active variables (coordinates, square cosine, contributions, v.test)
`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 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)
`quali.sup`	a list of matrices with all the results for the supplementary qualitative variables (coordinates of each categories of each variables, square cosine and v.test which is a criterion with a Normal distribution)
`call`	a list with some statistics

Returns the individuals factor map and the variables factor map.

Author(s)

Jeremy Mazet, Francois Husson ⟨Francois.Husson@agrocampus-ouest.fr⟩

See Also

`plotellipses`,`print.MCA`, `plot.MCA`, `dimdesc`

Examples

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

`MFA` *Multiple Factor Analysis (MFA)*

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 or qualitative.
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))
```

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; three possibilities: "c" or "s" for quantitative variables (the difference is that for "s" variables are scaled to unit variance), "n" for qualitative variables; 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, uniform row weights)
axes	a length 2 vector specifying the components to plot

Value

summary.quali	a summary of the results for the qualitative variables
summary.quanti	a summary of the results for the quantitative variables
separate.analyses	the results for the separate analyses
eig	a matrix containing all the eigenvalues, the percentage of variance and the cumulative percentage of variance
group	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)
rapport.inertie	inertia ratio
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.var`	a list of matrices containing all the results for the quantitative variables (coordinates, correlation between variables and axes)
`quali.var`	a list of matrices containing all the results for the supplementary qualitative variables (coordinates of each categories of each variables, and v.test which is a criterion with a Normal distribution)
`partial.axes`	a list of matrices containing all the results for the partial axes (coordinates, correlation between variables and axes, correlation between partial axes)

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

Author(s)

Jeremy Mazet, Francois Husson ⟨Francois.Husson@agrocampus-ouest.fr⟩

References

Escofier, B. and Pagès, J. (1994) Multiple Factor Analysis (AFMULT package), *Computational Statistics and Data Analysis*, **18**, 121-140.

See Also

`print.MFA`, `plot.MFA`, `dimdesc`

Examples

```
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))
barplot(aa$eig[,1],main="Eigenvalues",names.arg=1:nrow(aa$eig))

## Not run:
#### Interactive graph
liste = plot.MFApartial(aa)
plot(aa,choix="ind",habillage = "Terroir")

###Example 2
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)
## End(Not run)
```

PCA *Principal Component Analysis (PCA)*

Description

Performs Principal Component Analysis (PCA) with supplementary individuals, supplementary quantitative variables and supplementary qualitative 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

X	a data frame with *n* rows (individuals) and *p* columns (numeric variables)
ncp	number of dimensions kept in the results (by default 5)
scale.unit	a boolean, if TRUE (value set by default) then data are scaled to unit variance
ind.sup	a vector indicating the indexes of the supplementary individuals
quanti.sup	a vector indicating the indexes of the quantitative supplementary variables
quali.sup	a vector indicating the indexes of the qualitative supplementary variables
row.w	an optional row weights (by default, uniform row weights)
col.w	an optional column weights (by default, uniform column weights)
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 qualitative 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)

Jeremy Mazet, Francois Husson ⟨Francois.Husson@agrocampus-ouest.fr⟩

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)
aa=cbind.data.frame(decathlon[,13],res.pca$ind$coord)
bb=coord.ellipse(aa,bary=TRUE)
plot.PCA(res.pca,habillage=13,ellipse=bb)
```

---

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 = TRUE, ...)
```

Arguments

x — an object of class AFDM

choix — a string corresponding to the graph that you want to do ("ind" for the individual or qualitative variables graph, "group" for all the variables (quantitative and qualitative), "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 drwan
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 qualitative 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 qualitative 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 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@agrocampus-ouest.fr⟩

See Also

AFDM

`plot.CA` *Draw the Correspondance Analysis (CA) graphs*

Description

Draw the Correspondance Analysis (CA) graphs.

Usage

```
plot.CA(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 `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 individuals factor map and the variables factor map.

Author(s)

Jeremy Mazet, Francois Husson ⟨Francois.Husson@agrocampus-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

```
plot.catdes (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 qualitative variables are used. If "quanti", only the 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` *Make 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 qualitative variables.

Usage

```
plot.DMFA(x, axes = c(1, 2), choix = "ind", label="all",
    invisible = NULL, palette = NULL, new.plot = TRUE, ...)
```

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"
`invisible`	string indicating if some points should not be drawn ("ind" or "quali")
`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 ⟨Francois.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

```
plot.GPA(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 drwan
`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.GPApartial` *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

```
plot.GPApartial(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 drwan
`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

`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

```
plot.HCPC(x, axes=c(1,2), choice="3D.map", rect=TRUE,
  draw.tree=TRUE, ind.names=TRUE, t.level="all", title=NULL,
  new.plot=TRUE, 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 ploted. 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

```
plot.HMFA(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 = TRUE, ...)
```

Arguments

x	an object of class HMFA
axes	a length 2 vector specifying the components to plot
num	number of grpahs in a same windows
choix	a string corresponding to the graph that you want to do ("ind" for the individual or qualitative 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 drwan
lab.var	boolean, if TRUE, the label of the variables are drwan
lab.ind.moy	boolean, if TRUE, the label of the mean points are drwan

invisible	list of string; for choix ="ind", the individuals can be omit (invisible = "ind"), or the centers of gravity of the qualitative 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)

Jere½my 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 Correspondance Analysis (MCA) graphs*

Description

Draw the Multiple Correspondance Analysis (MCA) graphs.

Usage

```
plot.MCA (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 = TRUE, ...)
```

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 qualitative variables, if color ="none" the label is not written
col.quali.sup	a color for the qualitative 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.quanti.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 qualitative variables; if "quali", one color is used for each qualitative variables; else if it is the position of a qualitative 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

```
plot.MFA(x, axes = c(1, 2), choix = "ind", ellipse = NULL, ellipse.par = NULL,
    lab.grpe = TRUE, lab.var = TRUE, lab.ind.moy = TRUE,
    lab.par = FALSE, 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 = TRUE, ...)
```

Arguments

`x`	an object of class MFA
`choix`	a string corresponding to the graph that you want to do ("ind" for the individual or qualitative variables graph, "var" for the quantitative variables graph, "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 label of the groups are drwan
`lab.var`	boolean, if TRUE, the label of the variables are drwan
`lab.ind.moy`	boolean, if TRUE, the label of the mean points are drwan
`lab.par`	boolean, if TRUE, the label of the partial points are drwan
`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 qualitative 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 omit (invisible = "ind"), or supplementary individuals (invisible="ind.sup") or the centerg of gravity of the qualitative 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 with the points are not 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
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

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.MFApartial` *Plot an interactive Multiple Factor Analysis (MFA) graph*

Description

Draw an interactive Multiple Factor Analysis (MFA) graphs.

Usage

```
plot.MFApartial(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 drwan
`lab.par`	boolean, if TRUE, the label of the partial points are drwan
`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 qualitative 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 centerg of gravity of the qualitative 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@agrocampus-ouest.fr⟩

See Also

MFA, plot.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)
liste = plot.MFApartial(aa)
## End(Not run)
```

---

plot.PCA *Make the Principal Component Analysis (PCA) graphs*

Description

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

Usage

```
plot.PCA(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.quanti.sup = "blue",
    col.var = "black", label = "all", invisible = NULL,
    lim.cos2.var = 0., cex = 1, title = NULL, palette=NULL,
    new.plot = TRUE, ...)
```

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 qualitative variable (give the number of the qualitative 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 qualitative variable sonly if there is not habillage
col.quanti.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", "quanti.sup"))
invisible	string indicating if some points should not be drawn ("ind", "ind.sup" or "quali" for the individual graph and "var" or "quanti.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@agrocampus-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, ...)
```

Arguments

model	an object of class MCA
keepvar	a boolean or numeric vector of indexes of variables or a character vector of names of variables. If keepvar is "all", "quanti" or "quanti.sup" variables which are plotted are all the qualitative variables, only those which are used to compute the MCA (active variables) or only the supplementary qualitative 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 par in the **graphics** package
pch	plotting character for coordinates, cf. function par in the **graphics** package
pch.means	plotting character for means, cf. function par in the **graphics** package
type	cf. function xyplot 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 qualitative variable, each variable are stacked under each other. If NULL, names are taken from the (modified) dataset extracted from modele
...	further arguments passed to or from other methods

Value

Return a graph with the ellipses.

Author(s)

Pierre-Andre Cornillon, Francois Husson ⟨Francois.Husson@agrocampus-ouest.fr⟩

See Also

MCA

Examples

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

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

---

poison *Poison*

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 *Poison*

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, 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)))
```

---

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","p
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 donnee)
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 invividuals
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 *Print the Correspondance Analysis (CA) results*

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)
```

`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 simualte 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)
```

Arguments

X	a data matrix
row.w	vector with the weights of each row (NULL by default and the weights are uniform)
col.w	vector with the weights of each column (NULL by default and the weights are uniform)

Value

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

See Also

svd

tab.disjonctif *Make a disjonctif table*

Description

Make a disjonctif table.

Usage

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

Arguments

tab	a data frame with factors

Value

The disjonctif table

`tea` *tea (data)*

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(children)
```

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 qualitative 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
res.mca=MCA(tea,quanti.sup=19,quali.sup=20:36,ncp=20,graph=FALSE)
library(cluster)
classif = agnes(res.mca$ind$coord,method="ward")
plot(classif,main="Dendrogram",ask=FALSE,which.plots=2,labels=FALSE)
clust = cutree(classif,k=3)
tea.comp = cbind.data.frame(tea,res.mca$ind$coord[,1:3],factor(clust))
## describe the clusters
catdes(tea.comp,ncol(tea.comp))
## represent the clusters*
res.aux=MCA(tea.comp,quanti.sup=c(19,37:39),quali.sup=c(20:36,40),graph=FALSE)
plot(res.aux,invisible=c("quali.sup","var","quanti.sup"),habillage=40)
```

`textual` *Text mining*

Description

Calculates the number of occurence of each words and a contingence table

Usage

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

Arguments

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

Value

Returns a list including:

`cont.table`	the contingence table with in rows the categories of the qualitative variables (or the couple of categories), and in column the words, and in each cell the number of occurence
`nb.words`	a data.frame with all the words and for each word, the number of lists in which it is present, and the number of occurence

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)

## 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)
```

`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, 49
tab.disjonctif, 49
∗Topic **datasets**
children, 5
decathlon, 9
JO, 17
poison, 40
poison.text, 41
poulet, 41
tea, 50
wine, 52
∗Topic **dplot**
coord.ellipse, 8
graph.var, 13
plot.AFDM, 23
plot.CA, 25
plot.catdes, 26
plot.DMFA, 27
plot.GPA, 28
plot.GPApartial, 29
plot.HCPC, 30
plot.HMFA, 31
plot.MCA, 32
plot.MFA, 34
plot.MFApartial, 36
plot.PCA, 37
prefpls, 42
simule, 48
∗Topic **multivariate**
AFDM, 1
CA, 3
catdes, 4
coeffRV, 6
condes, 7
dimdesc, 9
DMFA, 10
GPA, 11
HCPC, 14
HMFA, 16
MCA, 18
MFA, 19
PCA, 22
plotellipses, 39
textual, 51
∗Topic **print**
print.AFDM, 43
print.CA, 44
print.GPA, 44
print.HMFA, 45
print.MCA, 46
print.MFA, 46
print.PCA, 47
write.infile, 53

AFDM, 1, *24*, *43*
agnes, *15*

CA, 3, *10*, *26*, *44*, *51*
catdes, 4, *7*, *15, 16*, *26*
children, 5
coeffRV, 6
condes, *5*, 7
coord.ellipse, 8

decathlon, 9
dimdesc, *4*, 9, *11*, *17*, *19*, *21*, *23*
DMFA, 10, *14*, *27*

GPA, 11, *28*, *30*, *45*
graph.var, 13

HCPC, 14, *30, 31*
HMFA, *10*, *14*, 16, *32*, *45*

JO, 17

MCA, *10*, *14*, 18, *33*, *40*, *46*
MFA, *10*, *14*, 19, *35*, *37*, *47*

par, *14*, *24, 25*, *28, 29*, *32, 33*, *35, 36*, *38, 39*
PCA, *10*, *14*, 22, *38*, *47*
plot.AFDM, *2*, 23
plot.CA, *4*, 25
plot.catdes, *5*, 26
plot.DMFA, *11*, 27
plot.GPA, 28
plot.GPApartial, 29
plot.HCPC, *16*, 30
plot.HMFA, *17*, 31
plot.MCA, *19*, 32
plot.MFA, *21*, 34, *37*
plot.MFApartial, 36
plot.PCA, *23*, 37
plotellipses, *19*, 39
poison, 40
poison.text, 41
poulet, 41
prefpls, 42
print.AFDM, *2*, 43
print.CA, *4*, 44
print.GPA, 44
print.HMFA, *17*, 45
print.MCA, *19*, 46
print.MFA, *21*, 46
print.PCA, *23*, 47

simule, *8*, 48
svd, *49*
svd.triplet, 49

tab.disjonctif, 49
tea, 50
textual, 51

wine, 52
write.infile, *44–47*, 53

xyplot, *39*