

Package ‘MCAtools’

October 10, 2026

Type Package

Title Multiple Correspondence Analysis Toolkit (Phi-Divergence Framework)

Version 1.0.0

Description A toolkit for performing Multiple Correspondence Analysis (MCA) based on the phi-divergence framework of Cressie and Read (1984) <[doi:10.1111/j.2517-6161.1984.tb01318.x](https://doi.org/10.1111/j.2517-6161.1984.tb01318.x)>. Provides implementations for MCA computation `mca_analysis()`, visualization `plot_mca()`, and symbolic equation generation `generate_mca_equation()`, designed for interpretability, flexibility, and analytical consistency with generalized divergence measures.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 4.1.0)

Imports stats, graphics, utils

Suggests FactoMineR, ggplot2, dplyr

URL <https://github.com/nskamdem/MCAtools>

BugReports <https://github.com/nskamdem/MCAtools/issues>

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Awa Traoré [aut],
Assi Lazare N'GUESSAN [aut],
MOUSSA Kouamé Richard [aut],
Nona Servais KAMDEM SIMO [aut, cre],
Muhamad Setyawan Bahari [aut]

Maintainer Nona Servais KAMDEM SIMO <nona.kamdem@gmail.com>

Repository CRAN

Date/Publication 2026-10-10 11:40:02 UTC

Contents

electricity_africa	2
generate_mca_equation	7
mca_analysis	9
plot_mca	14
Index	17

electricity_africa	<i>Electricity market structures and electricity supply costs in Sub-Saharan Africa</i>
--------------------	---

Description

A cross-country dataset describing electricity sector organization, ownership, network configuration, geographic constraints, and average electricity supply costs in Sub-Saharan Africa.

The dataset combines observed country-level electricity system structures with counterfactual reform scenarios for Cameroon (Cameroon_2018, Cameroon_h1, Cameroon_h2), constructed to assess the potential impact of alternative market designs on electricity supply costs.

Usage

```
data(electricity_africa)
```

Format

A data frame with 42 country–system observations and the following variables:

country Country or system name. Includes observed countries and hypothetical reform scenarios for Cameroon.

iso_code ISO 3-letter country code or scenario identifier (e.g. CMR_2018, CMR_h1, CMR_h2).

total_cost Average electricity supply cost, expressed in USD per kWh. For hypothetical Cameroon scenarios, this variable may be missing and treated as an outcome of counterfactual analysis.

vertical_integration Binary indicator equal to 1 if generation, transmission and distribution are vertically integrated within a single utility, 0 otherwise.

private_in_generation Indicator for the presence of private sector participation in electricity generation (yes/no).

public_in_generation Indicator for public ownership or participation in electricity generation (yes/no).

stand_alone_generation Indicator equal to yes if generation assets operate independently from transmission and distribution.

transmission_bundled_distribution Indicator equal to 1 if transmission is institutionally bundled with distribution.

transmission_bundled_generation Indicator equal to 1 if transmission is institutionally bundled with generation.


```

        "multiple_transmission_system",
        "transmission_bundled_distribution",
        "multiple_distributor",
        "independent_distributor",
        "not_interconnected",
        "landlocked_country",
        "fragmented_island",
        "openness"
    ])

## Convert to factors
df[] <- lapply(df, as.factor)

## Cameroon reform scenarios treated as supplementary individuals
sup_rows <- which(is.na(electricity_africa$total_cost))

## -----
## 2. Ordinary MCA (lambda = 1)
## -----

res_ord <- mca_analysis(
  data = df,
  id_ind = "iso_code",
  sup_ind = sup_rows,
  lambda = 1
)

res_ord$eig$percentage[1]

## Extract first dimension (market structure)
dim1 <- res_ord$ind$coord$Dim1

## -----
## 3. Relationship between market structure and cost
## -----

analysis_df <- data.frame(
  country = electricity_africa$country[1:39],
  cost = log(electricity_africa$total_cost)[1:39],
  dim1 = scale(dim1)
)

analysis_df <- analysis_df[!is.na(analysis_df$cost), ]

## Quadratic relationship (U-shaped pattern)
model <- lm(cost ~ poly(dim1, 2), data = analysis_df)
summary(model)

## =====
## ROBUSTNESS ANALYSIS OVER THE STABILITY INTERVAL OF lambda
## =====

```

```

## -----
## 4. Selection of phi-divergence parameter lambda
## -----

lambda_grid <- seq(-2, 2, by = 0.05)

results <- data.frame(
  lambda = lambda_grid,
  inertia1 = NA_real_
)

for (k in seq_along(lambda_grid)) {
  res <- mca_analysis(
    data = df,
    id_ind = "iso_code",
    sup_ind = sup_rows,
    lambda = lambda_grid[k]
  )
  results$inertia1[k] <- res$eig$eigenvalue[1]
}

## Smooth inertia profile to avoid singularities at lambda = 0 and lambda = -1
singular <- abs(results$lambda) < 1e-6 |
  abs(results$lambda + 1) < 1e-6

results$inertia1_clean <- results$inertia1
results$inertia1_clean[singular] <- NA

for (k in seq_along(lambda_grid)) {
  ifelse(
    is.na(results$inertia1_clean[k]),
    results$inertia1_smooth[k] <-
      (results$inertia1_clean[k - 1] + results$inertia1_clean[k + 1]) / 2,
    results$inertia1_smooth[k] <- results$inertia1_clean[k]
  )
}

plot(
  results$lambda,
  results$inertia1_smooth,
  type = "l",
  lwd = 2,
  xlab = "lambda",
  ylab = "Inertie du premier axe",
  main = "Stabilite empirique de l'ACM phi-divergence"
)

abline(v = 1, col = "red", lty = 2)
text(1, max(results$inertia1), "lambda = 1 (chi-square)", pos = 4)

## Optimal lambda and 95% stability interval
lambda_opt <- results$lambda[which.max(results$inertia1_smooth)]

```

```

I_max <- max(results$inertial_smooth, na.rm = TRUE)
lambda_ci <- range(results$lambda[
  results$inertial_smooth >= 0.95 * I_max
])

lambda_opt
lambda_ci

lambda_min <- -0.335
lambda_max <- 0.095

lambda_grid <- seq(lambda_min, lambda_max, length.out = 25)

## Convert to factors
df[] <- lapply(df, as.factor)

## Container for pooled results
df_U_list <- vector("list", length(lambda_grid))

for (k in seq_along(lambda_grid)) {

  lambda_k <- lambda_grid[k]

  fit <- tryCatch(
    mca_analysis(
      data = df,
      id_ind = "iso_code",
      sup_ind = sup_rows,
      lambda = lambda_k
    ),
    error = function(e) NULL
  )

  if (is.null(fit)) next

  ## Active countries only (excluding Cameroon scenarios)
  n_active <- sum(!is.na(electricity_africa$total_cost))

  df_U_list[[k]] <- data.frame(
    country = rownames(fit$ind$coord)[1:n_active],
    dim1 = fit$ind$coord[1:n_active, 1],
    cost = log(electricity_africa$total_cost[1:n_active]),
    lambda = rep(lambda_k, n_active)
  )
}

df_U <- do.call(rbind, df_U_list)

## -----
## Normalization of Dim 1 within each lambda
## -----

```

```

df_U$dim1_z <- NA_real_

for (l in unique(df_U$lambda)) {
  idx <- df_U$lambda == l
  df_U$dim1_z[idx] <- as.numeric(scale(df_U$dim1[idx]))
}

## Central lambda (previously selected optimal value)
lambda_central <- lambda_opt

df_central <- df_U[
  abs(df_U$lambda - lambda_central) ==
  min(abs(df_U$lambda - lambda_central)),
]

df_central <- df_central[!duplicated(df_central$country), ]

## -----
## Pooled quadratic regression (robustness test)
## -----

curve_fit <- lm(
  cost ~ poly(dim1_z, 2),
  data = df_U
)

summary(curve_fit)

```

generate_mca_equation *Extract a Symbolic Equation for an MCA Dimension*

Description

Generates a symbolic representation of a selected dimension from a Multiple Correspondence Analysis (MCA) result. The equation highlights the modalities that contribute most to the chosen dimension, based on their coordinates and reconstructed contributions.

Usage

```

generate_mca_equation(
  mca_result,
  dimension = 1,
  top = 5,
  digits = 4
)

```

Arguments

<code>mca_result</code>	An object returned by <code>mca_analysis</code> , containing at least the component <code>\$var\$coord</code> . If available, <code>\$var\$contrib</code> is used; otherwise, contributions are reconstructed from squared coordinates.
<code>dimension</code>	Integer. Index of the MCA dimension for which the equation is extracted. Default is 1.
<code>top</code>	Integer. Number of most contributing modalities to include in the equation. Default is 5.
<code>digits</code>	Integer. Number of digits used to round the coefficients in the symbolic equation. Default is 4.

Details

For the selected dimension, the function extracts modality coordinates from the MCA result. If contribution values are not explicitly available, they are reconstructed as squared coordinates and normalized to sum to 100.

The symbolic equation is formed by retaining the top modalities with the highest absolute contributions. This provides an interpretable approximation of the latent axis in terms of the most influential modalities.

This approach is particularly useful in generalized or phi-divergence-based MCA settings, where classical contribution tables may not always be stored explicitly.

Value

A list with the following components:

<code>dimension</code>	The selected MCA dimension.
<code>equation</code>	A character string representing the symbolic equation of the dimension.
<code>table</code>	A data frame containing the selected modalities, their coordinates, and their percentage contributions.

Author(s)

Awa Traoré, Assi Lazare N'GUESSAN, MOUSSA Kouamé Richard, Nona Servais KAMDEM SIMO, Muhamad Setyawan Bahari

See Also

[mca_analysis](#) for computing MCA results.

Examples

```
# Example usage:
## Load package
library(MCAtools)

## Load data
data(electricity_africa)
```

```
## -----
## 1. Preparation of qualitative data for MCA
## -----

## Keep only qualitative variables (excluding cost)
df <- electricity_africa[, c("iso_code",
                             "vertical_integration",
                             "private_in_generation",
                             "public_in_generation",
                             "stand_alone_generation",
                             "transmission_bundled_generation",
                             "government_own_transmission",
                             "multiple_transmission_system",
                             "transmission_bundled_distribution",
                             "multiple_distributor",
                             "independent_distributor",
                             "not_interconnected",
                             "landlocked_country",
                             "fragmented_island",
                             "openness"
                           )]

## Convert to factors
df[] <- lapply(df, as.factor)

## Cameroon reform scenarios treated as supplementary individuals
sup_rows <- which(is.na(electricity_africa$total_cost))

res_ord <- mca_analysis(
  data = df,
  id_ind = "iso_code",
  sup_ind = sup_rows,
  lambda = 1
)

eq1 <- generate_mca_equation(
  mca_result = res_ord,
  dimension = 1,
  top = 6
)

eq1$equation
eq1$table
```

Description

Performs a Multiple Correspondence Analysis (MCA) on a `data.frame` of categorical variables using a general lambda-divergence residuals formulation. The function builds the indicator matrix, computes joint proportions, constructs residuals depending on the 'lambda' parameter, performs a singular value decomposition (SVD) and returns coordinates, contributions, $\cos^2$, η^2 , v-test for active variables and optionally handles supplementary qualitative and quantitative variables and supplementary individuals.

Usage

```
mca_analysis(data,
              id_ind = NULL,
              sup_ind = NULL,
              sup_qual = NULL,
              sup_quant = NULL,
              lambda = 1,
              nonzero = 0.05,
              dim1 = 1, dim2 = 2,
              plottype = "s",
              giveplot = TRUE,
              mag = 0.8,
              acc = 5,
              scaleplot = 1.2)
```

Arguments

<code>data</code>	A <code>data.frame</code> (or <code>tibble</code>) where each column is a factor (categorical variable). Each factor's levels are treated as the modalities used to build the indicator matrix.
<code>id_ind</code>	An optional character string giving the name of the column in <code>data</code> that contains individual identifiers. This column is not used to compute the analysis dimensions but is used to label individuals in the output. Required when <code>sup_ind</code> is supplied. Default is <code>NULL</code> .
<code>sup_ind</code>	An optional integer vector. If supplied, this vector is used to identify individuals (observations) that will be isolated as supplementary individuals. These individuals are not used to compute the analysis dimensions but their coordinates are computed a posteriori and returned in <code>ind_sup</code> . Default is <code>NULL</code> .
<code>sup_qual</code>	An optional <code>data.frame</code> with supplementary qualitative (categorical) variables, aligned row-for-row with <code>data</code> . If supplied, these variables are not used to compute the analysis dimensions but their group-mean coordinates and η^2 are computed a posteriori. Default is <code>NULL</code> .
<code>sup_quant</code>	An optional <code>data.frame</code> with supplementary quantitative (numeric) variables, aligned row-for-row with <code>data</code> . If supplied, these variables are not used to compute the analysis dimensions but their correlations with the individual coordinates are computed a posteriori. Default is <code>NULL</code> .
<code>lambda</code>	Numeric. Divergence parameter used to compute residuals. Common values: 1 (default, standard MCA / chi-square residuals formulation), 0 (log-based,

	Kullback-Leibler-type alternative), -1 (reverse Kullback-Leibler-type special case). The function contains safeguarded computations (small offsets) to avoid division by zero.
nonzero	Numeric scalar. Small positive value used in the <code>lambda == 0</code> branch to replace zero cell probabilities. Default 0.05.
dim1	Integer. First dimension index, kept for interface compatibility with <code>plot_mca</code> . Default is 1.
dim2	Integer. Second dimension index, kept for interface compatibility with <code>plot_mca</code> . Default is 2.
plottype	Character, kept for interface compatibility with <code>plot_mca</code> . Default "s".
giveplot	Logical, kept for interface compatibility with <code>plot_mca</code> . Default TRUE.
mag	Numeric, kept for interface compatibility with <code>plot_mca</code> . Default 0.8.
acc	Integer. Number of decimal digits used when rounding reported summary statistics such as Chi-squared, total inertia, and the p-value. Default 5.
scaleplot	Numeric, kept for interface compatibility with <code>plot_mca</code> . Default 1.2.

Details

This implementation follows these main steps:

1. Build the indicator (dummy) matrix Z from the factor columns of data. The number of columns of Z equals the total number of modalities across variables.
2. Compute the joint proportions matrix $P = Z / \text{sum}(Z)$, row and column margins ($p_{i.}$ and $p_{.j}$), and the independence matrix $E = p_{i.} p_{.j}^T$.
3. Compute residuals r_{ij} using the chosen lambda divergence. Three branches are implemented: general lambda (not 0 or -1), `lambda == 0` (log-based form with replacement of zeros by nonzero), and `lambda == -1` (special formulation). Tiny offsets (e.g. 1e-13) are added to avoid numerical instabilities.
4. Perform singular value decomposition (SVD) of the residual matrix r and compute singular values, principal coordinates for individuals and variables, contributions, cos2, v-test, and eta-squared statistics.
5. If `sup_ind` is provided, project the corresponding individuals onto the estimated factorial space using the same divergence branch as for active individuals (the branch is re-selected for the supplementary residuals, so the projection is consistent for `lambda = 0` and `lambda = -1` as well).
6. If `sup_qual` is provided, compute supplementary coordinates as group means of individual coordinates and compute η^2 for each supplementary variable and each dimension.
7. If `sup_quant` is provided, compute the correlation between each supplementary quantitative variable and the individual coordinates on each dimension.

Note: Input columns of data should be factors. If they are character vectors, convert using `data[] <- lapply(data, factor)` prior to calling the function.

The arguments `dim1`, `dim2`, `plottype`, `giveplot`, `mag`, and `scaleplot` are accepted for interface compatibility with `plot_mca` but do not affect the computations of `mca_analysis`; no plot is produced by this function.

Value

A list with components:

eig A data.frame of eigenvalues and percentage of explained inertia per dimension and cumulative percentage.

ind A list with individual results: `coord` (data.frame of individual coordinates), `contrib` (contribution percentages for individuals), and `cos2` (quality of representation).

var A list with active-modality results: `coord` (data.frame of modality coordinates), `contrib` (contribution percentages of modalities), `cos2` (modalities' cos2), `v.test` (v-test values), and `eta2` (eta-squared per active variable and dimension).

var_sup If `sup_qual` was provided, a list with `coord` (a list of data.frames, one per supplementary qualitative variable, giving the group-mean coordinates of its modalities) and `eta2` (data.frame of η^2 per supplementary variable and dimension). Otherwise NULL.

quanti_sup If `sup_quant` was provided, a data.frame of correlations between each supplementary quantitative variable and the individual coordinates, with dimensions as rows. Otherwise NULL.

ind_sup If `sup_ind` was provided, a data.frame of the projected coordinates of the supplementary individuals. Otherwise NULL.

Z The indicator matrix (as a data.frame) used in the analysis.

lambda The chosen lambda value (numeric), as supplied by the user.

Chi.Squared Numeric. Rounded Chi-squared statistic used for testing association.

Total.Inertia Numeric. Rounded total inertia (sum of principal inertias).

P.Value Numeric. Rounded p-value for the Chi-squared test.

All numeric matrices/data.frames indexed by dimension have column names `Dim1`, `Dim2`, ... corresponding to components extracted from the SVD.

Author(s)

Awa Traoré, Assi Lazare N'GUESSAN, MOUSSA Kouamé Richard, Nona Servais KAMDEM SIMO, Muhamad Setyawan Bahari

References

See standard texts on Multiple Correspondence Analysis and phi-divergence approaches for related formulations.

See Also

[plot_mca](#) for visualizing MCA results, and [generate_mca_equation](#) for extracting a symbolic interpretation of a dimension.

Examples

```

## Load data
data(electricity_africa)

## Load package
library(MCAtools)

## -----
## 1. Preparation of qualitative data for MCA
## -----

## Keep only qualitative variables (excluding cost)
df <- electricity_africa[, c("iso_code",
                           "vertical_integration",
                           "private_in_generation",
                           "public_in_generation",
                           "stand_alone_generation",
                           "transmission_bundled_generation",
                           "government_own_transmission",
                           "multiple_transmission_system",
                           "transmission_bundled_distribution",
                           "multiple_distributor",
                           "independent_distributor",
                           "not_interconnected",
                           "landlocked_country",
                           "fragmented_island",
                           "openness"
                           )]

## Convert to factors
df[] <- lapply(df, as.factor)

## Cameroon reform scenarios treated as supplementary individuals
sup_rows <- which(is.na(electricity_africa$total_cost))

## -----
## 2.Ordinary MCA (lambda = 1)
## -----

res_ord <- mca_analysis(
  data = df,
  id_ind = "iso_code",
  sup_ind = sup_rows,
  lambda = 1
)

res_ord$eig$percentage[1]

```

plot_mca

*Plot Multiple Correspondence Analysis Results***Description**

Produces 2D scatterplots from an MCA result object created by `mca_analysis`, displaying individuals and/or variable modalities on selected principal dimensions. The function provides several plot types (individuals, variables, or both) and allows customization of scaling and label size.

Usage

```
plot_mca(custom_result, plot_type = "both", scaleplot = 1.0,
        dims = c(1, 2), mag = 1.0, plottype = "s",
        label_ind = TRUE, label_var = TRUE, legend_pos = "topright")
```

Arguments

<code>custom_result</code>	An MCA result list obtained from <code>mca_analysis</code> . The object must contain the elements <code>ind\$coord</code> , <code>var\$coord</code> , and <code>eig</code> .
<code>plot_type</code>	Character string indicating what to display. One of "both" (individuals and variables, default), "ind" (individuals only), or "var" (variables only).
<code>scaleplot</code>	Numeric. Scaling factor applied to determine x/y axis limits. Default is 1.0.
<code>dims</code>	Integer vector of length two specifying which dimensions to plot (e.g. <code>c(1,2)</code>).
<code>mag</code>	Numeric. Magnification factor (similar to <code>cex</code>) for text labels. Default 1.0.
<code>plottype</code>	Character, passed to <code>par(pty=)</code> . Common options: "s" (square) or "m" (maximized). Default "s".
<code>label_ind</code>	Logical. If TRUE (the default), individual labels (row identifiers) are displayed on the plot. Set to FALSE to omit individual labels, which can improve readability when the number of observations is large.
<code>label_var</code>	Logical. If TRUE (the default), category (modality) labels are displayed on the plot. Set to FALSE to omit category labels, for example when combined with <code>mag</code> to emphasize the point cloud rather than the text.
<code>legend_pos</code>	Character string indicating the position of the legend, passed to legend . Accepted values include "topright" (the default), "topleft", "bottomright", "bottomleft", and other keywords accepted by legend . Ignored when <code>plot_type</code> does not require a legend.

Details

The function extracts the coordinates of individuals and variable modalities for the selected dimensions and generates a biplot-like figure:

- Individuals (rows) are represented by blue '+' symbols.
- Variable modalities (columns) are represented by red '*' symbols.

Labels are automatically annotated beside their respective points.

The axes labels indicate the dimension number and its percentage of explained inertia (from `custom_result$eig`).

A subtitle shows the cumulative percentage of total inertia explained by the displayed dimensions.

Horizontal and vertical reference lines are drawn at zero.

If `plot_type = "ind"`, only individuals are shown; if `plot_type = "var"`, only variables are shown.

Value

This function returns no value. It is used for its graphical output (a base R plot). The function produces a 2D plot of MCA results using the provided object.

Author(s)

Awa Traoré, Assi Lazare N'GUESSAN, MOUSSA Kouamé Richard, Nona Servais KAMDEM SIMO, Muhamad Setyawan Bahari

See Also

[mca_analysis](#) for generating the MCA result used as input.

Examples

```
library(MCAtools)

## -----
## 1. Preparation of qualitative data for MCA
## -----

## Keep only qualitative variables (excluding cost)
df <- electricity_africa[, c("iso_code",
                           "vertical_integration",
                           "private_in_generation",
                           "public_in_generation",
                           "stand_alone_generation",
                           "transmission_bundled_generation",
                           "government_own_transmission",
                           "multiple_transmission_system",
                           "transmission_bundled_distribution",
                           "multiple_distributor",
                           "independent_distributor",
                           "not_interconnected",
                           "landlocked_country",
                           "fragmented_island",
                           "openness"
                           )]

## Convert to factors
df[] <- lapply(df, as.factor)

## Cameroon reform scenarios treated as supplementary individuals
sup_rows <- which(is.na(electricity_africa$total_cost))
```

```
res_ord <- mca_analysis(  
  data = df,  
  id_ind = "iso_code",  
  sup_ind = sup_rows,  
  lambda = 1  
)  
  
# Plot individuals and variables together  
plot_mca(  
  res_ord,  
  plot_type = "both",  
  dims = c(1, 2),  
  mag = 0.8  
)  
  
# Plot individuals only  
plot_mca(  
  res_ord,  
  plot_type = "ind",  
  dims = c(1, 2),  
  mag = 0.8  
)  
  
# Plot variable modalities only  
plot_mca(  
  res_ord,  
  plot_type = "var",  
  dims = c(1, 2),  
  mag = 0.8  
)
```

Index

- * **hplot**

- plot_mca, [14](#)

- * **multivariate**

- generate_mca_equation, [7](#)

- mca_analysis, [9](#)

- plot_mca, [14](#)

electricity_africa, [2](#)

generate_mca_equation, [7](#), [12](#)

legend, [14](#)

mca_analysis, [8](#), [9](#), [15](#)

plot_mca, [11](#), [12](#), [14](#)