Package 'mpm'

July 23, 2025

Version 1.0-23

Date 2022-03-31

Title Multivariate Projection Methods

Depends R (>= 2.10)

Imports MASS, KernSmooth

Description Exploratory graphical analysis of multivariate data, specifically gene expression data with different projection methods: principal component analysis, correspondence analysis, spectral map analysis.

License GPL (>= 2)

URL http://mpm.r-forge.r-project.org

Collate 'export.summary.mpm.R' 'Famin81A.R' 'Golub.R' 'mpm.R' 'plot.mpm.R' 'print.mpm.R' 'print.summary.mpm.R' 'summary.mpm.R' 'zzz.R'

Repository CRAN

Date/Publication 2022-04-01 12:40:09 UTC

RoxygenNote 7.1.2

NeedsCompilation no

Author Luc Wouters [aut], Tobias Verbeke [ctb], Laure Cougnaud [cre]

Maintainer Laure Cougnaud <laure.cougnaud@openanalytics.eu>

Contents

`export` *Generic Function to Export Output to Files*

Description

Generic Function to Export Output to Files

Usage

```
export(x, filename, ...)
```

Arguments

`x`	object to export to a file
`filename`	name of the file to which the output should be exported
`...`	further arguments for the method

Author(s)

Tobias Verbeke

`export.summary.mpm` *Export the summary output for an mpm object to a text file Output the mpm summary to a tab-demimited file for processing by other programs (Excel, Spotfire...) If the filename is empty, return the data instead of writing to file (useful for web services).*

Description

Polar (spherical) coordinates are added if the `summary.spm` object contains 2 (3) dimensions.

Usage

```
## S3 method for class 'summary.mpm'
export(x, filename = "", ...)
```

Arguments

`x`	object of class `summary.mpm` as produced by the function of the same name
`filename`	prefix used to name the output file following <filename>_xyz.txt
`...`	further arguments; currently none are used

Value

the output is returned invisibly

Author(s)

Rudi Verbeeck, Tobias Verbeke

See Also

summary.mpm

Famin81A *Famin81A Data Data with demographic indicators by region of the world*

Description

Famin81A Data Data with demographic indicators by region of the world

Format

A data frame with 18 observations on the following 5 variables.

Region a factor with 18 levels giving the region

Population.growth a numeric vector

Infant.Mortality a numeric vector; Infant mortality

Dietary.Energy a numeric vector; Dietary Energy

Productivity a numeric vector; Productivity

References

Friday, L. and Laskey, R. (1989). The Fragile Environment, The Darwin College Lecture. Cambridge University Press, UK.

Golub *Golub (1999) Data*

Description

Golub et al. (1999) data on gene expression profiles of 38 patients suffering from acute leukemia and a validation sample of 34 patients.

Format

The expression data are available in data frame `Golub` with 5327 observations on the following 73 variables.

list("Gene") a character vector with gene identifiers

list("1") gene expression data for sample 1

list("2") gene expression data for sample 2

list("3") gene expression data for sample 3

list("4") gene expression data for sample 4

list("5") gene expression data for sample 5

list("6") gene expression data for sample 6

list("7") gene expression data for sample 7

list("8") gene expression data for sample 8

list("9") gene expression data for sample 9

list("10") gene expression data for sample 10

list("11") gene expression data for sample 11

list("12") gene expression data for sample 12

list("13") gene expression data for sample 13

list("14") gene expression data for sample 14

list("15") gene expression data for sample 15

list("16") gene expression data for sample 16

list("17") gene expression data for sample 17

list("18") gene expression data for sample 18

list("19") gene expression data for sample 19

list("20") gene expression data for sample 20

list("21") gene expression data for sample 21

list("22") gene expression data for sample 22

list("23") gene expression data for sample 23

list("24") gene expression data for sample 24

list("25") gene expression data for sample 25

list("26") gene expression data for sample 26
list("27") gene expression data for sample 27
list("34") gene expression data for sample 34
list("35") gene expression data for sample 35
list("36") gene expression data for sample 36
list("37") gene expression data for sample 37
list("38") gene expression data for sample 38
list("28") gene expression data for sample 28
list("29") gene expression data for sample 29
list("30") gene expression data for sample 30
list("31") gene expression data for sample 31
list("32") gene expression data for sample 32
list("33") gene expression data for sample 33
list("39") gene expression data for sample 39
list("40") gene expression data for sample 40
list("42") gene expression data for sample 42
list("47") gene expression data for sample 47
list("48") gene expression data for sample 48
list("49") gene expression data for sample 49
list("41") gene expression data for sample 41
list("43") gene expression data for sample 43
list("44") gene expression data for sample 44
list("45") gene expression data for sample 45
list("46") gene expression data for sample 46
list("70") gene expression data for sample 70
list("71") gene expression data for sample 71
list("72") gene expression data for sample 72
list("68") gene expression data for sample 68
list("69") gene expression data for sample 69
list("67") gene expression data for sample 67
list("55") gene expression data for sample 55
list("56") gene expression data for sample 56
list("59") gene expression data for sample 59
list("52") gene expression data for sample 52
list("53") gene expression data for sample 53
list("51") gene expression data for sample 51
list("50") gene expression data for sample 50

list("54") gene expression data for sample 54

list("57") gene expression data for sample 57

list("58") gene expression data for sample 58

list("60") gene expression data for sample 60

list("61") gene expression data for sample 61

list("65") gene expression data for sample 65

list("66") gene expression data for sample 66

list("63") gene expression data for sample 63

list("64") gene expression data for sample 64

list("62") gene expression data for sample 62

The classes are in a separate numeric vector `Golub.grp` with values `1` for the 38 ALL B-Cell samples, `2` for the 9 ALL T-Cell samples and `3` for the 25 AML samples.

Details

The original data of Golub et al. (1999) were preprocessed as follows: genes that were called 'absent' in all samples were removed from the data sets, since these measurements are considered unreliable by the manufacturer of the technology. Negative measurements in the data were set to 1.

The resulting data frame contains 5327 genes of the 6817 originally reported by Golub et al. (1999).

Note

Luc Wouters et al. (2003), p. 1134 contains a typo concerning the sample sizes of AML- and ALL-type and erroneously reported

Source

Golub, T. R., Slonim, D. K., Tamayo, P., et al. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science 286, 531 – 537.

References

Luc Wouters et al. (2003). Graphical Exploration of Gene Expression Data: A Comparative Study of Three Multivariate Methods, Biometrics, 59, 1131-1139.

mpm *Spectral Map Analysis*

Description

Produces an object of class mpm that allows for exploratory multivariate analysis of large data matrices, such as gene expression data from microarray experiments.

Usage

```
mpm(
  data,
  logtrans = TRUE,
  logrepl = 1e-09,
  center = c("double", "row", "column", "global", "none"),
  normal = c("global", "row", "column", "none"),
  closure = c("none", "row", "column", "global", "double"),
  row.weight = c("constant", "mean", "median", "max", "logmean", "RW"),
  col.weight = c("constant", "mean", "median", "max", "logmean", "CW"),
  CW = rep(1, ncol(data) - 1),
  RW = rep(1, nrow(data)),
  pos.row = rep(FALSE, nrow(data)),
  pos.column = rep(FALSE, ncol(data) - 1)
)
```

Arguments

data	a data frame with the row descriptors in the first column. For microarray data rows indicate genes and columns biological samples.
logtrans	an optional logical value. If TRUE, data are first transformed to logarithms (base e) before the other operations. Non-positive numbers are replaced by logrepl. If FALSE, data are left unchanged. Defaults to TRUE.
logrepl	an optional numeric value that replaces non-positive numbers in log-transformations. Defaults to 1e-9.
center	optional character string specifying the centering operation that is carried out on the optionally log-transformed, closed data matrix. If "double" both row- and column-means are subtracted. If "row" row-means are subtracted. If "column" column-means are subtracted. If "none" the data are left uncentered. Defaults to "double".
normal	optional character string specifying the normalization operation that is carried out on the optionally log-transformed, closed, and centered data matrix. If "global" the data are normalized using the global standard deviation. If "row" data are divided by the standard deviations of the respective row. If "column" data are divided by their respective column standard deviation. If "none" no normalization is carried out. Defaults to "global".

closure	optional character string specifying the closure operation that is carried out on the optionally log-transformed data matrix. If "double", data are divided by row- and column-totals. If "row" data are divided by row-totals. If "column" data are divided by column-totals. If "none" no closure is carried out. Defaults to "none".
row.weight	optional character string specifying the weights of the different rows in the analysis. This can be "constant", "mean", "median", "max", "logmean", or "RW". If "RW" is specified, weights must be supplied in the vector RW. In other cases weights are computed from the data. Defaults to "constant", i.e. constant weighting.
col.weight	optional character string specifying the weights of the different columns in the analysis. This can be "constant", "mean", "median", "max", "logmean", or "CW". If "CW" is specified, weights must be supplied in the vector CW. In other cases weights are computed from the data. Defaults to "constant", i.e. constant weighting.
CW	optional numeric vector with external column weights. Defaults to 1 (constant weights).
RW	optional numeric vector with external row weights. Defaults to 1 (constant weights).
pos.row	logical vector indicating rows that are not to be included in the analysis but must be positioned on the projection obtained with the remaining rows. Defaults to FALSE.
pos.column	logical vector indicating columns that are not to be included in the analysis but must be positioned on the projection obtained with the remaining columns. Defaults to FALSE.

Details

The function mpm presents a unified approach to exploratory multivariate analysis encompassing principal component analysis, correspondence factor analysis, and spectral map analysis. The algorithm computes projections of high dimensional data in an orthogonal space. The resulting object can subsequently be used in the construction of biplots (i.e. plot.mpm).

The projection of the pre-processed data matrix in the orthogonal space is calculated using the La.svd function.

Value

An object of class mpm representing the projection of data after the different operations of transformation, closure, centering, and normalization in an orthogonal space. Generic functions plot and summary have methods to show the results of the analysis in more detail. The object consists of the following components:

TData	matrix with the data after optional log-transformation, closure, centering and normalization.
row.names	character vector with names of the row elements as supplied in the first column of the original data matrix

col.names	character vector with the names of columns obtained from the column names from the original data matrix
closure	closure operation as specified in the function call
center	centering operation as specified in the function call
normal	normalization operation as specified in the function call
row.weight	type of weighting used for rows as specified in the function call
col.weight	type of weighting used for columns as specified in the function call
Wn	vector with calculated weights for rows
Wp	vector with calculated weights for columns
RM	vector with row means of original data
CM	vector with column means of original data
pos.row	logical vector indicating positioned rows as specified in the function call
pos.column	logical vector indicating positioned columns as specified in the function call
SVD	list with components returned by La.svd
eigen	eigenvalues for each orthogonal factor from obtained from the weighted singular value decomposition
contrib	contributions of each factor to the total variance of the pre-processed data, i.e. the eigenvalues as a fraction of the total eigenvalue.
call	the matched call.

Note

Principal component analysis is defined as the projection onto an orthogonal space of the column-centered and column-normalized data. In correspondence factor analysis the data are pre-processed by double closure, double centering, and global normalization. Orthogonal projection is carried out using the weighted singular value decomposition. Spectral map analysis is in essence a principal component analysis on the log-transformed, double centered and global normalized data. Weighted spectral map analysis has been proven to be successful in the detection of patterns in gene expression data (Wouters et al., 2003).

Author(s)

Luc Wouters, Rudi Verbeeck, Tobias Verbeke

References

Wouters, L., Goehlmann, H., Bijnens, L., Kass, S.U., Molenberghs, G., Lewi, P.J. (2003). Graphical exploration of gene expression data: a comparative study of three multivariate methods. *Biometrics* **59**, 1131-1140.

See Also

plot.mpm, summary.mpm

Examples

```
data(Golub)
# Principal component analysis
r.pca <- mpm(Golub[,1:39], center = "column", normal = "column")
# Correspondence factor analysis
r.cfa <- mpm(Golub[,1:39],logtrans = FALSE, row.weight = "mean",
           col.weight = "mean", closure = "double")
# Weighted spectral map analysis
r.sma <- mpm(Golub[,1:39], row.weight = "mean", col.weight = "mean")
```

`plot.mpm` *Spectral Map Plot of Multivariate Data Produces a spectral map plot (biplot) of an object of class* `mpm`

Description

Spectral maps are special types of biplots with the area of the symbols proportional to some measure, usually the row or column mean value and an identification of row- and column-items. For large matrices, such as gene expression data, where there is an abundance of rows, this can obscure the plot. In this case, the argument `label.tol` can be used to select the most informative rows, i.e. rows that are most distant from the center of the plot. Only these row-items are then labeled and represented as circles with their areas proportional to the marginal mean value. For the column-items it can be useful to apply some grouping specified by `col.group`. Examples of groupings are different pathologies, such as specified in `Golub.grp`

Usage

```
## S3 method for class 'mpm'
plot(
  x,
  scale = c("singul", "eigen", "uvr", "uvc"),
  dim = c(1, 2),
  zoom = rep(1, 2),
  show.row = c("all", "position"),
  show.col = c("all", "position"),
  col.group = rep(1, length(x$col.names)),
 colors = c("orange1", "red", rainbow(length(unique(col.group)), start = 2/6, end =
   4/6)),
  col.areas = TRUE,
  col.symbols = c(1, rep(2, length(unique(col.group)))),
  sampleNames = TRUE,
  rot = rep(-1, length(dim)),
  labels = NULL,
  label.tol = 1,
  label.col.tol = 1,
  lab.size = 0.725,
  col.size = 10,
```

```
  row.size = 10,
  do.smoothScatter = FALSE,
  do.plot = TRUE,
  ...
)
```

Arguments

x	object of class mpm a result of a call to mpm.
scale	optional character string specifying the type of factor scaling of the biplot. This can be either "singul" (singular value scaling), "eigen" (eigenvalue scaling), "uvr" (unit row-variance scaling), "uvc" (unit column-variance scaling). The latter is of particular value when analyzing large matrices, such as gene expression data. Singular value scaling "singul" is customary in spectral map analysis. Defaults to "singul".
dim	optional principal factors that are plotted along the horizontal and vertical axis. Defaults to c(1,2).
zoom	optional zoom factor for row and column items. Defaults to c(1,1).
show.row	optional character string indicating whether all rows ("all") are to be plotted or just the positioned rows "position".
show.col	optional character string indicating whether all columns ("all") are to be plotted or just the positioned columns "position".
col.group	optional vector (character or numeric) indicating the different groupings of the columns, e.g. Golub.grp. Defaults to 1.
colors	vector specifying the colors for the annotation of the plot; the first two elements concern the rows; the third till the last element concern the columns; the first element will be used to color the unlabeled rows; the second element for the labeled rows and the remaining elements to give different colors to different groups of columns.
col.areas	logical value indicating whether columns should be plotted as squares with areas proportional to their marginal mean and colors representing the different groups (TRUE), or with symbols representing the groupings and identical size (FALSE). Defaults to TRUE.
col.symbols	vector of symbols when col.areas=FALSE corresponds to the pch argument of the function plot.
sampleNames	Either a logical vector of length one or a character vector of length equal to the number of samples in the dataset. If a logical is provided, sample names will be displayed on the plot (TRUE; default) or not (FALSE); if a character vector is provided, the names provided will be used to label the samples instead of the default column names.
rot	rotation of plot. Defaults to c(-1,-1).
labels	character vector to be used for labeling points on the graph; if NULL, the row names of x are used instead

label.tol	numerical value specifying either the percentile (label.tol<=1) of rows or the number of rows (label.tol>1) most distant from the plot-center (0,0) that are labeled and are plotted as circles with area proportional to the marginal means of the original data.
label.col.tol	numerical value specifying either the percentile (label.col.tol<=1) of columns or the number of columns (label.col.tol>1) most distant from the plot-center (0,0) that are labeled and are plotted as circles with area proportional to the marginal means of the original data.
lab.size	size of identifying labels for row- and column-items as cex parameter of the text function
col.size	size in mm of the column symbols
row.size	size in mm of the row symbols
do.smoothScatter	use smoothScatter or not instead of plotting individual points
do.plot	produce a plot or not
...	further arguments to eqscaleplot which draws the canvas for the plot; useful for adding a main or a custom sub

Value

An object of class plot.mpm that has the following components:

Rows	a data frame with the X and Y coordinates of the rows and an indication Select of whether the row was selected according to label.tol
Columns	a data frame with the X and Y coordinates of the columns

Note

value is returned invisibly, but is available for further use when an explicit assignment is made

Author(s)

Luc Wouters

References

Wouters, L., Goehlmann, H., Bijnens, L., Kass, S.U., Molenberghs, G., Lewi, P.J. (2003). Graphical exploration of gene expression data: a comparative study of three multivariate methods. *Biometrics* **59**, 1131-1140.

See Also

mpm, summary.mpm

Examples

```
# Weighted spectral map analysis
data(Golub) # Gene expression data of leukemia patients
data(Golub.grp) # Pathological classes coded as 1, 2, 3
r.sma <- mpm(Golub[,1:39], row.weight = "mean", col.weight = "mean")
# Spectral map biplot with result
r <- plot(r.sma, label.tol = 20, scale = "uvc",
          col.group = (Golub.grp)[1:38], zoom = c(1,1.2), col.size = 5)
Golub[r$Rows$Select, 1] # 20 most extreme genes
```

---

print.mpm *Print Method for mpm Objects*

Description

Print Method for mpm Objects

Usage

```
## S3 method for class 'mpm'
print(x, digits = 3, ...)
```

Arguments

x	object of class mpm
digits	minimum number of significant digits to be printed
...	further arguments for the print method (for printing the contributions)

Value

x is returned invisibly

See Also

print.default

`print.summary.mpm` *Print Method for summary.mpm Objects*

Description

Print Method for summary.mpm Objects

Usage

```
## S3 method for class 'summary.mpm'
print(x, digits = 2, what = c("columns", "rows", "all"), ...)
```

Arguments

`x`	object of class summary.mpm
`digits`	minimum number of significant digits to print, defaults to 2
`what`	one of `"columns"` (default), `"rows"` or `"all"`, specifying respectively whether columns, rows or both need to be printed
`...`	further arguments for the print method

Value

x is returned invisibly

See Also

`print.default`

`summary.mpm` *Summary Statistics for Spectral Map Analysis Summary method for object of class* `mpm`.

Description

The function `summary.mpm` computes and returns a list of summary statistics of the spectral map analysis given in `x`.

Usage

```
## S3 method for class 'mpm'
summary(object, maxdim = 4, ...)
```

Arguments

object	an object of class mpm resulting from a call to mpm
maxdim	maximum number of principal factors to be reported. Defaults to 4
...	further arguments; currently none are used

Value

An object of class summary.mpm with the following components:

call	the call to mpm
Vxy	sum of eigenvalues
VPF	a matrix with on the first line the eigenvalues and on the second line the cumulative eigenvalues of each of the principal factors (PRF1 to PRFmaxdim) followed by the residual eigenvalues and the total eigenvalue.
Rows	a data frame with summary statistics for the row-items, as described below.
Columns	a data frame with with summary statistics for the column-items, as described below.

The Rows and Columns data frames contain the following columns:

Posit	binary indication of whether the row or column was positioned (1) or not (0).
Weight	weight applied to the row or column in the function mpm.
PRF1-PRFmaxdim	factor scores or loadings for the first maxdim factors using eigenvalue scaling.
Resid	residual score or loading not accounted for by the first maxdim factors.
Norm	length of the vector representing the row or column in factor space.
Contrib	contribution of row or column to the sum of eigenvalues.
Accuracy	accuracy of the representation of the row or column by means of the first maxdim principal factors.

Author(s)

Luc Wouters

References

Wouters, L., Goehlmann, H., Bijnens, L., Kass, S.U., Molenberghs, G., Lewi, P.J. (2003). Graphical exploration of gene expression data: a comparative study of three multivariate methods. *Biometrics* **59**, 1131-1140.

See Also

mpm, plot.mpm

Examples

```
# Example 1 weighted spectral map analysis Golub data
data(Golub)
r.sma <- mpm(Golub[,1:39], row.weight = "mean", col.weight = "mean")
# summary report
summary(r.sma)
# Example 2 using print function
data(Famin81A)
r.fam <- mpm(Famin81A, row.weight = "mean", col.weight = "mean")
r.sum <- summary(r.fam)
print(r.sum, what = "all")
```

Index