Package 'pams'

March 31, 2026

Type Package

Title Profile Analysis via Multidimensional Scaling

Version 0.1.0

Date 2026-03-22

Description Implements Profile Analysis via Multidimensional Scaling (PAMS)
for the identification of population-level core response profiles from
cross-sectional and longitudinal person-score data. Each person profile
is decomposed into a level component (the person mean) and a pattern
component (ipsatized subscores). PAMS uses nonmetric multidimensional
scaling via the SMACOF algorithm to identify a small number of core
profiles that represent the central response patterns in a sample of any
size. Bootstrap standard errors and bias-corrected and accelerated (BCa)
confidence intervals for individual core profile coordinates are
estimated, enabling significance testing of coordinates that is not
available in other profile analysis methods such as cluster profile
analysis or latent profile analysis. Person-level weights, R-squared
values, and correlations with core profiles are also estimated, allowing
individual profiles to be interpreted in terms of the core profile
structure. PAMS can be applied to both cross-sectional data and
longitudinal data, where core trajectory profiles describe how response
patterns change over time. Methods are described in
Kim and Kim (2024) <doi:10.20982/tqmp.20.3.p230>,
de Leeuw and Mair (2009) <doi:10.18637/jss.v031.i03>,
and Kruskal (1964) <doi:10.1007/BF02289565>.

License MIT + file LICENSE

URL https://github.com/sekangakim/pams

BugReports https://github.com/sekangakim/pams/issues

Depends R (>= 4.0.0)

Imports smacof (>= 2.1.0)

Suggests lmtest (>= 0.9.40), testthat (>= 3.0.0)

Encoding UTF-8

RoxygenNote 7.3.3

Contents

BootSmacof	*Profile Analysis via Multidimensional Scaling*

Description

Identifies population-level core response profiles from cross-sectional or longitudinal person-score data using nonmetric multidimensional scaling (SMACOF algorithm). Each person profile is decomposed into a level component (person mean) and a pattern component (ipsatized subscores). `BootSmacof` fits a nonmetric MDS solution to the $J \times J$ inter-variable distance matrix, bootstraps the solution to generate empirical sampling distributions of core profile coordinates, and computes bias-corrected and accelerated (BCa) confidence intervals for each coordinate. Person-level weights, R-squared values, and correlations with core profiles are estimated for all participants, with optional bootstrap confidence intervals for a selected subset.

Usage

```
BootSmacof(
  testdata,
  participant = NULL,
  mds = c("smacof", "classical"),
  type = c("ratio", "interval", "ordinal", "mspline"),
  distance = c("euclid", "sqeuclid"),
  scale = FALSE,
  nprofile = 3,
  direction = rep(1, nprofile),
  cl = 0.95,
  nBoot = 2000,
  testname = NULL,
  file = NULL
)
```

Arguments

testdata	A data frame or matrix of persons (rows) by subscales (columns). Subscores are assumed to be related and continuous. For longitudinal data, columns should be ordered as all subscales at Time 1 followed by all subscales at Time 2, and so on.
participant	An integer vector of row indices identifying persons for whom individual bootstrap confidence intervals on weights and partial correlations are computed. If NULL (the default), individual bootstrapping is skipped and only population-level results are returned.
mds	Character string specifying the MDS algorithm. Either "smacof" (default, recommended; uses the majorization algorithm of de Leeuw & Mair, 2009) or "classical" (Torgerson's classical metric MDS via [cmdscale](#)).
type	Character string specifying the optimal scaling transformation passed to [smacofSym](#). One of "ordinal" (default, nonmetric; recommended for most social-science data), "interval", "ratio", or "mspline". Ignored when mds = "classical".
distance	Character string specifying the distance measure used to compute the $J \times J$ inter-variable proximity matrix. Either "euclid" (default, Euclidean distance) or "sqeuclid" (squared Euclidean distance). Note that squaring amplifies large distances and compresses small ones; "euclid" is recommended unless faster convergence is specifically required.
scale	Logical. If TRUE, columns of testdata are standardised (zero mean, unit variance) before analysis. Set to TRUE when subscales have different measurement units. Default is FALSE.
nprofile	A positive integer specifying the number of core profiles (MDS dimensions) to extract. Choose by inspecting stress values across 2-, 3-, and 4-dimensional solutions; Kruskal's (1964) criterion of stress ≤ 0.05 is recommended. Must be less than the number of subscales (columns) in testdata.
direction	An integer vector of length nprofile, with each element either 1 or -1. Multiplying a dimension by -1 flips its sign to aid substantive interpretation (e.g., so that the first core profile aligns with the subscale mean profile). Default is rep(1, nprofile) (no flipping).
cl	Numeric confidence level for BCa intervals. Default is 0.95. Common alternatives are 0.99 and 0.90.
nBoot	A positive integer specifying the number of bootstrap samples. A minimum of 1000 is recommended for stable confidence interval estimation (Efron & Tibshirani, 1993); 2000 is the default.
testname	An optional character vector of length equal to the number of columns in testdata, giving subscale names used as row labels in summary output and plots. If NULL, labels "T1", "T2", ... are generated automatically.
file	An optional character string giving a file path stem. If supplied, three CSV files are written: <file>MDS.csv (stress summary and core profile coordinates with BCa CIs), <file>Weight.csv (person weights, levels, R-squared values, and core-profile correlations), and <file>WeightB.csv (bootstrap summaries for selected participants). If NULL (the default), no files are written.

Value

A named list with the following components:

MDS The MDS fit object for the original data. When mds = "smacof" this is the full object returned by smacofSym, including $conf (core profile coordinate matrix, $J \times K$) and $stress. When mds = "classical" this is a minimal list with $conf only.

MDSsummary A list of nprofile data frames, one per core profile. Each data frame has rows corresponding to subscales and columns: Ori (original coordinate), Mean (bootstrap mean), SE (bootstrap standard error), Lower and Upper (percentile CI bounds), BCaLower and BCaUpper (BCa CI bounds). Coordinates whose BCa CI does not include zero are statistically significant.

MDSprofile A list of nprofile matrices, each of dimension nBoot $\times J$, containing the full bootstrap distribution of core profile coordinates.

stresssummary A one-row data frame with bootstrap summary statistics for the smacof stress value: Ori, Mean, SE, Lower, Upper, BCaLower, BCaUpper. NULL when mds = "classical".

stressprofile A numeric vector of length nBoot containing bootstrap stress values. NULL when mds = "classical".

MDSR2 A numeric vector of length nprofile containing the R-squared values from regressing each core profile dimension on the remaining dimensions. Low values confirm that the core profiles are not collinear.

Weight A matrix of dimension $I \times (2K + 2)$ containing, for every person: raw weights (w1, ..., wK), level estimate, R-squared value, and correlations with each core profile (corDim1, ..., corDimK). Row names are "#1", "#2", ...

WeightmeanR2 The mean R-squared value across all I persons, summarising how well the nprofile core profiles account for pattern variance in the sample.

WeightB A matrix of bootstrap summary statistics (original estimate, mean, SE, lower and upper CI bounds) for the weights of each person in participant. NULL if participant is NULL.

PcorrB A matrix of bootstrap summary statistics for the partial correlations of each person in participant. NULL if participant is NULL.

nprofile The number of core profiles extracted.

nBoot The number of bootstrap samples used.

scale Logical; whether columns were standardised.

testname Character vector of subscale names used.

References

Davison, M. L. (1996). *Multidimensional scaling interest and aptitude profiles: Idiographic dimensions, nomothetic factors.* Presidential address to Division 5, American Psychological Association, Toronto.

de Leeuw, J., & Mair, P. (2009). Multidimensional scaling using majorization: SMACOF in R. *Journal of Statistical Software*, *31*(3), 1–30. doi:10.18637/jss.v031.i03

Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap.* Chapman & Hall.

Kim, S.-K., & Kim, D. (2024). Utility of profile analysis via multidimensional scaling in R for the study of person response profiles in cross-sectional and longitudinal data. *The Quantitative Methods for Psychology*, *20*(3), 230–247. doi:10.20982/tqmp.20.3.p230

Kruskal, J. B. (1964). Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, *29*, 1–27. doi:10.1007/BF02289565

See Also

smacofSym for the underlying MDS algorithm.

Examples

```
# Small toy example (runs automatically)
set.seed(42)
toy_data <- as.data.frame(matrix(rnorm(50 * 5, mean = 10, sd = 2),
                                  nrow = 50, ncol = 5))
colnames(toy_data) <- paste0("S", 1:5)

result <- BootSmacof(
  testdata    = toy_data,
  participant = 1:3,
  mds         = "smacof",
  type        = "ordinal",
  distance    = "euclid",
  nprofile    = 2,
  direction   = c(1, 1),
  cl          = 0.95,
  nBoot       = 20,
  testname    = colnames(toy_data)
)
result$MDS$stress
round(result$WeightmeanR2, 2)
round(result$MDSsummary[[1]], 3)


# Larger synthetic example demonstrating 3 core profiles (takes > 5 sec).
# In practice, replace syn_data with your own persons x subscales data frame.
set.seed(1)
syn_data <- as.data.frame(matrix(rnorm(200 * 8, mean = 50, sd = 10),
                                  nrow = 200, ncol = 8))
colnames(syn_data) <- paste0("Sub", 1:8)

result3 <- BootSmacof(
  testdata    = syn_data,
  participant = 1:5,
  mds         = "smacof",
  type        = "ordinal",
  distance    = "euclid",
  nprofile    = 3,
  direction   = c(1, 1, 1),
  cl          = 0.95,
  nBoot       = 200,
```

```
  testname    = colnames(syn_data),
  file        = file.path(tempdir(), "syn_profile")
)

result3$MDS$stress
round(result3$WeightmeanR2, 2)
round(result3$MDSsummary[[1]], 3)
round(result3$MDSsummary[[2]], 3)
round(result3$MDSsummary[[3]], 3)
round(result3$Weight[1:5, ], 2)
```

Index