Package ‘GDAtools’

October 12, 2022

Type Package
Title A Toolbox for Geometric Data Analysis and More
Version 1.7.2
Author Nicolas Robette
Maintainer Nicolas Robette <nicolas.robette@uvsq.fr>
Imports MASS, wdm, FactoMineR, nleqslv, nnet, ggplot2, ggrepel, RColorBrewer, rlang, GGally
Suggests rmarkdown, knitr, rmdformats, cluster, WeightedCluster, vcd, R.rsp
Description Contains functions for 'specific' Multiple Correspondence Analysis, Class Specific Analysis, Multiple Factor Analysis, 'standardized' MCA, computing and plotting structuring factors and concentration ellipses, inductive tests and others tools for Geometric Data Analysis (Le Roux & Rouanet (2005) <doi:10.1007/1-4020-2236-0>). It also provides functions for the translation of logit models coefficients into percentages (Deauvieau (2010) <doi:10.1177/0759106309352586>), weighted contingency tables, an association measure for contingency tables ("Percentages of Maximum Deviation from Independence", aka PEM, see Ci- bois (1993) <doi:10.1177/07591063930400103>) and some tools to measure and plot bivariate associations between variables (phi, Cramér V, correlation coefficient, eta-squared...).
VignetteBuilder R.rsp
License GPL (>= 2)
Encoding UTF-8
LazyData true
NeedsCompilation no
Repository CRAN
Date/Publication 2022-02-22 14:40:02 UTC
### R topics documented:

<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>angles.csa</td>
<td>3</td>
</tr>
<tr>
<td>assoc.catcont</td>
<td>4</td>
</tr>
<tr>
<td>assoc.twocat</td>
<td>5</td>
</tr>
<tr>
<td>assoc.twocont</td>
<td>7</td>
</tr>
<tr>
<td>assoc.yx</td>
<td>8</td>
</tr>
<tr>
<td>burt</td>
<td>9</td>
</tr>
<tr>
<td>catdesc</td>
<td>10</td>
</tr>
<tr>
<td>conc.ellipse</td>
<td>12</td>
</tr>
<tr>
<td>condesc</td>
<td>13</td>
</tr>
<tr>
<td>contrib</td>
<td>15</td>
</tr>
<tr>
<td>csMCA</td>
<td>16</td>
</tr>
<tr>
<td>darma</td>
<td>17</td>
</tr>
<tr>
<td>dichotom</td>
<td>19</td>
</tr>
<tr>
<td>dimcontrib</td>
<td>19</td>
</tr>
<tr>
<td>dimdesc</td>
<td>21</td>
</tr>
<tr>
<td>dimeta2</td>
<td>22</td>
</tr>
<tr>
<td>dimtypicality</td>
<td>23</td>
</tr>
<tr>
<td>flip.mca</td>
<td>24</td>
</tr>
<tr>
<td>getindexcat</td>
<td>25</td>
</tr>
<tr>
<td>ggadd_attractions</td>
<td>26</td>
</tr>
<tr>
<td>ggadd_chulls</td>
<td>27</td>
</tr>
<tr>
<td>ggadd_corr</td>
<td>29</td>
</tr>
<tr>
<td>ggadd_density</td>
<td>30</td>
</tr>
<tr>
<td>ggadd_ellipses</td>
<td>32</td>
</tr>
<tr>
<td>ggadd_interaction</td>
<td>33</td>
</tr>
<tr>
<td>ggadd_kellipses</td>
<td>35</td>
</tr>
<tr>
<td>ggadd_supind</td>
<td>36</td>
</tr>
<tr>
<td>ggadd_supvar</td>
<td>37</td>
</tr>
<tr>
<td>ggassoc_boxplot</td>
<td>39</td>
</tr>
<tr>
<td>ggassoc_crosstab</td>
<td>40</td>
</tr>
<tr>
<td>ggassoc_phiplot</td>
<td>41</td>
</tr>
<tr>
<td>ggassoc_scatter</td>
<td>43</td>
</tr>
<tr>
<td>ggcloud_indiv</td>
<td>44</td>
</tr>
<tr>
<td>ggcloud_variables</td>
<td>46</td>
</tr>
<tr>
<td>homog.test</td>
<td>48</td>
</tr>
<tr>
<td>indsup</td>
<td>49</td>
</tr>
<tr>
<td>medoids</td>
<td>50</td>
</tr>
<tr>
<td>modif.rate</td>
<td>51</td>
</tr>
<tr>
<td>Movies</td>
<td>52</td>
</tr>
<tr>
<td>multiMCA</td>
<td>53</td>
</tr>
<tr>
<td>Music</td>
<td>54</td>
</tr>
<tr>
<td>pem</td>
<td>55</td>
</tr>
<tr>
<td>phi.table</td>
<td>56</td>
</tr>
<tr>
<td>plot.csMCA</td>
<td>57</td>
</tr>
<tr>
<td>plot.multiMCA</td>
<td>58</td>
</tr>
<tr>
<td>plot.speMCA</td>
<td>60</td>
</tr>
</tbody>
</table>
angles.csa

Cosine similarities and angles between CSA and MCA

Description

Computes the cosines similarities and angles between the dimensions of a CSA and those of a MCA.

Usage

angles.csa(rescsa, resmca)

Arguments

rescsa object of class 'csMCA'
resmca object of class 'MCA' or 'speMCA'

Details

This function is adapted from 'csa.measures' in 'sco.ca' package.

Value

A list of matrices:

- cosines Cosine similarities
- angles Angles

Author(s)

Nicolas Robette
References


See Also

MCA, speMCA, csMCA

Examples

```r
## Performs a specific MCA and a CSA on the 'Music' example data set
## and compute cosine similarities and angles
data(Music)
resmca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
female <- Music$Gender=='Women'
rescsa <- csMCA(Music[,1:5],subcloud=female,excl=c(3,6,9,12,15))
angles.csa(rescsa, resmca)
```

assoc.catcont

*Measures the association between a categorical variable and a continuous variable*

Description

Measures the association between a categorical variable and a continuous variable

Usage

```r
assoc.catcont(x,y,weights=rep(1,length(x)),
               nperm=NULL,distrib="asympt",digits=3)
```

Arguments

- `x` the categorical variable (must be a factor)
- `y` the continuous variable (must be a numeric vector)
- `weights` an optional numeric vector of weights (by default, a vector of 1 for uniform weights)
- `nperm` numeric. Number of permutations for the permutation test of independence. If NULL (default), no permutation test is performed.
- `distrib` the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").
- `digits` integer. The number of digits (default is 3).
**assoc.twocat**

Cross-tabulation and measures of association between two categorical variables

**Value**

A list with the following elements:

- `eta.squared`: eta-squared between the two variables
- `permutation.pvalue`: p-value from a permutation (so non-parametric) test of independence
- `cor`: point biserial correlation between the two variables, for each level of the categorical variable
- `cor.perm.pval`: permutation p-value of the correlation between the two variables, for each level of the categorical variable

**Author(s)**

Nicolas Robette

**References**


**See Also**

`assoc.twocat`, `assoc.twocont`, `assoc.yx`, `condesc`, `catdesc`, `darma`, `ggassoc_boxplot`

**Examples**

```r
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2, 10, 20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
assoc.catcont(group,weight,nperm=100)
```

**Description**

Cross-tabulation and measures of association between two categorical variables

**Usage**

```r
assoc.twocat(x,y,weights=rep.int(1,length(x)),na_value=NULL,nperm=NULL,distrib="asympt")
```
**Arguments**

x  
the first categorical variable (must be a factor)

y  
the second categorical variable (must be a factor)

weights  
an optional numeric vector of weights (by default, a vector of 1 for uniform weights)

na_value  
character. Name of the level for NA category. If NULL (default), NA values are ignored.

nperm  
numeric. Number of permutations for the permutation test of independence. If NULL (default), no permutation test is performed. Default is 1000.

distrib  
the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").

**Value**

A list with the following elements:

freq  
cross-tabulation

prop  
percentages

rprop  
row percentages

cprop  
column percentages

expected  
expected values

chi.squared  
chi-squared value

cramer.v  
Cramer’s V between the two variables

permutation.pvalue  
p-value from a permutation (so non-parametric) test of independence

pearson.residuals  
the table of Pearson residuals, i.e. (observed - expected) / sqrt(expected).

phi  
the table of the phi coefficients for each pair of levels

phi.perm.pval  
the table of permutation p-values for each pair of levels

gather  
a data frame gathering informations, with one row per cell of the cross-tabulation

**Author(s)**

Nicolas Robette

**References**


**See Also**

assoc.catcont, assoc.twocont, assoc.yx, condesc, catdesc, darma, ggassoc_crosstab, ggassoc_phiplot
assoc.twocont

Examples

data(Music)
assoc.twocat(Music$Jazz, Music$Age, nperm=100)

assoc.twocont Measures the association between two continuous variables

Description

Measures the association between two continuous variables with Pearson, Spearman and Kendall correlations.

Usage

assoc.twocont(x, y, weights=rep(1, length(x)),
         nperm=NULL, distrib="asympt")

Arguments

x a continuous variable (must be a numeric vector)
y a continuous variable (must be a numeric vector)
weights an optional numeric vector of weights (by default, a vector of 1 for uniform weights)
nperm numeric. Number of permutations for the permutation test of independence. If NULL (default), no permutation test is performed.
distrib the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").

Value

A data frame with Pearson, Spearman and Kendall correlations. The correlation value is in the first row and a p-value from a permutation (so non parametric) test of independence is in the second row.

Author(s)

Nicolas Robette

See Also

assoc.twocat, assoc.catcont, assoc.yx, condesc, catdesc, darma, ggassoc_scatter
Examples

## Assessment of tuna quality. We compare the Hunter L measure of lightness to the averages of consumer panel scores (recoded as integer values from 1 to 6 and averaged over 80 such values) in 9 lots of canned tuna.
x <- c(44.4, 45.9, 41.9, 53.3, 44.7, 44.1, 50.7, 45.2, 60.1)
y <- c(2.6, 3.1, 2.5, 5.0, 3.6, 4.0, 5.2, 2.8, 3.8)
assoc.twocont(x,y,nperm=100)

---

**assoc.yx**

Bivariate association measures between a response and predictor variables.

### Description

Computes bivariate association measures between a response and predictor variables (and, optionally, between every pairs of predictor variables.)

### Usage

```r
assoc.yx(y, x, weights=rep(1,length(y)), xx = TRUE, twocont="kendall", nperm=NULL, distrib="asympt", dec=c(3,3))
```

### Arguments

- **y**
  
  the response variable

- **x**

  the predictor variables

- **weights**

  an optional numeric vector of weights (by default, a vector of 1 for uniform weights)

- **xx**

  whether the association measures should be computed for couples of predictor variables (default) or not. With a lot of predictors, consider setting xx to FALSE (for reasons of computation time).

- **twocont**

  character. The type of measure of correlation measure to use between two continuous variables : "pearson", "spearman" or "kendall" (default).

- **nperm**

  numeric. Number of permutations for the permutation test of independence. If NULL (default), no permutation test is performed.

- **distrib**

  the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").

- **dec**

  vector of 2 integers for number of decimals. The first value if for association measures, the second for permutation p-values. Default is c(3,3).
Details

The function computes an association measure: Pearson’s, Spearman’s or Kendall’s correlation for pairs of numeric variables, Cramer’s V for pairs of factors and eta-squared for pairs numeric-factor. It can also compute the p-value of a permutation test of association for each pair of variables.

Value

A list of the following items:

- **YX**: a table with the association measures between the response and predictor variables
- **XX**: a table with the association measures between every pairs of predictor variables

In each table:

- **measure**: name of the association measure
- **association**: value of the association measure
- **permutation.pvalue**: p-value from the permutation test

Author(s)

Nicolas Robette

See Also

darma, assoc.twocat, assoc.twocont, assoc.catcont, condesc, catdesc

Examples

data(iris)
iris2 = iris
iris2$Species = factor(iris$Species == "versicolor")
assoc.yx(iris2$Species,iris2[,1:4],nperm=100)

burt (Computes a Burt table)

Description

Computes a Burt table from a data frame composed of categorical variables.

Usage

burt(data)

Arguments

data data frame with n rows (individuals) and p columns (categorical variables)
Details

A Burt table is a symmetric table that is used in correspondence analysis. It shows the frequencies for all combinations of categories of pairs of variables.

Value

Returns a square matrix. Its dimension is equal to the total number of categories in the data frame.

Author(s)

Nicolas Robette

References


See Also

dichotom

Examples

```r
# Burt table of variables in columns 1 to 5
# in the 'Music' example data set
data(Music)
burt(Music[,1:5])
```

---

**catdesc**

*Measures the association between a categorical variable and some continuous and/or categorical variables*

Description

Measures the association between a categorical variable and some continuous and/or categorical variables

Usage

```r
catdesc(y, x, weights=rep(1,length(y)), min.phi=NULL, robust=TRUE, nperm=NULL, distrib="asympt", dec=c(3,3,3,3,1,3))
```
**Arguments**

- **y** the categorical variable to describe (must be a factor)
- **x** a data frame with continuous and/or categorical variables
- **weights** an optional numeric vector of weights (by default, a vector of 1 for uniform weights)
- **min.phi** for the relationship between y and a categorical variable, only associations higher or equal to min.phi will be displayed. If NULL (default), they are all displayed.
- **robust** logical. If FALSE, mean and standard deviation are used instead of median and mad. Default is TRUE.
- **nperm** numeric. Number of permutations for the permutation test of independence. If NULL (default), no permutation test is performed.
- **distrib** the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").
- **dec** vector of 6 integers for number of decimals. The first value if for association measures, the second for permutation p-values, the third for percents, the fourth for phi coefficients, the fifth for medians and mads, the sixth for point biserial correlations. Default is c(3,3,3,3,1,3).

**Value**

A list of the following items:

- **variables** associations between y and the variables in x
- **bylevel** a list with one element for each level of y

Each element in bylevel has the following items:

- **categories** a data frame with categorical variables from x and associations measured by phi
- **continuous.var** a data frame with continuous variables from x and associations measured by correlation coefficients

**Author(s)**

Nicolas Robette

**References**


**See Also**

catdes, condesc, assoc.yx, darma

**Examples**

data(Movies)
catdesc(Movies$ArtHouse, Movies[,c("Budget","Genre","Country")])
**conc.ellipse**

_Adds concentration ellipses to a cloud of individuals._

**Description**

Adds inertia ellipses to the cloud of individuals of a MCA. By default, concentration ellipses are plotted.

**Usage**

```r
conc.ellipse(resmca, var, sel = 1:nlevels(var), axes = c(1, 2),
             kappa = 2,
             col = rainbow(length(sel)), pcol = rainbow(length(sel)), pcex = 0.2,
             lty = 1, lwd = 1, tcex = 1, text.lab = TRUE)
```

**Arguments**

- `resmca`: object of class 'MCA', 'speMCA', 'csMCA', 'multiMCA' or 'stMCA'
- `var`: supplementary variable to plot
- `sel`: numeric vector of indexes of the categories to plot (by default, ellipses are plotted for every categories)
- `axes`: length 2 vector specifying the components to plot (default is c(1,2))
- `kappa`: numeric. The kappa value (i.e. "index") of the inertia ellipses. By default, kappa = 2, which means that concentration ellipses are plotted.
- `col`: vector of colors for the ellipses of plotted categories (by default, rainbow palette is used)
- `pcol`: vector of colors for the points at the center of ellipses of plotted categories (by default, rainbow palette is used)
- `pcex`: numerical value giving the amount by which points at the center of ellipses should be magnified (default is 0.2)
- `lty`: line type for ellipses (default is 1)
- `lwd`: line width for the ellipses (default is 1)
- `tcex`: numerical value giving the amount by which labels at the center of ellipses should be magnified (default is 0.2)
- `text.lab`: whether the labels at the center of ellipses should be displayed (default is TRUE)

**Details**

If kappa=2, ellipses are called "concentration" ellipses and, for a normally shaped subcloud, contain 86.47 percents of the points of the subcloud. If kappa=1, ellipses are "indicator" ellipses and contain 39.35 percents of the points of the subcloud. If kappa=1.177, ellipses are "median" ellipses and contain 50 percents of the points of the subcloud. This function has to be used after the cloud of individuals has been drawn.
condesc

Author(s)
Nicolas Robette

References


See Also
plot.speMCA, plot.csMCA, plot.multiMCA, plot.stMCA

Examples

```r
## Performs specific MCA (excluding 'NA' categories) of 'Taste' example data set,
## plots the cloud of categories
## and adds concentration ellipses for gender variable
data(Taste)
mca <- speMCA(Taste[,1:11], excl=c(3,6,9,12,15,18,21,24,27,30,33))
plot(mca,type='i')
conc.ellipse(mca,Taste$Gender)

## Draws a blue concentration ellipse for men only
plot(mca,type='i')
conc.ellipse(mca,Taste$Gender, sel=1, col='blue')
```

condesc  Measures the association between a continuous variable and some continuous and/or categorical variables

Description
Measures the association between a continuous variable and some continuous and/or categorical variables

Usage

```r
condesc(y, x, weights=rep(1,length(y)), min.cor=NULL, robust=TRUE, nperm=NULL, distrib="asympt", dec=c(3,3,0,3))
```
condesc

Arguments

- **y**
  - the continuous variable to describe

- **x**
  - a data frame with continuous and/or categorical variables

- **weights**
  - an optional numeric vector of weights (by default, a vector of 1 for uniform weights)

- **min.cor**
  - for the relationship between y and a categorical variable, only associations higher or equal to min.cor will be displayed. If NULL (default), they are all displayed.

- **robust**
  - logical. If FALSE, mean and standard deviation are used instead of median and mad. Default is TRUE.

- **nperm**
  - numeric. Number of permutations for the permutation test of independence. If NULL (default), no permutation test is performed.

- **distrib**
  - the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").

- **dec**
  - vector of 4 integers for number of decimals. The first value if for association measures, the second for permutation p-values, the third for medians and mad, the fourth for point biserial correlations. Default is c(3,3,0,3).

Value

A list of the following items:

- **variables**
  - associations between y and the variables in x

- **categories**
  - a data frame with categorical variables from x and associations measured by point biserial correlation

Author(s)

Nicolas Robette

References


See Also

condes, catdesc, assoc.yx, darma

Examples

data(Movies)
condesc(Movies$BoxOffice, Movies[,c("Budget","Genre","Country")])
Computes contributions for a MCA

Description

From MCA results, computes contributions of categories and variables to the axes and the overall cloud.

Usage

contrib(resmca)

Arguments

resmca object of class 'MCA', 'speMCA', or 'csMCA'

Details

The contribution of a point to an axis depends both on the distance from the point to the origin point along the axis and on the weight of the point. The contributions of points to axes are the main aid to interpretation (see Le Roux and Rouanet, 2004 and 2010).

Value

A list of data frames:

ctr Data frame with the contributions of categories to axes
var.ctr Data frame with the contributions of variables to axes
ctr.cloud Data frame with the contributions of categories to the overall cloud
vctr.cloud Data frame with the contributions of variables to the overall cloud

Author(s)

Nicolas Robette

References


See Also

MCA, speMCA, csMCA, varsup
Examples

```r
## Performs a specific MCA on the 'Music' example data set
## and compute contributions
data(Music)
mca <- speMCA(Music[,1:5], excl=c(3,6,9,12,15))
contrib(mca)
```

### csMCA

**Performs a 'class specific' MCA**

#### Description

Performs a 'class specific' Multiple Correspondence Analysis, i.e. a variant of MCA consisting in analyzing a subcloud of individuals.

#### Usage

```r
csMCA(data, subcloud = rep(TRUE, times = nrow(data)), excl = NULL, ncp = 5,
      row.w = rep(1, times = nrow(data)))
```

#### Arguments

- `data` : data frame with n rows (individuals) and p columns (categorical variables)
- `subcloud` : a vector of logical values and length n. The subcloud of individuals analyzed with 'class specific' MCA is made of the individuals with value TRUE.
- `excl` : numeric vector indicating the indexes of the 'junk' categories (default is NULL). See `getindexcat` to identify these indexes.
- `ncp` : number of dimensions kept in the results (default is 5)
- `row.w` : an optional numeric vector of row weights (by default, a vector of 1 for uniform row weights)

#### Details

This variant of MCA is used to study a subset of individuals with reference to the whole set of individuals, i.e. to determine the specific features of the subset. It consists in proceeding to the search of the principal axes of the subcloud associated with the subset of individuals (see Le Roux and Rouanet, 2004 and 2010).

#### Value

Returns an object of class 'csMCA', i.e. a list including:

- `eig` : a list of vectors containing all the eigenvalues, the percentage of variance, the cumulative percentage of variance, the modified rates and the cumulative modified rates
- `call` : a list with informations about input data
ind  a list of matrices containing the results for the individuals (coordinates, contributions)

var a list of matrices containing all the results for the categories and variables (weights, coordinates, square cosine, categories contributions to axes and cloud, test values (v.test), square correlation ratio (eta2), variable contributions to axes and cloud

Author(s)
Nicolas Robette

References


See Also
getindexcat, plot.csMCA, varsup, contrib, modif.rate, dimdescr, speMCA, MCA

Examples
```r
define data
female <- Music$Gender == 'Women'
mca <- csMCA(Music[,1:5], subcloud=female, excl=c(3,6,9,12,15))
plot(mca)
```

Darima

Describes Associations as in a Regression Model Analysis.

Description
Computes bivariate association measures between a response and predictor variables, producing a summary looking like a regression analysis.

Usage
```r
darma(y, x, weights=rep(1,length(y)), target=1, twocont="kendall", nperm=NULL, distrib="asympt", dec=c(1,3,3))
```
Arguments

- **y**: the response variable
- **x**: the predictor variables
- **weights**: an optional numeric vector of weights (by default, a vector of 1 for uniform weights)
- **target**: rank or name of the category of interest when y is categorical
- **twocont**: character. The type of measure of correlation measure to use between two continuous variables: "pearson", "spearman" or "kendall" (default).
- **nperm**: numeric. Number of permutations for the permutation test of independence. If NULL (default), no permutation test is performed.
- **distrib**: the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").
- **dec**: vector of 3 integers for number of decimals. The first value if for percents or medians, the second for association measures, the third for permutation p-values. Default is c(1,3,3).

Details

The function computes association measures (phi, correlation coefficient, Kendall’s correlation) between the variable of interest and the other variables. It can also compute the p-values permutation tests.

Value

A data frame

Author(s)

Nicolas Robette

See Also

assoc.yx, assoc.twocat, assoc.twocont, assoc.catcont, condesc, catdesc

Examples

```r
data(iris)
iris2 = iris
iris2$Species = factor(iris$Species == "versicolor")
darma(iris2$Species, iris2[,1:4], target=2, nperm=100)
```
dichotom

Dichotomizes the variables in a data frame

Description

Dichotomizes the variables in a data frame exclusively composed of categorical variables

Usage

dichotom(data, out = "numeric")

Arguments

data: data frame of categorical variables
out: character string defining the format for dichotomized variables in the output data frame. Format may be 'numeric' or 'factor' (default is 'numeric').

Value

Returns a data frame with dichotomized variables. The number of columns is equal to the total number of categories in the input data.

Author(s)

Nicolas Robette, Julien Barnier

Examples

## Dichotomizes 'Music' example data frame
data(Music)
dic <- dichotom(Music[,1:5])
str(dic)

## with output variables in factor format
dic <- dichotom(Music[,1:5], out='factor')
str(dic)

dimcontrib

Describes the contributions to axes for a MCA

Description

Identifies the categories and individuals that contribute the most to each dimension obtained by a Multiple Correspondence Analysis. It allows to analyze variants of MCA, such as 'specific' MCA or 'class specific' MCA.
Usage

`dimcontrib(resmca, dim = c(1,2), best = TRUE)`

Arguments

- `resmca`: object of class 'MCA', 'speMCA', or 'csMCA'
- `dim`: dimensions to describe (default is c(1,2))
- `best`: if FALSE, displays all the categories; if TRUE (default), displays only categories and individuals with contributions higher than average

Details

Contributions are sorted and assigned a positive or negative sign according to the corresponding categories or individuals’ coordinates, so as to facilitate interpretation.

Value

Returns a list:

- `var`: a list of categories’ contributions to axes
- `ind`: a list of individuals’ contributions to axes

Author(s)

Nicolas Robette

References


See Also

`dindescr, dimeta2, dimtypicality, condes, speMCA, csMCA`

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then describes the contributions to axes.
data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
dimcontrib(mca)
```
dimdescr

Describes the dimensions of a MCA

Description

Identifies the variables and the categories that are the most characteristic according to each dimension obtained by a MCA. It is inspired by `dimdesc` function in `FactoMineR` package (see Husson et al, 2010), but allows to analyze variants of MCA, such as ‘specific’ MCA or ‘class specific’ MCA.

Usage

```r
dimdescr(resmca, vars=NULL, dim = c(1,2), min.cor = NULL, nperm = 100, distrib = "asympt")
```

Arguments

- `resmca`: object of class `MCA`, `speMCA`, `csMCA`, `stMCA` or `multiMCA`
- `vars`: data frame of variables to describes the MCA dimensions with. If NULL (default), the active variables of the MCA will be used.
- `dim`: the axes which are described. Default is c(1,2)
- `min.cor`: for the relationship between y and a categorical variable, only associations higher or equal to min.cor will be displayed. If NULL (default), they are all displayed.
- `nperm`: numeric. Number of permutations for the permutation test of independence. If NULL, no permutation test is performed.
- `distrib`: the null distribution of permutation test of independence can be approximated by its asymptotic distribution ("asympt", default) or via Monte Carlo resampling ("approx").

Details

See `condesc`.

Value

Returns a list of ncp lists including:

- `variables`: associations between y and the variables in x
- `categories`: a data frame with categorical variables from x and associations measured by correlation coefficients

Author(s)

Nicolas Robette
References

See Also
condesc, speMCA, csMCA, dimdesc

Examples
```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then describe the dimensions.
data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
dimdescr(mca,min.cor=0.1,nperm=10)
```

---

dimeta2

*Describes the eta2 of supplementary variables for the axes of a MCA*

Description
Computes eta2 for a list of supplementary variables. It allows to analyze variants of MCA, such as 'specific' MCA or 'class specific' MCA.

Usage
```r
dimeta2(resmca, vars, dim = c(1,2))
```

Arguments
- `resmca` object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `vars` a data frame of supplementary variables
- `dim` the axes for which eta2 are computed. Default is c(1,2)

Value
Returns a data frame with supplementary variables as rows and axes as columns.

Author(s)
Nicolas Robette
References


See Also

dimdesc, dimdescr, dimcontrib, condes, speMCA, csMCA, dimtypicality

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then describes the eta2 for Gender and Age (axes 1 and 2).
data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
dimeta2(mca, Music[,c("Gender","Age")])
```

---

**dimtypicality**

*Typicality tests for supplementary variables of a MCA*

Description

Computes typicality tests for a list of supplementary variables. It allows to analyze variants of MCA, such as 'specific' MCA or 'class specific' MCA.

Usage

```r
dimtypicality(resmca, vars, dim = c(1,2), max.pval=1)
```

Arguments

- `resmca`: object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `vars`: a data frame of supplementary variables
- `dim`: the axes for which typicality tests are computed. Default is c(1,2)
- `max.pval`: only categories with a p-value lower or equal to max.pval are displayed. By default, all categories are displayed

Value

Returns a list of data frames giving the test statistics and p-values of the supplementary categories for the different axes.
flip.mca

**Description**

Flips the coordinates of the individuals and the categories on one or more dimensions of a MCA.

**Usage**

```r
flip.mca(resmca, dim=1)
```

**Arguments**

- `resmca`: object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `dim`: the dimensions for which the coordinates are flipped. By default, only the 1st dimension is flipped

**Value**

Returns an object of the same class as `resmca`

**Author(s)**

Nicolas Robette
getindexcat

References


See Also

MCA, speMCA, csMCA

Examples

```r
## Flips dimensions 1 and 2
## of the MCA of 'Music' example data set
data(Music)
mca <- speMCA(Music[,1:5])
ggcloud_variables(mca, legend="none")
flipped_mca <- flip.mca(mca, dim=c(1,2))
ggcloud_variables(flipped_mca, legend="none")
```

Description

Returns the names of the categories in a data frame exclusively composed of categorical variables.

Usage

```r
getindexcat(data)
```

Arguments

data data frame of categorical variables

Details

This function may be useful prior to a 'specific' MCA, to identify the indexes of the 'junk' categories to exclude.

Value

Returns a character vector with the names of the categories of the variables in the data frame

Author(s)

Nicolas Robette
See Also

speMCA, csMCA

Examples

data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))

Description

Add attractions between categories, as measured by phi coefficients or percentages of maximum deviation (PEM), by plotting segments onto a MCA cloud of variables, using the ggplot2 framework.

Usage

ggadd_attractions(p, resmca, axes=c(1,2), measure="phi", min.asso=0.3, col.segment="lightgray", col.text="black", text.size=3)

Arguments

p ggplot object with the cloud of variables
resmca object of class MCA, speMCA, csMCA, stMCA or multiMCA.
axes numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
measure character. The measure for attractions: "phi" (default) for phi coefficients, "pem" for percentages of maximum deviation (PEM).
min.asso numerical value between 0 and 1. The minimal attraction value for segments to be plotted. Default is 0.3.
col.segment Character. Color of the segments. Default is lightgray.
col.text Character. Color of the labels of the categories. Default is black.
text.size Size of the labels of categories. Default is 3.

Value

a ggplot object

Author(s)

Nicolas Robette
References


See Also

ggcloud_variables, ggcloud_indiv, ggadd_ellipses, ggadd_corr, ggadd_interaction, ggadd_density

Examples

```r
## Performs a specific MCA on 'Taste' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## draws the cloud of categories
## and adds segments between categories with attractions.
data(Taste)
getindexcat(Taste[,1:11])
mca <- speMCA(Taste[,1:11], excl=c(3,6,9,12,15,18,21,24,27,30,33))
p <- ggcloud_variables(mca, col="white", legend="none")
ggadd_attractions(p, mca, measure="phi", min.assoc=0.1)
```

---

**ggadd_chulls**

Adds convex hulls to a cloud of individuals

Description

Adds convex hulls for a categorical variable to a MCA cloud of individuals, using the ggplot2 framework.

Usage

```r
ggadd_chulls(p, resmca, var, sel=1:nlevels(var), axes=c(1,2), col=NULL, alpha=0.2, label=TRUE, label.size=5, legend="right")
```

Arguments

- `p`: ggplot object with the cloud of variables
- `resmca`: object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `var`: Factor. The categorical variable used to plot ellipses.
- `sel`: numeric vector of indexes of the categories to plot (by default, ellipses are plotted for every categories)
- `axes`: numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
- `col`: Colors for the ellipses and labels of the categories. Can be the name of a palette from the RcolorBrewer package, ‘bw’ for a black and white palette (uses scale_color_grey()), a character vector of colors for a custom palette, or the name of a color for a single color. If NULL (default), the default palette of ggplot2 is used.
alpha  Numerical value from 0 to 1. Transparency of the polygon’s fill. Default is 0.2
label Logical. Should the labels of the categories be plotted at the center of ellipses? Default is TRUE.
label.size Size of the labels of the categories at the center of ellipses. Default is 5.
legend the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector). Default is right.

Value

a ggplot object

Author(s)

Nicolas Robette

References


See Also

`ggcloud_variables, ggcloud_indiv, ggadd_supvar, ggadd_corr, ggadd_interaction, ggadd_density, ggadd_kellipses, ggadd_ellipses`

Examples

```r
## Performs a specific MCA on 'Taste' example data set
## ignoring every "NA" (i.e. 'not available') categories,
## then performs hierarchical clustering and
## partitions the individuals into 3 clusters,
## draws the cloud of individuals
## and adds convex hulls for the clusters.

data(Taste)
getindexcat(Taste)
mca <- speMCA(Taste[,1:11],excl=c(3,6,9,12,15,18,21,24,27,30,33))
d <- dist(mca$ind$coord[,c(1,2)])
hca <- hclust(d, "ward.D2")
cluster <- factor(cutree(hca, 3))
p <- ggcloud_indiv(mca, col="black")
ggadd_chulls(p, mca, cluster)
```
**ggadd_corr**

Adds a heatmap of under/over-representation of a supplementary variable to a cloud of individuals

**Description**

Adds a heatmap representing the correlation coefficients to a MCA cloud of individuals, for a numerical supplementary variable or one category of a categorical supplementary variable, using the ggplot2 framework.

**Usage**

```r
ggadd_corr(p, resmca, var, cat=levels(var)[1], axes=c(1,2),
xbins=20, ybins=20, min.n=1, pal="RdYlBu", limits=NULL, legend="right")
```

**Arguments**

- `p`: ggplot object with the cloud of variables
- `resmca`: object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `var`: factor or numerical vector. The supplementary variable used for the heatmap.
- `cat`: character string. numeric vector of indexes of the categories to plot (by default, ellipses are plotted for every categories). Only used if var is a factor.
- `axes`: numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
- `xbins`: integer. Number of bins in the x axis. Default is 20.
- `ybins`: integer. Number of bins in the y axis. Default is 20.
- `min.n`: integer. Minimal number of points for a tile to be drawn. By default, every tiles are drawn.
- `pal`: character string. Name of a diverging ColorBrewer palette. Default is "RdYlBu".
- `limits`: numerical vector of length 2. Lower and upper limits of the correlation coefficients for the color scale. Should be centered around 0 for a better view of under/over-representations (for example c(-0.2,0.2)). By default, the maximal absolute value of the correlation coefficients is used.
- `legend`: the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector). Default is right.

**Details**

For each tile of the heatmap, a correlation coefficient is computed between the supplementary variable and the fact of belonging to the tile. This gives a view of the under/over-representation of the supplementary variable according to the position in the cloud of individuals.

**Value**

a ggplot object
Author(s)

Nicolas Robette

References


See Also

`ggcloud_variables`, `ggcloud_indiv`, `ggadd_supvar`, `ggadd_interaction`, `ggadd_ellipses`, `ggadd_density`

Examples

```r
## Performs a specific MCA on 'Taste' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## draws the cloud of categories
## and adds a correlation heatmap for Age=50+.
data(Taste)
getindexcat(Taste[,1:11])
mca <- speMCA(Taste[,1:11],excl=c(3,6,9,12,15,18,21,24,27,30,33))
p <- ggcloud_indiv(mca, col="lightgrey")
ggadd_corr(p, mca, var=Taste$Age, cat="50+", xbins=10, ybins=10)
```

---

**ggadd_density**

*Adds a density layer to the cloud of individuals for a category of a supplementary variable*

Description

For a given category of a supplementary variable, adds a layer representing the density of points to the cloud of individuals, either with contours or areas.

Usage

```r
ggadd_density(p, resmca, var, cat=levels(var)[1], axes=c(1,2),
density="contour", col.contour="darkred", pal.area="viridis", alpha.area=0.2,
ellipse=FALSE, col.ellipse="black")
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>p</code></td>
<td>ggplot object with the cloud of variables</td>
</tr>
<tr>
<td><code>resmca</code></td>
<td>object of class MCA, speMCA, csMCA, stMCA or multiMCA</td>
</tr>
<tr>
<td><code>var</code></td>
<td>factor or numerical vector. The supplementary variable used for the heatmap.</td>
</tr>
<tr>
<td><code>cat</code></td>
<td>character string. numeric vector of indexes of the categories to plot (by default, ellipses are plotted for every categories)</td>
</tr>
<tr>
<td><code>axes</code></td>
<td>numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).</td>
</tr>
<tr>
<td><code>density</code></td>
<td>If &quot;contour&quot;, density is plotted with contours. If &quot;area&quot;, density is plotted with areas.</td>
</tr>
<tr>
<td><code>col.contour</code></td>
<td>character string. The color of the contours.</td>
</tr>
<tr>
<td><code>pal.area</code></td>
<td>character string. The name of a viridis palette for areas.</td>
</tr>
<tr>
<td><code>alpha.area</code></td>
<td>numeric. Transparency of the areas. Default is 0.2.</td>
</tr>
<tr>
<td><code>ellipse</code></td>
<td>logical. If TRUE, a concentration ellipse is added.</td>
</tr>
<tr>
<td><code>col.ellipse</code></td>
<td>character string. The color of the ellipse. Default is black.</td>
</tr>
</tbody>
</table>

Value

a ggplot object

Author(s)

Nicolas Robette

References


See Also

`ggcloud_variables`, `ggcloud_indiv`, `ggadd_supvar`, `ggadd_interaction`, `ggadd_ellipses`, `ggadd_corr`

Examples

```r
## Performs a specific MCA on 'Taste' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## draws the cloud of categories
## and adds a density layer for Age=50+.
data(Taste)
getindexcat(Taste[,1:11])
mca <- speMCA(Taste[,1:11],excl=c(3,6,9,12,15,18,21,24,27,30,33))
p <- ggcloud_indiv(mca, col='lightgrey')
```
```r
# density layer with contours
ggadd_density(p, mca, var=Taste$Age, cat="50+")

# density layer with areas
ggadd_density(p, mca, var=Taste$Age, cat="50+", density="area")
```

---

**ggadd_ellipses**

* Adds confidence ellipses to a cloud of individuals

**Description**

Adds confidence ellipses for a categorical variable to a MCA cloud of individuals, using the ggplot2 framework.

**Usage**

```r
ggadd_ellipses(p, resmca, var, sel=1:nlevels(var), axes=c(1,2),
level=0.05, label=TRUE, label.size=3, col=NULL, size=0.5, points=TRUE, legend='right')
```

**Arguments**

- **p**: ggplot object with the cloud of variables
- **resmca**: object of class `MCA`, `speMCA`, `csMCA`, `stMCA` or `multiMCA`
- **var**: Factor. The categorical variable used to plot ellipses.
- **sel**: numeric vector of indexes of the categories to plot (by default, ellipses are plotted for every categories)
- **axes**: numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
- **level**: The level at which to draw an ellipse (see `stat_ellipse`). Default is 0.05, which means 95 percents confidence ellipses are plotted.
- **label**: Logical. Should the labels of the categories be plotted at the center of ellipses ? Default is TRUE.
- **label.size**: Size of the labels of the categories at the center of ellipses. Default is 3.
- **col**: Colors for the ellipses and labels of the categories. Can be the name of a palette from the RcolorBrewer package, 'bw' for a black and white palette (uses `scale_color_grey()`), a character vector of colors for a custom palette, or the name of a color for a single color. If NULL (default), the default palette of ggplot2 is used.
- **size**: Size of the lines of the ellipses. Default is 0.5.
- **points**: If TRUE (default), the points are coloured according to their subcloud.
- **legend**: the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector). Default is right.
Details

A confidence ellipse aims at measuring how the "true" mean point of a category differs from its
observed mean point. This is achieved by constructing a confidence zone around the observed
mean point. If we choose a conventional level alpha (e.g. 0.05), a (1 - alpha) (e.g. 95 percents)
confidence zone is defined as the set of possible mean points that are not significantly different from
the observed mean point.

Value

a ggplot object

Author(s)

Nicolas Robette

References

Le Roux B. and Rouanet H., *Multiple Correspondence Analysis*, SAGE, Series: Quantitative Ap-

Le Roux B. and Rouanet H., *Geometric Data Analysis: From Correspondence Analysis to Struc-

See Also

`ggcloud_variables, ggcloud_indiv, ggadd_supvar, ggadd_corr, ggadd_interaction, ggadd_density, ggadd_kellipses`

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## draws the cloud of categories
## and adds confidence ellipses for Age.

data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
p <- ggcloud_indiv(mca, col='lightgrey')
ggadd_ellipses(p, mca, Music$Age)
```

---

**ggadd_interaction**

Adds the interaction between two categorical supplementary variables
to a cloud of variables

Description

Adds the interaction between two categorical supplementary variables to a MCA cloud of variables,
using the ggplot2 framework.
ggadd_interaction

Usage

```
ggadd_interaction(p, resmca, v1, v2, sel1=1:nlevels(v1), sel2=1:nlevels(v2),
axes=c(1,2), col=NULL, textsize=5, legend='right')
```

Arguments

- `p`  ggplot object with the cloud of variables
- `resmca` object of class MCA, speMCA, csMCA, stMCA or multiMCA.
- `v1`  Factor. The first categorical supplementary variable.
- `v2`  Factor. The second categorical supplementary variable.
- `sel1` Numeric vector of indexes of the categories of the first supplementary variable to be used in interaction. By default, every categories are used.
- `sel2` Numeric vector of indexes of the categories of the second supplementary variable to be used in interaction. By default, every categories are used.
- `axes` numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
- `col`  Colors for the labels of the categories. Can be the name of a palette from the RcolorBrewer package, 'bw' for a black and white palette (uses scale_color_grey()), a character vector of colors for a custom palette, or the name of a color for a single color. If NULL (default), the default palette of ggplot2 is used.
- `textsize` Size of the labels of categories. Default is 5.
- `legend` the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector). Default is right.

Value

a ggplot object

Author(s)

Nicolas Robette

References


See Also

`ggcloud_variables`, `ggadd_supvar`, `ggcloud_indiv`, `ggadd_corr`, `ggadd_ellipses`, `ggadd_density`
Examples

```r
## Performs a specific MCA on 'Taste' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## draws the cloud of categories
## and adds the interaction between Gender and Age.
data(Taste)
getindexcat(Taste[,1:11])
mca <- speMCA(Taste[,1:11], excl=c(3,6,9,12,15,18,21,24,27,30,33))
p <- ggcloud_variables(mca, col='lightgrey', shapes=FALSE)
ggadd_interaction(p, mca, Taste$Gender, Taste$Age)
```

---

**ggadd_kellipses**  
Adds k-inertia ellipses to a cloud of individuals

**Description**

Adds k-inertia ellipses for a categorical variable to a MCA cloud of individuals, using the ggplot2 framework. By default, concentration ellipses are plotted.

**Usage**

```r
ggadd_kellipses(p, resmca, var, sel=1:nlevels(var), axes=c(1,2),
kappa=2, label=TRUE, label.size=3, col=NULL, size=0.5, points=TRUE, legend='right')
```

**Arguments**

- `p`: ggplot object with the cloud of variables
- `resmca`: object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `var`: Factor. The categorical variable used to plot ellipses.
- `sel`: numeric vector of indexes of the categories to plot (by default, ellipses are plotted for every categories)
- `axes`: numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
- `kappa`: numeric. The kappa value (i.e. "index") of the inertia ellipses. By default, kappa = 2, which means that concentration ellipses are plotted.
- `label`: Logical. Should the labels of the categories be plotted at the center of ellipses ? Default is TRUE.
- `label.size`: Size of the labels of the categories at the center of ellipses. Default is 3.
- `col`: Colors for the ellipses and labels of the categories. Can be the name of a palette from the RColorBrewer package, 'bw' for a black and white palette (uses scale_color_grey()), a character vector of colors for a custom palette, or the name of a color for a single color. If NULL (default), the default palette of ggplot2 is used.
- `size`: Size of the lines of the ellipses. Default is 0.5.
- `points`: If TRUE (default), the points are coloured according to their subcloud.
- `legend`: the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector). Default is right.
Details

If kappa=2, ellipses are called "concentration" ellipses and, for a normally shaped subcloud, contain 86.47 percents of the points of the subcloud. If kappa=1, ellipses are "indicator" ellipses and contain 39.35 percents of the points of the subcloud. If kappa=1.177, ellipses are "median" ellipses and contain 50 percents of the points of the subcloud. This function has to be used after the cloud of individuals has been drawn.

Value

a ggplot object

Author(s)

Nicolas Robette

References


See Also

`ggcloud_variables`, `ggcloud_indiv`, `ggadd_supvar`, `ggadd_corr`, `ggadd_interaction`, `ggadd_density`, `ggadd_ellipses`

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## draws the cloud of categories
## and adds concentration ellipses for Age.

data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5], excl=c(3,6,9,12,15))
p <- ggcloud_indiv(mca, col="lightgrey")
ggadd_kellipses(p, mca, Music$Age)
```

---

**ggadd_supind**

Adds supplementary individuals to a cloud of individuals

Description

Adds supplementary individuals to a MCA cloud of the individuals, using the ggplot2 framework.

Usage

`ggadd_supind(p, resmca, dfsup, axes=c(1,2), col="black", textsize=5, pointsize=2)`
**ggadd_supvar**

**Arguments**

- **p**
  - ggplot object with the cloud of variables
- **resmca**
  - object of class MCA, speMCA, csMCA, stMCA or multiMCA.
- **dfsup**
  - data frame with the supplementary individuals. It must have the same factors as the data frame used as input for the initial MCA.
- **axes**
  - numeric vector of length 2, specifying the dimensions (axes) to plot (default is c(1,2))
- **col**
  - color for the labels and points of the individuals (default is black)
- **textsize**
  - Size of the labels of the individuals. Default is 5.
- **pointsize**
  - Size of the points of the individuals. If NULL, only labels are plotted. Default is 2.

**Details**

The function uses the row names of **dfsup** as labels for the individuals.

**Author(s)**

Nicolas Robette

**See Also**

indsup, ggcloud_indiv

**Examples**

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## plots the cloud of individuals,
## and then adds individuals 1, 20 and 300 as supplementary individuals.
data(Music)ownames(Music) <- paste0("i", 1:nrow(Music))
mca <- speMCA(Music[,1:5], excl=c(3,6,9,12,15))
p <- ggcloud_indiv(mca, col="lightgrey")
ggadd_supind(p, mca, Music[c(1,20,300),1:5])
```

---

**ggadd_supvar**

_Adds a categorical supplementary variable to a cloud of variables_

**Description**

Adds a categorical supplementary variable to a MCA cloud of variables, using the ggplot2 framework.
Usage

ggadd_supvar(p, resmca, var, sel=1:nlevels(var), axes=c(1,2), col='black', shape=1, prop=NULL, textsize=3, shapesize=6, segment=FALSE, vname=NULL)

Arguments

p ggplot object with the cloud of variables
resmca object of class MCA, speMCA, csMCA, stMCA or multiMCA.
var Factor. The categorical supplementary variable. It does not need to have been used at the MCA step.
sel Numeric vector of indexes of the categories of the supplementary variable to be added to the plot. By default, labels are plotted for every categories.
axes numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
col Character. Color of the shapes and labels of the categories. Default is black.
shape Symbol to be used in addition the the labels of categories (default is 1). If NULL, only labels are plotted.
prop If NULL, the size of the labels (if shape=NULL) or the shapes (otherwise) is constant. If ‘n’, the size is proportional the the weights of categories; if ‘vtest1’, the size is proportional to the test values of the categories on the first dimension of the plot; if ‘vtest2’, the size is proportional to the test values of the categories on the second dimension of the plot; if ‘cos1’, the size is proportional to the cosines of the categories on the first dimension of the plot; if ‘cos2’, the size is proportional to the cosines of the categories on the second dimension of the plot; if ‘cos12’, the size is proportional to the total cosines of the categories on the two dimensions of the plot.
textsize Size of the labels of categories if shape is not NULL, or if shape=NULL and prop=NULL. Default is 3.
shapesize Size of the shapes if prop=NULL, maximum size of the shapes in other cases. Default is 6.
segment Logical. Should one add lines between categories ? Default is FALSE.
vname A character string to be used as a prefix for the labels of the categories. If NULL (default), no prefix is added.

Value

a ggplot object

Author(s)

Nicolas Robette
References


See Also

`ggcloud_variables`, `ggcloud_indiv`, `ggadd_ellipses`, `ggadd_corr`, `ggadd_interaction`, `ggadd_density`

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## draws the cloud of categories
## and adds Age as a supplementary variable.

data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5], excl=c(3,6,9,12,15))
p <- ggcloud_variables(mca, palette='lightgrey', shapes=FALSE)
ggadd_supvar(p, mca, Music$Age, segment=TRUE)
```

---

**ggassoc_boxplot**  
*Boxplots with violins*

**Description**

Displays of boxplot and combines it with a violin plot, using ggplot2.

**Usage**

```r
ggassoc_boxplot(data, mapping, axes.labs=TRUE, ticks.labs=TRUE, text.size=3, box=TRUE, notch=FALSE, violin=TRUE)
```

**Arguments**

- `data`: dataset to use for plot
- `mapping`: aesthetic being used. It must specify x and y.
- `axes.labs`: Whether to display the labels of the axes, i.e. the names of x and y. Default is TRUE.
- `ticks.labs`: Whether to display the labels of the categories of x and y. Default is TRUE.
- `text.size`: Size of the association measure. If NULL, the text is not added to the plot.
- `box`: Whether to draw boxplots. Default is TRUE.
- `notch`: If FALSE (default) make a standard box plot. If TRUE, make a notched box plot. Notches are used to compare groups; if the notches of two boxes do not overlap, this suggests that the medians are significantly different.
- `violin`: Whether to draw a violin plot. Default is TRUE.
ggassoc_crosstab

Details
Eta-squared measure of global association between x and y is displayed in upper-left corner of the plot.
This function can be used as a high-level plot with ggduo and ggpairs functions of the GGally package.

Value
a ggplot object

Author(s)
Nicolas Robette

See Also
assoc.catcont, condesc, assoc.yx, darma, ggpairs

Examples
data(decathlon, package="FactoMineR")
ggassoc_boxplot(decathlon, mapping = ggplot2::aes(x = Javeline, y = Competition))

ggassoc_crosstab
Plots counts and associations of a crosstabulation

Description
For a cross-tabulation, plots the number of observations by using rectangles with proportional areas, and the phi measures of association between the categories with a diverging gradient of colour, using ggplot2.

Usage
ggassoc_crosstab(data, mapping, max.phi=.8, sort="none", axes.labs=TRUE, ticks.labs=TRUE, text.size=3)

Arguments
data dataset to use for plot
mapping aesthetic being used. It must specify x and y.
max.phi numeric value, specifying the lower and upper limit of the scale for the color gradient filling the rectangles, i.e. for the phi measures of association. It corresponds to the maximum absolute value of phi one wants to represent in the plot.
sort
character. If "both", rows and columns are sorted according to the first factor of a correspondence analysis of the contingency table. If "x", only rows are sorted. If "y", only columns are sorted. If "none" (default), no sorting is done.

axes.labs
Whether to display the labels of the axes, i.e. the names of x and y. Default is TRUE.

ticks.labs
Whether to display the labels of the categories of x and y. Default is TRUE.

text.size
Size of the association measure. If NULL, the text is not added to the plot.

Details
The phi measure of association measures how much each combination of categories of x and y is over/under-represented.

Cramer's V measure of global association between x and y is displayed in upper-left corner of the plot.

This function can be used as a high-level plot with ggduo and ggpairs functions of the GGally package.

Value
a ggplot object

Author(s)
Nicolas Robette

See Also
assoc.twocat, phi.table, catdesc, assoc.yx, darma, ggassoc_phiplot, ggpairs

Examples
data(Taste)


ggassoc_crosstab(data=Taste, mapping=ggplot2::aes(x=Age, y=Classical))

---

**ggassoc_phiplot**

*Bar plot of phi measures of association of a crosstabulation*

Description
For a cross-tabulation, plots the phi measures of association with bars of varying height, using ggplot2.

Usage

```
ggassoc_phiplot(data, mapping, measure="phi", max.asso=NULL, sort="none", axes.labs=TRUE, ticks.labs=TRUE, text.size=3)```

---
Arguments

- **data**: dataset to use for plot
- **mapping**: aesthetic being used. It must specify x and y.
- **measure**: character. The measure for attractions: "phi" (default) for phi coefficients, "pem" for percentages of maximum deviation (PEM).
- **max.asso**: numeric value, specifying the lower and upper limit of the scale for the color gradient filling the rectangles, i.e. for the measures of association. It corresponds to the maximum absolute value of association one wants to represent in the plot. If NULL (default), the limit is automatically adjusted to the data.
- **sort**: character. If "both", rows and columns are sorted according to the first factor of a correspondence analysis of the contingency table. If "x", only rows are sorted. If "y", only columns are sorted. If "none" (default), no sorting is done.
- **axes.labs**: Whether to display the labels of the axes, i.e. the names of x and y. Default is TRUE.
- **ticks.labs**: Whether to display the labels of the categories of x and y. Default is TRUE.
- **text.size**: Size of the association measure. If NULL, the text is not added to the plot.

Details

The phi measure of association measures how much each combination of categories of x and y is over/under-represented. The bars vary in width according to the number of observations in the categories of the x variables. They vary in height according to the phi measure of association. Bars are black if phi is positive and white if phi is negative.

Cramer's V measure of global association between x and y is displayed in upper-left corner of the plot.

This function can be used as a high-level plot with ggduo and ggpairs functions of the GGally package.

Value

a ggplot object

Author(s)

Nicolas Robette

References


See Also

assoc.twocat, phi.table, catdesc, assoc.yx, darma, ggassoc_crosstab, ggpairs
Examples

```r
data(Taste)
ggassoc_phiplot(data=Taste, mapping=ggplot2::aes(x=Age, y=Classical))
```

---

**ggassoc_scatter**

*Scatter plot with a smoothing line*

Description

Displays of scatter plot and adds a smoothing line, using ggplot2.

Usage

```r
ggassoc_scatter(data, mapping, axes.labs=TRUE, ticks.labs=TRUE, text.size=3)
```

Arguments

- `data` dataset to use for plot
- `mapping` aesthetic being used. It must specify x and y.
- `axes.labs` Whether to display the labels of the axes, i.e. the names of x and y. Default is TRUE.
- `ticks.labs` Whether to display the labels of the categories of x and y. Default is TRUE.
- `text.size` Size of the association measure. If NULL, the text is not added to the plot.

Details

Kendall’s tau rank correlation between x and y is displayed in upper-left corner of the plot. Smoothing is performed with gam.

This function can be used as a high-level plot with `ggduo` and `ggpairs` functions of the `GGally` package.

Value

a ggplot object

Author(s)

Nicolas Robette

See Also

`assoc.twocont, condesc, assoc.yx, darma, ggpairs`

Examples

```r
data(decathlon, package="FactoMineR")
ggassoc_scatter(decathlon, mapping = ggplot2::aes(x = Javeline, y = Points))
```
Plots a Multiple Correspondence Analysis cloud of individuals, using ggplots functions.

Usage

ggcloud_indiv(resmca, type='i', points='all', axes=1:2, col='dodgerblue4', palette='Set2', size = 0.5, alpha=0.6, repel=FALSE, density=NULL, col.contour="darkred", hex.bins=50, hex.pal="viridis")

Arguments

resmca object of class MCA, speMCA, csMCA, stMCA or multiMCA
type If 'i', points are plotted; if 'inames', labels of individuals are plotted.
points character string. If 'all' all points are plotted (default); if 'besh' only those who contribute most to horizontal axis are plotted; if 'bestv' only those who contribute most to vertical axis are plotted; if 'best' only those who contribute most to horizontal or vertical axis are plotted.
axes numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
col If a factor, points or labels are colored according to their category regarding this factor. If a string with color name, every points or labels have the same color. Default is 'dodgerblue4'.
palette If col is a factor, can be the name of a RColorBrewer palette, 'bw' for a black and white palette (uses scale_color_grey()) or a character vector with a custom palette. If NULL, ggplot2 palette is used. Default is 'Set2'.
size Size of the points or labels of individuals. Default is 0.5.
alpha Transparency of the points or labels of individuals. Default is 0.6.
repel Logical. When type='inames', should labels of individuals be repelled? Default is FALSE.
density If NULL (default), no density layer is added. If "contour", density is plotted with contours. If "hex", density is plotted with hexagon bins.
col.contour character string. The color of the contours. Only used if density="contour".
hex.bins integer. The number of bins in both vertical and horizontal directions. Only used if density="hex".
hex.pal character string. The name of a viridis palette for hexagon bins. Only used if density="hex".
Details

Sometimes the dots are too many and overlap. It is then difficult to get an accurate idea of the distribution of the cloud of individuals. The "density" option allows you to add an additional layer to represent the density of points in the plane, in the form of contours or hexagonal areas.

Value

a ggplot object

Author(s)

Anton Perdoncin, Nicolas Robette

References


See Also

`ggcloud_variables`, `ggadd_ellipses`, `ggadd_supvar`, `ggadd_corr`, `ggadd_interaction`, `ggadd_density`

Examples

```r
## Performs a specific MCA on 'Taste' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then draws the cloud of individuals.
data(Taste)
getindexcat(Taste[,1:11])
mca <- speMCA(Taste[,1:11], excl=c(3,6,9,12,15,18,21,24,27,30,33))
ggcloud_indiv(mca)

# Points are colored according to gender
ggcloud_indiv(mca, col=Taste$Gender, palette='Set2')

# A density layer of contours is added
ggcloud_indiv(mca, density="contour")

# A density layer of hexagon bins is added
ggcloud_indiv(mca, density="hex", hex.bin=10)
```
Plots MCA cloud of variables with ggplot2

Description
Plots a Multiple Correspondence Analysis cloud of variables, using ggplots functions.

Usage

```r
ggcloud_variables(resmca, axes=c(1,2), points='all',
min.ctr=NULL, max.pval=0.01, face="pp",
shapes=TRUE, prop=NULL, textsize=3, shapesize=3,
col=NULL, palette=NULL, col.by.group=TRUE, alpha=1,
segment.alpha=0.5, vlab=TRUE, sep='.', legend='right')
```

Arguments

- **resmca**: object of class MCA, speMCA, csMCA, stMCA or multiMCA.
- **axes**: numeric vector of length 2, specifying the components (axes) to plot. Default is c(1,2).
- **points**: character string. If 'all' all categories are plotted (default); if 'besth' only those who contribute most to horizontal axis are plotted; if 'bestv' only those who contribute most to vertical axis are plotted; if 'best' only those who contribute most to horizontal or vertical axis are plotted.
- **min.ctr**: Numerical value between 0 and 100. The minimum contribution (in percent) for a category to be displayed if the `points` argument is equal to "best", "besth" or "bestv" and `resmca` is of type MCA, speMCA or csMCA. If NULL (default), only the categories that contribute more than the average (i.e. 100 / number of modalities) are displayed.
- **max.pval**: Numerical value between 0 and 100. The maximal p-value derived from test-values for a category to be displayed if the `points` argument is equal to "best", "besth" or "bestv" and `resmca` is of type stMCA or multiMCA.
- **face**: character string. Changes the face of the category labels when their contribution is greater than `min.ctr`. The first letter refers to the first represented axis, the second letter to the second. "p" is for plain text, "u" for underlined, "i" for italic and "b" for bold. For example, "ui" means that the labels of the most contributing categories on the first axis will be underlined and the labels of the most contributing categories on the second axis will be italicized. By default ("pp"), no font face change is made.
- **shapes**: Logical. Should shapes be plotted for categories (in addition to labels) ? Default is TRUE.
- **prop**: If NULL, the size of the labels (if `shapes`=FALSE) or the shapes (if `shapes`=TRUE) is constant. If 'n', the size is proportional the the weights of categories; if 'ctr1',
the size is proportional to the contributions of the categories on the first dimension of the plot; if ’ctr2’, the size is proportional to the contributions of the categories on the second dimension of the plot; if ’ctr.cloud’, the size is proportional to the total contributions of the categories on the whole cloud; if ’cos1’, the size is proportional to the cosines of the categories on the first dimension of the plot; if ’cos2’, the size is proportional to the cosines of the categories on the second dimension of the plot; if ’cos12’, the size is proportional to the total cosines of the categories on the two dimensions of the plot; if ’vtest1’, the size is proportional to the test-values of the categories on the first dimension of the plot; if ’vtest2’, the size is proportional to the test-values of the categories on the second dimension of the plot.

textsize
Size of the labels of categories if shapes=TRUE, or if shapes=FALSE and prop=NULL. Default is 3.

shapesize
Size if the shapes of categories if shapes=TRUE and prop=FALSE. Default is 3.

col
Character. A unique color for the shapes and labels of the categories. Default is NULL, which means a palette will be used instead of a unique color (see palette argument).

palette
Character string or character vector. Only used if col is NULL. Colors for the shapes and labels of the categories, differentiated by variable. Can be the name of a palette from the RColorBrewer package, "bw" for a black and white palette (uses scale_color_grey()) or a character vector of colors for a custom palette. If NULL (default), the default palette of ggplot2 is used.

col.by.group
Logical. If resmca is of type multimCA, categories are colored by group from the MFA if TRUE (default) and by variable if FALSE.

alpha
Transparency of the shapes and labels of categories. Default is 1.

segment.alpha
Transparency of the line segment beside labels of categories. Default is 0.5.

vlab
Logical. Should the variable names be used as a prefix for the labels of the categories. Default is TRUE.

sep
Character string used as a separator if vlab=TRUE.

legend
the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector). Default is right.

Value
a ggplot object

Author(s)
Anton Perdoncin, Nicolas Robette

References

### homog.test

**Computes a homogeneity test for a categorical supplementary variable**

**Description**

From MCA results, computes a homogeneity test for a categorical supplementary variable, i.e. characterizes the homogeneity of several subclouds.

**Usage**

```r
homog.test(resmca, var, dim=c(1,2))
```

**Arguments**

- `resmca` object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `var` the categorical supplementary variable. It does not need to have been used at the MCA step.
- `dim` the axes which are described. Default is c(1,2)

**Value**

Returns a list of lists, one for each selected dimension in the MCA. Each list has 2 elements:

- `test.stat` The square matrix of test statistics
- `p.values` The square matrix of p.values

**Author(s)**

Nicolas Robette
References


See Also

speMCA, csMCA, stMCA, multiMCA, textvarsup

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then computes a homogeneity test for age supplementary variable.
data(Music)
getindexcat(Music)
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
homog.test(mca,Music$Age)
```

--

**indsup**

*Computes statistics for supplementary individuals*

Description

From MCA results, computes statistics (coordinates, squared cosines) for supplementary individuals.

Usage

```r
indsup(resmca, supdata)
```

Arguments

- `resmca` object of class 'MCA', 'speMCA' or 'csMCA'
- `supdata` data frame with the supplementary individuals. It must have the same factors as the data frame used as input for the initial MCA.

Value

Returns a list:

- `coord` matrix of individuals’ coordinates
- `cos2` matrix of individuals’ square cosines

Author(s)

Nicolas Robette
medoids

Computes the medoids of clusters

Description

Computes the medoids of a cluster solution.

Usage

medoids(D, cl)

Arguments

D square distance matrix (n rows * n columns, i.e. n individuals) or dist object
cl vector with the clustering solution (its length should be n)

Details

Medoids are representative objects of a cluster whose average dissimilarity to all the objects in the
c cluster is minimal. Medoids are always members of the data set (contrary to means or centroids).

Value

Returns a numeric vector with the indexes of medoids.

Author(s)

Nicolas Robette

References


See Also
textindsup, speMCA, csMCA, varsup

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then computes statistics for supplementary individuals.
data(Music)
getindexcat(Music)
mca <- speMCA(Music[3:nrow(Music),1:5],excl=c(3,6,9,12,15))
indsup(mca,Music[1:2,1:5])
```

```r
table
```
modif.rate

Computes Benzecri's modified rates of variance of a MCA

Description

Computes the modified rates of variance of a multiple correspondence analysis.

Usage

modif.rate(resmca)

Arguments

resmca object of class MCA, speMCA, csMCA, stMCA or multiMCA

Details

As MCA clouds often have a high dimensionality, the variance rates of the first principle axes may be quite low, which makes them hard to interpret. Benzecri (1992, p.412) proposed to use modified rates to better appreciate the relative importance of the principal axes.

Value

Returns a data frame with 2 variables:

mrate Numeric vector of modified rates
cum.mrate Numeric vector of cumulative modified rates

References


See Also

dist, cluster, hclust, cutree, pam

Examples

## Performs of classification of the 'Music' example data set,## a clustering in 3 groups## and then computes the medoids.
data(Music)
temp <- dichotom(Music[,1:5])
d <- dist(temp)
clus <- cutree(hclust(d),3)
medoids(d,clus)
Movies

Author(s)
Nicolas Robette

References

See Also
MCA, speMCA, csMCA

Examples
```r
## Computes the modified rates of variance
## of the MCA of 'Music' example data set
data(Music)
mca <- speMCA(Music[,1:5])
modif.rate(mca)
```

Movies

Movies (data)

Description
The data concerns a sample of 1000 Movies which were on screens in France and some of their characteristics.

Usage
data(Movies)

Format
A data frame with 1000 observations and the following 7 variables:

- **Budget** numeric vector of movie budgets
- **Genre** is a factor with 9 levels
- **Country** is a factor with 4 levels. Country of origin of the movie.
- **ArtHouse** is a factor with levels No, Yes. Whether the movie had the "Art House" label.
- **Festival** is a factor with levels No, Yes. Whether the movie was selected in Cannes, Berlin or Venice film festivals.
- **Critics** numeric vector of average ratings from intellectual criticism.
- **BoxOffice** numeric vector of number of admissions.
multimCA

Examples

data(Movies)
str(Movies)

Performs Multiple Factor Analysis

Description
Performs Multiple Factor Analysis, drawing on the work of Escoffier and Pages (1994). It allows
the use of MCA variants (e.g. specific MCA or class specific MCA) as inputs.

Usage

multiMCA(l_mca, ncp = 5, compute.rv = FALSE)

Arguments

l_mca a list of objects of class MCA, speMCA or csMCA
ncp number of dimensions kept in the results (default is 5)
compute.rv whether RV coefficients should be computed or not (default is FALSE, which
makes the function execute faster)

Details
This function binds individual coordinates from every MCA in l_mca argument, weights them by
the first eigenvalue, and the resulting data frame is used as input for Principal Component Analysis
(PCA).

Value
Returns an object of class 'multimCA', i.e. a list:

eig a list of numeric vector for eigenvalues, percentage of variance and cumulative
percentage of variance

var a list of matrices with results for input MCAs components (coordinates, corre-
lation between variables and axes, square cosine, contributions)

ind a list of matrices with results for individuals (coordinates, square cosine, contrib-
utions)

call a list with informations about input data

VAR a list of matrices with results for categories and variables in the input MCAs
(coordinates, square cosine, test-values, variances)

my.mca lists the content of the objects in l_mca argument

RV a matrix of RV coefficients
Author(s)
Nicolas Robette

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

See Also
plot.multiMCA, varsup, speMCA, csMCA, MFA, PCA

Examples
## Performs a specific MCA on music variables of 'Taste' example data set,
## another one on movie variables of 'Taste' example data set,
## and then a Multiple Factor Analysis.
data(Taste)
getindexcat(Taste[,1:5])
mca1 <- speMCA(Taste[,1:5],excl=c(3,6,9,12,15))
getindexcat(Taste[,6:11])
mca2 <- speMCA(Taste[,6:11],excl=c(3,6,9,12,15,18))
mfa <- multiMCA(list(mca1,mca2))
plot.multiMCA(mfa)

---

Music (data)

Description
The data concerns tastes for music of a set of 500 individuals. It contains 5 variables of likes for music genres (french pop, rap, rock, jazz and classical), 2 about music listening and 2 additional variables (gender and age).

Usage
data(Music)

Format
A data frame with 500 observations and the following 7 variables:

FrenchPop  is a factor with levels No, Yes, NA
Rap  is a factor with levels No, Yes, NA
Rock  is a factor with levels No, Yes, NA
Jazz  is a factor with levels No, Yes, NA
Classical  is a factor with levels No, Yes, NA
Gender is a factor with levels Men, Women
Age is a factor with levels 15–24, 25–49, 50+
OnlyMus is a factor with levels Daily, Often, Rare, Never, indicating how often one only listens to music.
Daily is a factor with levels No, Yes indicating if one listens to music every day.

Details

'NA' stands for 'not available'

Examples

data(Music)
str(Music)

---

**pem**

*Computes the local and global Percentages of Maximum Deviation from Independence (PEM)*

---

**Description**

Computes the local and global Percentages of Maximum Deviation from Independence (PEM) of a contingency table.

**Usage**

```r
pem(x,y,weights=rep(1,length(x)),digits=1,sort=TRUE)
```

**Arguments**

- `x` : the first categorical variable
- `y` : the second categorical variable
- `weights` : an optional numeric vector of weights (by default, a vector of 1 for uniform weights)
- `digits` : integer. The number of digits (default is 3).
- `sort` : logical. Whether rows and columns are sorted according to a correspondence analysis or not (default is TRUE).

**Details**

The Percentage of Maximum Deviation from Independence (PEM) is an association measure for contingency tables and also provides attraction (resp. repulsion) measures in each cell of the crosstabulation (see Cibois, 1993). It is an alternative to khi2, Cramer’s V coefficient, etc.
Value
Returns a list:

- peml: Table with local percentages of maximum deviation from independence
- pemg: Numeric value, i.e. the global percentage of maximum deviation from independence

Author(s)
Nicolas Robette

References

See Also
table, chisq.test, phi.table, assocstats

Examples
```r
## Computes the PEM for the contingency table
## of jazz and age variables
## from the 'Music' example data set
data(Music)
pem(Music$Jazz,Music$Age)
```

phi.table

### Computes the phi coefficient for every cells of a contingency table

Description
Computes the phi coefficient for every cells of the cross-tabulation between two categorical variables

Usage
```r
phi.table(x,y,weights=rep(1,length(x)),digits=3)
```

Arguments
- x: the first categorical variable
- y: the second categorical variable
- weights: an optional numeric vector of weights (by default, a vector of 1 for uniform weights)
- digits: integer. The number of digits (default is 3).
Value

A table with the phi coefficients

Author(s)

Nicolas Robette

References


See Also

assoc.twocat, assoc.catcont, condesc, catdesc

Examples

data(Music)
phi.table(Music$Jazz, Music$Age)

plot.csMCA

Plots 'class specific' MCA results

Description

Plots a 'class specific' Multiple Correspondence Analysis (resulting from csMCA function), i.e. the clouds of individuals or categories.

Usage

## S3 method for class 'csMCA'
plot(x, type = "v", axes = 1:2, points = "all", col = "dodgerblue4", app = 0, ...)

Arguments

x object of class 'csMCA'
type character string: 'v' to plot the categories (default), 'i' to plot individuals' points, 'inames' to plot individuals' names
axes numeric vector of length 2, specifying the components (axes) to plot (c(1,2) is default)
points character string. If 'all' all points are plotted (default); if 'besth' only those who contribute most to horizontal axis are plotted; if 'bestv' only those who contribute most to vertical axis are plotted; if 'best' only those who contribute most to horizontal or vertical axis are plotted.
col color for the points of the individuals or for the labels of the categories (default is 'dodgerblue4')
app

numerical value. If 0 (default), only the labels of the categories are plotted and
their size is constant; if 1, only the labels are plotted and their size is proportional
to the weights of the categories; if 2, points (triangles) and labels are plotted, and
points size is proportional to the weight of the categories.

... further arguments passed to or from other methods, such as cex, cex.main, ...

Details

A category is considered to be one of the most contributing to a given axis if its contribution is
higher than the average contribution, i.e. 100 divided by the total number of categories.

Author(s)

Nicolas Robette

References

Le Roux B. and Rouanet H., *Multiple Correspondence Analysis*, SAGE, Series: Quantitative Ap-
Le Roux B. and Rouanet H., *Geometric Data Analysis: From Correspondence Analysis to Stuctured

See Also

csMCA, textvarsup, conc.ellipse

Examples

## Performs a class specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories
## and focusing on the subset of women,
## and then draws the cloud of categories.
data(Music)
female <- Music$Gender=='Women'
getindexcat(Music[,1:5])
mca <- csMCA(Music[,1:5],subcloud=female,excl=c(3,6,9,12,15))
plot(mca)
plot(mca,axes=c(2,3),points='best',col='darkred',app=1)

---

**plot.multiMCA**  
Plots Multiple Factor Analysis

Description

Plots Multiple Factor Analysis data, resulting from `multiMCA` function.
Usage

```r
## S3 method for class 'multiMCA'
plot(x, type = "v", axes = c(1, 2), points = "all", threshold = 2.58,
groups = 1:x$call$ngroups, col = rainbow(x$call$ngroups), app = 0, ...)
```

Arguments

- `x`  
  object of class `multiMCA`

- `type`  
  character string: 'v' to plot the categories (default), 'i' to plot individuals' points, 'inames' to plot individuals' names

- `axes`  
  numeric vector of length 2, specifying the components (axes) to plot (c(1,2) is default)

- `points`  
  character string. If 'all' all points are plotted (default); if 'besth' only those who are the most correlated to horizontal axis are plotted; if 'bestv' only those who are the most correlated to vertical axis are plotted; if 'best' only those who are the most coorelated to horizontal or vertical axis are plotted.

- `threshold`  
  numeric value. V-test minimal value for the selection of plotted categories.

- `groups`  
  numeric vector specifying the groups of categories to plot. By default, every groups of categories will be plotted

- `col`  
  a color for the points of the individuals or a vector of colors for the labels of the groups of categories (by default, rainbow palette is used)

- `app`  
  numerical value. If 0 (default), only the labels of the categories are plotted and their size is constant; if 1, only the labels are plotted and their size is proportional to the weights of the categories; if 2, points (triangles) and labels are plotted, and points size is proportional to the weight of the categories.

- `...`  
  further arguments passed to or from other methods, such as cex, cex.main, ...

Details

A category is considered to be one of the most correlated to a given axis if its test-value is higher then 2.58 (which corresponds to a 0.05 threshold).

Author(s)

Nicolas Robette

References

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

See Also

`multiMCA`, `textvarsup`, `speMCA`, `csMCA`, `MFA`
Examples

```r
## Performs a specific MCA on music variables of 'Taste' example data set,
## another one on movie variables of 'Taste' example data set,
## and then a Multiple Factor Analysis and plots the results.
data(Taste)
mca1 <- speMCA(Taste[,1:5],excl=c(3,6,9,12,15))
mca2 <- speMCA(Taste[,6:11],excl=c(3,6,9,12,15,18))
mfa <- multiMCA(list(mca1,mca2))
plot.multiMCA(mfa,col=c('darkred','darkblue'))
plot.multiMCA(mfa,groups=2,app=1)
```

---

**plot.speMCA**

Plots 'specific' MCA results

Description

Plots a 'specific' Multiple Correspondence Analysis (resulting from `speMCA` function), i.e. the clouds of individuals or categories.

Usage

```r
## S3 method for class 'speMCA'
plot(x, type = "v", axes = 1:2, points = "all", col = "dodgerblue4", app = 0, ...)
```

Arguments

- `x` object of class 'speMCA'
- `type` character string: 'v' to plot the categories (default), 'i' to plot individuals’ points, 'inames' to plot individuals’ names
- `axes` numeric vector of length 2, specifying the components (axes) to plot (c(1,2) is default)
- `points` character string. If 'all' all points are plotted (default); if 'besth' only those who contribute most to horizontal axis are plotted; if 'bestv' only those who contribute most to vertical axis are plotted; if 'best' only those who contribute most to horizontal or vertical axis are plotted.
- `col` color for the points of the individuals or for the labels of the categories (default is 'dodgerblue4')
- `app` numerical value. If 0 (default), only the labels of the categories are plotted and their size is constant; if 1, only the labels are plotted and their size is proportional to the weights of the categories; if 2, points (triangles) and labels are plotted, and points size is proportional to the weight of the categories.
- `...` further arguments passed to or from other methods, such as cex, cex.main, ...

Details

A category is considered to be one of the most contributing to a given axis if its contribution is higher than the average contribution, i.e. 100 divided by the total number of categories.
plot.stMCA

Author(s)
Nicolas Robette

References

See Also
speMCA, textvarsup, conc.ellipse

Examples
```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then draws the cloud of categories.
data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5], excl=c(3,6,9,12,15))
plot(mca)
plot(mca, axes=c(2,3), points="best", col="darkred", app=1)
```

plot.stMCA

Plots 'standardized' MCA results

Description
Plots a 'standardized' Multiple Correspondence Analysis (resulting from stMCA function), i.e. the clouds of individuals or categories.

Usage
```r
## S3 method for class 'stMCA'
plot(x, type = "v", axes = 1:2, points = "all", threshold = 2.58, groups=NULL, col = "dodgerblue4", app = 0, ...)
```

Arguments
- **x**: object of class 'stMCA'
- **type**: character string: 'v' to plot the categories (default), 'i' to plot individuals' points, 'inames' to plot individuals' names
- **axes**: numeric vector of length 2, specifying the components (axes) to plot (c(1,2) is default)
points character string. If 'all' all points are plotted (default); if 'besth' only those who are the most correlated to horizontal axis are plotted; if 'bestv' only those who are the most correlated to vertical axis are plotted; if 'best' only those who are the most coorelated to horizontal or vertical axis are plotted.

threshold numeric value. V-test minimal value for the selection of plotted categories.

groups only if x$call$input.mca = 'multiMCA', i.e. if the MCA standardized to x object was a multiMCA object. Numeric vector specifying the groups of categories to plot. By default, every groups of categories will be plotted.

col color for the points of the individuals or for the labels of the categories (default is 'dodgerblue4')

app numerical value. If 0 (default), only the labels of the categories are plotted and their size is constant; if 1, only the labels are plotted and their size is proportional to the weights of the categories; if 2, points (triangles) and labels are plotted, and points size is proportional to the weight of the categories.

... further arguments passed to or from other methods, such as cex, cex.main, ...

Details
A category is considered to be one of the most correlated to a given axis if its test-value is higher then 2.58 (which corresponds to a 0.05 threshold).

Author(s)
Nicolas Robette

References

See Also
stMCA, textvarsup, conc.ellipse

Examples
```r
## Performs a standardized MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories
## and controlling for age,
## and then draws the cloud of categories.
data(Music)
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
stmca <- stMCA(mca,control=list(Music$Age))
plot(stmca)
plot(stmca,axes=c(2,3),points='best',col='darkred',app=1)
```
speMCA

Performs a 'specific' MCA

Description

Performs a 'specific' Multiple Correspondence Analysis, i.e. a variant of MCA that allows to treat undesirable categories as passive categories.

Usage

speMCA(data, excl = NULL, ncp = 5, row.w = rep(1, times = nrow(data)))

Arguments

data : data frame with n rows (individuals) and p columns (categorical variables)
excl : numeric vector indicating the indexes of the "junk" categories (default is NULL). See "getindexcat" to identify these indexes.
cnp : number of dimensions kept in the results (default is 5)
row.w : an optional numeric vector of row weights (by default, a vector of 1 for uniform row weights)

Details

Undesirable categories may be of several kinds: infrequent categories (say, <5 percents), heterogeneous categories (e.g. 'others') or uninterpretable categories (e.g. 'not available'). In these cases, 'specific' MCA may be useful to ignore these categories for the determination of distances between individuals (see Le Roux and Rouanet, 2004 and 2010).

Value

Returns an object of class 'speMCA', i.e. a list including:

eig : a list of vectors containing all the eigenvalues, the percentage of variance, the cumulative percentage of variance, the modified rates and the cumulative modified rates
call : a list with informations about input data
ind : a list of matrices containing the results for the individuals (coordinates, contributions)
var : a list of matrices containing all the results for the categories and variables (weights, coordinates, square cosine, categories contributions to axes and cloud, test values (v.test), square correlation ratio (eta2), variable contributions to axes and cloud

Author(s)

Nicolas Robette
References


See Also

`getindexcat`, `plot.speMCA`, `varsup`, `contrib`, `modif.rate`, `dimdescr`, `MCA`, `csMCA`

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories.
data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
str(mca)
```

---

**stMCA**

*Performs a 'standardized' MCA*

### Description

Performs a 'standardized' Multiple Correspondence Analysis, i.e it takes MCA results and forces all the dimensions to be orthogonal to a supplementary 'control' variable.

### Usage

```r
stMCA(resmca, control)
```

### Arguments

- `resmca`: an object of class 'MCA', 'speMCA', 'csMCA' or 'multiMCA'
- `control`: a list of 'control' variables

### Details

Standardized MCA unfolds in several steps. First, for each dimension of an input MCA, individual coordinates are used as dependent variable in a linear regression model and the 'control' variable is included as covariate in the same model. The residuals from every models are retained and bound together. The resulting data frame is composed of continuous variables and its number of columns is equal to the number of dimensions in the input MCA. Lastly, this data frame is used as input in a Principal Component Analysis.
tabcontrib

Displays the categories contributing most to axes for a MCA

description

Identifies the categories that contribute the most to a given dimension of a Multiple Correspondence Analysis and organizes these informations into a fancy table. It allows to analyze variants of MCA, such as 'specific' MCA or 'class specific' MCA.

Usage

```
tabcontrib(resmca, dim = 1)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>resmca</td>
<td>object of class MCA, speMCA, or csMCA</td>
</tr>
<tr>
<td>dim</td>
<td>dimension to describe (default is 1st dimension)</td>
</tr>
</tbody>
</table>
Details

Best contributions - i.e. higher than average - are assigned a positive or negative sign according to the corresponding categories’ coordinates, so as to facilitate interpretation. Then they are sorted and organized according to the most contributing variables.

Value

Returns a data frame with the following columns:

- `var`: the names of the most contributing variables
- ` moda`: the names of the most contributing categories
- `ctr1`: ‘negative’ contributions, i.e. corresponding to categories with coordinates lower than zero
- `ctr2`: ‘positive’ contributions, i.e. corresponding to categories with coordinates higher than zero
- `weight`: weight of the categories
- `ctrtot`: sum of the best contributions for a given variable
- `cumctrtot`: cumulated contributions

Author(s)

Nicolas Robette

References


See Also

`dimcontrib`, `dimdesc`, `dimdescr`, `dimeta2`, `condes`, `speMCA`, `csMCA`

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then describes the contributions to axes.
data(Music)
getindexcat(Music[,1:5])
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
tabcontrib(mca,1)
tabcontrib(mca,2)
```
Description

The data concerns tastes for music and movies of a set of 2000 individuals. It contains 5 variables of likes for music genres (french pop, rap, rock, jazz and classical), 6 variables of likes for movie genres (comedy, crime, animation, science fiction, love, musical) and 2 additional variables (gender and age).

Usage

data(Taste)

Format

A data frame with 500 observations and the following 13 variables:

- FrenchPop is a factor with levels No, Yes, NA
- Rap is a factor with levels No, Yes, NA
- Rock is a factor with levels No, Yes, NA
- Jazz is a factor with levels No, Yes, NA
- Classical is a factor with levels No, Yes, NA
- Comedy is a factor with levels No, Yes, NA
- Crime is a factor with levels No, Yes, NA
- Animation is a factor with levels No, Yes, NA
- SciFi is a factor with levels No, Yes, NA
- Love is a factor with levels No, Yes, NA
- Musical is a factor with levels No, Yes, NA
- Gender is a factor with levels Men, Women
- Age is a factor with levels 15-24, 25-49, 50+
- Educ is a factor with levels none, low, medium, high

Details

'NA' stands for 'not available'

Examples

data(Taste)
str(Taste)
**Description**

Adds supplementary individuals to a MCA graph of the cloud of the individuals.

**Usage**

```r
textindsup(resmca, supdata, axes = c(1, 2), col = "darkred")
```

**Arguments**

- `resmca`: object of class 'MCA', 'speMCA', or 'csMCA'
- `supdata`: data frame with the supplementary individuals. It must have the same factors as the data frame used as input for the initial MCA.
- `axes`: numeric vector of length 2, specifying the dimensions (axes) to plot (default is c(1,2))
- `col`: color for the labels of the categories (default is 'darkred')

**Author(s)**

Nicolas Robette

**See Also**

`indsup`, `plot.speMCA`, `plot.csMCA`

**Examples**

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## plots the cloud of individuals,
## and then adds supplementary individuals.
data(Music)
getindexcat(Music)
mca <- speMCA(Music[3:nrow(Music),1:5], excl=c(3,6,9,12,15))
plot(mca,type='i')
textindsup(mca,Music[1:2,1:5])
```
Adds a categorical supplementary variable to a MCA graph

Description

Adds a categorical supplementary variable to a MCA graph of the cloud of categories.

Usage

textvarsup(resmca, var, sel = 1:nlevels(var), axes = c(1, 2),
       col = "black", app = 0, vname = NULL)

Arguments

resmca object of class 'MCA', 'speMCA', 'csMCA', 'stMCA' or 'multiMCA'
var the categorical supplementary variable. It does not need to have been used at the
MCA step.
sel numeric vector of indexes of the categories of the supplementary variable to be
added to the plot (by default, labels are plotted for every categories)
axes numeric vector of length 2, specifying the dimensions (axes) to plot (default is
c(1,2))
col color for the labels of the categories (default is black)
app numerical value. If 0 (default), only the labels are plotted and their size is con-
stant; if 1, only the labels are plotted and their size is proportional to the weights
of the categories; if 2, points (triangles) and labels are plotted, and points size is
proportional to the weight of the categories.
vname a character string to be used as a prefix for the labels of the categories (null by
default)

Author(s)

Nicolas Robette

See Also

plot.speMCA, plot.csMCA, plot.stMCA, plot.multiMCA, varsup

Examples

## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## plots the cloud of categories,
## and then adds gender and age supplementary categories.
data(Music)
getindexcat(Music)
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
translate.logit

Translates logit regression coefficients into percentages

Description

Performs a logit regression and then computes the effects of covariates expressed in percentages (through two methods: 'pure' effects and 'experimental' effects; see Deauvieau, 2010)

Usage

translate.logit(formula, data, nit=0)

Arguments

formula an object of class formula (or one that can be coerced to that class): a symbolic description of the model to be fitted. Every variables have to be factors.
data a data frame containing the variables in the model
nit number of bootstrap iterations for confidence interval computation. Default is 0, i.e. no confidence interval is computed.

Details

This function works with binomial as well as multinomial regression models. If the dependant variable has two factors, glm is used, if it has more than two factors multinom function (from nnet package) is used. The function expresses the regression coefficients as percentages through three distinct methods: raw percentages, 'pure effects' percentages and 'experimental effects' percentages (see Deauvieau, 2010). Bootstrap confidence interval are available only for binomial regressions.

Value

The function returns a list:

- glm An object of class glm or nnet (depending on the number of factors of the dependent variable)
- summary The results of summary function applied to reg element
- percents A matrix or a list of matrices (depending on the number of factors of the dependent variable) with regression coefficients expressed as percentages
- boot.ci A matrix or a list of matrices (depending on the number of factors of the dependent variable) with confidence intervals computed with bootstrap

Author(s)

Nicolas Robette
References


See Also

glm, multinom

Examples

```r
## An example for binomial logit regression
data(Music)
translate.logit(Daily ~ Gender + Age, Music)

## An example for multinomial logit regression
translate.logit(OnlyMus ~ Gender + Age, Music)
```

varsup

Computes statistics for a categorical supplementary variable

Description

From MCA results, computes statistics (weights, coordinates, contributions, test-values, variances) for a categorical supplementary variable.

Usage

```r
varsup(resmca, var)
```

Arguments

- `resmca`: object of class MCA, speMCA, csMCA, stMCA or multiMCA
- `var`: the categorical supplementary variable. It does not need to have been used at the MCA step.
Value

Returns a list:

- `weight` numeric vector of categories weights
- `coord` data frame of categories coordinates
- `cos2` data frame of categories square cosine
- `var` data frame of categories within variances, variance between and within categories and variable square correlation ratio (eta2)
- `typic` data frame of categories typicality test statistics
- `pval` data frame of categories p-values from typicality test statistics
- `cor` data frame of categories correlation coefficients

Author(s)

Nicolas Robette

References


See Also

`speMCA`, `csMCA`, `multiMCA`, `textvarsup`

Examples

```r
## Performs a specific MCA on 'Music' example data set
## ignoring every 'NA' (i.e. 'not available') categories,
## and then computes statistics for age supplementary variable.
data(Music)
getindexcat(Music)
mca <- speMCA(Music[,1:5],excl=c(3,6,9,12,15))
varsup(mca,Music$Age)
```

**wtable**

Computes a (possibly weighted) contingency table

Description

Computes a contingency table from one or two vectors, with the possibility of specifying weights.
Usage

```r
wtable(x, y=NULL, weights=rep.int(1,length(x)), stat="freq",
    digits=1, mar=TRUE, na_value=NULL)
```

Arguments

- `x`: an object which can be interpreted as factor
- `y`: an optional object which can be interpreted as factor
- `weights`: an optional numeric vector of weights (by default, a vector of 1 for uniform weights)
- `stat`: character. Whether to compute a contingency table ("freq", default, percentages ("prop"), row percentages ("rprop") or column percentages ("cprop").
- `digits`: integer indicating the number of decimal places (default is 1)
- `mar`: logical. If TRUE (default), margins are computed
- `na_value`: character. Name of the level for NA category. If NULL (default), NA values are ignored.

Value

Returns a contingency table.

Author(s)

Nicolas Robette

See Also

table, assoc.twocat

Examples

```r
## Computes a contingency table
## of jazz and age variables
## from the 'Music' example data set
## with or without weights
data(Music)
wtable(Music$Jazz)
wtable(Music$Jazz,Music$Age)

weight <- rep(c(0,0.5,1,1.5,2), length.out=nrow(Music))
wtable(Music$Jazz,w=weight,digits=1)
wtable(Music$Jazz,Music$Age,weight,1)
```
Index

* aplot
  conc.ellipse, 12
  ggadd_attractions, 26
  ggadd_interaction, 33
  ggadd_supind, 36
  ggadd_supvar, 37
  ggassoc_boxplot, 39
  ggassoc_crosstab, 40
  ggassoc_phiplot, 41
  ggassoc_scatter, 43
  ggcloud_indiv, 44
  ggcloud_variables, 46
  plot.csMCA, 57
  plot.multiMCA, 58
  plot.speMCA, 60
  plot.stMCA, 61
  textindsup, 68
  textvarsup, 69

* datasets
  Movies, 52
  Music, 54
  Taste, 67

* misc
  angles.csa, 3
  assoc.catcont, 4
  assoc.twocat, 5
  assoc.twocont, 7
  burt, 9
  catedesc, 10
  conc.ellipse, 12
  cndesc, 13
  contrib, 15
  dichotom, 19
  dimcontrib, 19
  dimdescr, 21
  dimmeta2, 22
  dimtypicality, 23
  flip.mca, 24
  getindexcat, 25
  homog.test, 48
  indsyp, 49
  medoids, 50
  modif.rate, 51
  pem, 55
  phi.table, 56
  tabcontrib, 65
  varsup, 71
  wtable, 72

* multivariate
  angles.csa, 3
  assoc.catcont, 4
  assoc.twocat, 5
  assoc.twocont, 7
  burt, 9
  catedesc, 10
  conc.ellipse, 12
  cndesc, 13
  contrib, 15
  csMCA, 16
  dichotom, 19
  dimcontrib, 19
  dimdescr, 21
  dimmeta2, 22
  dimtypicality, 23
  flip.mca, 24
  getindexcat, 25
  ggadd_attractions, 26
  ggadd_interaction, 33
  ggadd_supind, 36
  ggadd_supvar, 37
  ggassoc_boxplot, 39
  ggassoc_crosstab, 40
  ggassoc_phiplot, 41
  ggassoc_scatter, 43
  ggcloud_indiv, 44
  ggcloud_variables, 46
  homog.test, 48
  indsyp, 49
INDEX

medoids, 50
modif.rate, 51
multiMCA, 53
pem, 55
phi.table, 56
plot.csMCA, 57
plot.multiMCA, 58
plot.speMCA, 60
plot.stMCA, 61
speMCA, 63
stMCA, 64
tabcontrib, 65
textindsup, 68
textvarsup, 69
translate.logit, 70
varsup, 71
wtable, 72

angles.csa, 3
assoc.catcont, 4, 6, 7, 9, 18, 40, 57
assoc.twocat, 5, 5, 7, 9, 18, 41, 42, 57, 73
assoc.twocnt, 5, 6, 7, 9, 18, 43
assoc.xy, 5–7, 8, 11, 14, 18, 40–43
assocstats, 56

burt, 9
catdesc, 11
catdescr, 5–7, 9, 10, 14, 18, 41, 42, 57
chisq.test, 56
cluster, 51
conc.ellipse, 12, 58, 61, 62
condes, 14, 20, 23, 24, 66
condesc, 5–7, 9, 11, 13, 18, 21, 22, 40, 43, 57
contrib, 15, 17, 64
csMCA, 4, 15, 16, 20, 22–26, 49, 50, 52, 54,
75–59, 64–66, 72
cutree, 51
darma, 5–7, 9, 11, 14, 17, 40–43
dichotom, 10, 19
dimcontrib, 19, 23, 24, 66
dimdesc, 21–24, 66
dimdescr, 17, 20, 21, 23, 24, 64, 66
dimeta2, 20, 22, 24, 66
dimtypicality, 20, 23, 23
dist, 51

flip.mca, 24

getindexcat, 16, 17, 25, 64
ggadd_attractions, 26
ggadd_chulls, 27
ggadd_corr, 27, 28, 29, 31, 33, 34, 36, 39, 45,
48
ggadd_density, 27, 28, 30, 30, 33, 34, 36, 39,
45, 48
ggadd_ellipses, 27, 28, 30, 31, 32, 34, 36,
39, 45, 48
ggadd_interaction, 27, 28, 30, 31, 33, 33,
36, 39, 45, 48
ggadd_kellipses, 28, 33, 35
ggadd_supind, 36
ggadd_supvar, 28, 30, 31, 33, 34, 36, 37, 45,
48
ggassoc_boxplot, 5, 39
ggassoc_crosstab, 6, 40, 42
ggassoc_phiplot, 6, 41, 41
gassoc_scatter, 7, 43
ggcloud_indiv, 27, 28, 30, 31, 33, 34, 36, 37,
39, 44, 46
ggcloud_variables, 27, 28, 30, 31, 33, 34,
36, 39, 45, 46
ggpairs, 40–43
glm, 70, 71
hclust, 51
homog.test, 48
indsup, 37, 49, 68
MCA, 4, 15, 17, 25, 52, 64, 65
medoids, 50
MFA, 54, 59
modif.rate, 17, 51, 64
Movies, 52
multiMCA, 49, 53, 58, 59, 65, 72
multinom, 70, 71
Music, 54

pam, 51
PCA, 54, 65
pem, 55
phi.table, 41, 42, 56, 56
plot.csMCA, 13, 17, 57, 68, 69
plot.multiMCA, 13, 54, 58, 69
plot.speMCA, 13, 60, 64, 68, 69
plot.stMCA, 13, 61, 63, 69
speMCA, 4, 15, 17, 20, 22–26, 49, 50, 52, 54, 59–61, 63, 65, 66, 72
stMCA, 49, 61, 62, 64

tabcontrib, 65
table, 56, 73
Taste, 67
textindsup, 50, 68
textvarsup, 49, 58, 59, 61, 62, 69, 72
translate.logit, 70

varsup, 15, 17, 50, 54, 64, 69, 71

wtable, 72