Package ‘Correlplot’

April 2, 2026

Type Package

Title A Collection of Functions for Graphing Correlation Matrices

Version 1.1.3

Date 2026-03-15

Author Jan Graffelman [aut, cre],
Jan De Leeuw [aut]

Maintainer Jan Graffelman <jan.graffelman@upc.edu>

Depends R (>= 3.3.0), calibrate

Imports corrplot, xtable, MASS, lsei, ggplot2

Description Routines for the graphical representation of correlation matrices by means of correlograms, MDS maps and biplots obtained by PCA, PFA or WALS (weighted alternating least squares); See Graffelman & De Leeuw (2023) <doi:10.1080/00031305.2023.2186952>.

License GPL (>= 2)

URL https://www.r-project.org, https://www-eio.upc.edu/~jan/

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2026-04-02 05:10:07 UTC

Contents

`achievement` *Psychological Variables and Academic Performance*

Description

The data set contains psychological measures and academic achievements of 600 college freshmen. This is a classic example data set in multivariate analysis. The data consists of three psychological variables: locus of control, self concept and motivation; four acadamic variables: read, write, math, science and the demographic variables: female.

Usage

```
data("achievement")
```

Format

A data frame with 600 observations on the following 8 variables.

`locus` Locus of control

`self` Self concept

`motivation` Motivation

`read` Standarized test score

`write` Standarized test score

`math` Standarized test score

`science` Standarized test score

`female` Gender indicator (1=female,0=male)

Source

stats.oarc.ucla.edu

Examples

```
data(achievement)
```

`aircraft` *Characteristics of aircraft*

Description

Four variables registered for 21 types of aircraft.

Usage

```
data("aircraft")
```

Format

A data frame with 21 observations on the following 4 variables.

`SPR` specific power

`RGF` flight range factor

`PLF` payload

`SLF` sustained load factor

Source

Gower and Hand, Table 2.1

References

Gower, J.C. and Hand, D.J. (1996) *Biplots*, Chapman & Hall, London

Examples

```
data(aircraft)
str(aircraft)
```

`aircraftR` *Correlations between characteristics of aircraft*

Description

Correlations between SPR (specific power), RGF (flight range factor), PLF (payload) and SLF (sustained load factor) for 21 types of aircraft.

Usage

```
data(aircraftR)
```

Format

a matrix containing the correlations

Source

Gower and Hand, Table 2.1

References

Gower, J.C. and Hand, D.J. (1996) *Biplots*, Chapman & Hall, London

`angleToR` *Convert angles to correlations.*

Description

Function `angleToR` converts a vector of angles (in radians) to an estimate of the correlation matrix, given an interpretation function.

Usage

```
angleToR(x, ifun = "cos")
```

Arguments

`x`	a vector of angles (in radians)
`ifun`	the interpretation function ("cos" or "lincos")

Value

A correlation matrix

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. (2012) Linear-angle correlation plots: new graphs for revealing correlation structure. Journal of Computational and Graphical Statistics. 22(1): 92-106.

See Also

`cos`,`lincos`

Examples

```
angles <- c(0,pi/3)
R <- angleToR(angles)
print(R)
```

artificialR *Correlations for 10 generated variables*

Description

A 10 by 10 artificial correlation matrix

Usage

```
data(artificialR)
```

Format

A matrix of correlations

Source

Trosset (2005), Table 1.

References

Trosset, M.W. (2005) Visualizing correlation. *Journal of Computational and Graphical Statistics*, 14(1), pp. 1–19.

athletesR *Correlation matrix of characteristics of Australian athletes*

Description

Correlation matrix of 12 characteristics of Austration athletes (Sex, Height, Weight, Lean Body Mass, RCC, WCC, Hc, Hg, Ferr, BMI, SSF, Bfat)

Usage

```
data(athletesR)
```

Format

A matrix of correlations

Source

Weisberg (2005), file ais.txt

References

Weisberg, S. (2005) *Applied Linear Regression*. Third edition, John Wiley & Sons, New Jersey.

`banknotes` *Swiss banknote data*

Description

The Swiss banknote data consist of six measures taken on 200 banknotes, of which 100 are counterfeits, and 100 are normal.

Usage

```
data("banknotes")
```

Format

A data frame with 200 observations on the following 7 variables.

`Length` Banknote length

`Left` Left width

`Right` Right width

`Bottom` Bottom margin

`Top` Top margin

`Diagonal` Length of the diagonal of the image

`Counterfeit` 0 = normal, 1 = counterfeit

References

Weisberg, S. (2005) Applied Linear Regression. Third edition. John Wiley & Sons, New Jersey.

Examples

```
data(banknotes)
```

`berkeleyR` *Correlation matrix for boys of the Berkeley Guidance Study*

Description

Correlation matrix for sex, height and weight at age 2, 9 and 18 and somatotype

Usage

```
data(berkeleyR)
```

Format

A matrix of correlations

Source

Weisberg (2005), file BGSBoys.txt

References

Weisberg, S. (2005) *Applied Linear Regression*. Third edition, John Wiley & Sons, New Jersey.

`cathedralsR` *Correlation matrix for height and length*

Description

Correlation between nave height and total length

Usage

```
data(cathedralsR)
```

Format

A matrix of correlations

Source

Weisberg (2005), file cathedral.txt

References

Weisberg, S. (2005) *Applied Linear Regression*. Third edition, John Wiley & Sons, New Jersey.

correlogram — *Plot a correlogram*

Description

`correlogram` plots a correlogram for a correlation matrix.

Usage

```
correlogram(R,labs=colnames(R),ifun="cos",cex=1,main="",ntrials=50,
            xlim=c(-1.2,1.2),ylim=c(-1.2,1.2),pos=NULL,...)
```

Arguments

`R`	a correlation matrix.
`labs`	a vector of labels for the variables.
`ifun`	the interpretation function ("cos" or "lincos")
`cex`	character expansion factor for the variable labels
`main`	a title for the correlogram
`ntrials`	number of starting points for the optimization routine
`xlim`	limits for the x axis (e.g. c(-1.2,1.2))
`ylim`	limits for the y axis (e.g. c(-1.2,1.2))
`pos`	if specified, overrules the calculated label positions for the variables.
`...`	additional arguments for the `plot` function.

Details

`correlogram` makes a correlogram on the basis of a set of angles. All angles are given w.r.t the positive x-axis. Variables are represented by unit vectors emanating from the origin.

Value

A vector of angles

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Trosset, M.W. (2005) Visualizing correlation. Journal of Computational and Graphical Statistics 14(1), pp. 1–19

See Also

`fit_angles`,`nlminb`

Examples

```
X <- matrix(rnorm(90),ncol=3)
R <- cor(X)
angles <- correlogram(R)
```

---

countriesR — *Correlations between educational and demographic variables*

Description

Correlations between infant mortality, educational and demographic variables (infd, phys, dens, agds, lit, hied, gnp)

Usage

```
data(countriesR)
```

Format

A matrix of correlations

Source

Chatterjee and Hadi (1988)

References

Chatterjee, S. and Hadi, A.S. (1988), *Sensitivity Analysis in Regression*. Wiley, New York.

FitAllModelsRxy — *Fit all Models for the Between-Set Correlation Matrix*

Description

Function `FitAllModelsRxy` fits five models to approximate a between set correlation matrix. It calculates loss and RMSE for a canonical correlation analysis, and for four iterative alternating least squares algorithms that adjust the matrix for scalar, row and/or column effects.

Usage

```
FitAllModelsRxy(Rxy, Rxx, Ryy, eps = 1e-08, itmax = 1000, verbose = FALSE, digits = 12,
                ndim = 2)
```

Arguments

Rxy	The between set correlation matrix.
Rxx	The correlation matrix of the X variables.
Ryy	The correlation matrix of the Y variables.
eps	The numerical criterion for convergence (1e-08 by default).
itmax	The maximum number of iterations.
verbose	Print the iteration history (verbose=TRUE) of the iterative algorithms or not.
digits	Number of digits used for the final output.
ndim	Number of dimensions for the low-rank approximation.

Details

Function FitAllModelsRxy is useful for deciding if an adjustment is useful, and if so, which adjustment is most suitable.

Value

A dataframe with loss and RMSE statistics.

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman (2026) On the approximation of the between-set correlation matrix. Preprint.

See Also

FitRxy

Examples

```
data(achievement)
X <- achievement[,1:3]
Y <- achievement[,4:ncol(achievement)]
Rxy  <- cor(X,Y)
Rxx  <- cor(X)
Ryy  <- cor(Y)
Results <- FitAllModelsRxy(Rxy,Rxx,Ryy,verbose=FALSE,
                     eps=1e-08,ndim=2)
print(round(Results,6))
```

`FitRDeltaQSym` *Approximation of a correlation matrix with column adjustment and symmetric low rank factorization*

Description

Program `FitRDeltaQSym` calculates a low rank factorization for a correlation matrix. It adjusts for column effects, and the approximation is therefore asymmetric.

Usage

```
FitRDeltaQSym(R, W = NULL, nd = 2, eps = 1e-6, delta.init = 0,
              q.init = rep(0,ncol(R)),
              itmax = 1000, verbose = FALSE)
```

Arguments

`R`	A correlation matrix
`W`	A weight matrix (optional)
`nd`	The rank of the low rank approximation
`eps`	The convergence criterion
`delta.init`	Initial value for the scalar adjustment (zero by default)
`q.init`	Initial values for the column adjustments (a vector or zeroes by default)
`itmax`	Maximum number of iterations of the algorithm
`verbose`	Print information or not

Details

Program `FitRDeltaQSym` implements an iterative algorithm for the low rank factorization of the correlation matrix. It decomposes the correlation matrix as R = delta J + 1 q' + G G' + E. The approximation of R is ultimately asymmetric, but the low rank factorization used for biplotting (G G') is symmetric.

Value

A list object with fields:

`delta`	The final scalar adjustment
`q`	The final column adjustments
`G`	The matrix of biplot vectors
`fit.rmse`	The RSME of the approximation
`losshistory`	The value of the loss function for each iteration
`rmsehistory`	The RMSE of the approximation for each iteration
`Rhat`	The final approximation to the correlation matrix
`eps`	The threshold used for checking convergence
`nd`	The rank of the request approximation

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. (2025) Biplots for the correlation matrix. Journal of Computational and Graphical Statistics 34(4): 1591-1600. doi:10.1080/10618600.2025.2469757

See Also

wAddPCA,ipSymLS,Keller

Examples

```
data(HeartAttack)
X <- HeartAttack[,1:7]
X[,7] <- log(X[,7])
colnames(X)[7] <- "logPR"
R <- cor(X)
W <- matrix(1, 7, 7)
diag(W) <- 0
out.sym <- FitRDeltaQSym(R, W, eps=1e-6)
Rhat <- out.sym$Rhat
```

FitRwithPCAandWALS — *Calculate a low-rank approximation to the correlation matrix with four methods*

Description

Function `FitRwithPCAandWALS` uses principal component analysis (PCA) and weighted alternating least squares (WALS) to calculate different low-rank approximations to the correlation matrix.

Usage

```
FitRwithPCAandWALS(R, nd = 2, itmaxout = 10000, itmaxin = 10000, eps = 1e-08)
```

Arguments

`R`	The correlation matrix
`nd`	The dimensionality of the low-rank solution (2 by default)
`itmaxout`	Maximum number of iterations for the outer loop of the algorithm
`itmaxin`	Maximum number of iterations for the inner loop of the algorithm
`eps`	Numerical criterion for convergence of the outer loop

Details

Four methods are run succesively: standard PCA; PCA with an additive adjustment; WALS avoiding the fit of the diagonal; WALS avoiding the fit of the diagonal and with an additive adjustment.

Value

A list object with fields:

`Rhat.pca`	Low-rank approximation obtained by PCA
`Rhat.pca.adj`	Low-rank approximation obtained by PCA with adjustment
`Rhat.wals`	Low-rank approximation obtained by WALS without fitting the diagonal
`Rhat.wals.adj`	Low-rank approximation obtained by WALS without fitting the diagonal and with adjustment

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician 77(4): 432-442. doi:10.1080/00031305.2023.2186952

See Also

`wAddPCA`

Examples

```
data(HeartAttack)
X <- HeartAttack[,1:7]
X[,7] <- log(X[,7])
colnames(X)[7] <- "logPR"
R <- cor(X)
## Not run:
out <- FitRwithPCAandWALS(R)

## End(Not run)
```

`FitRxy` *Low rank approximation of the between-set correlation matrix*

Description

Function `FitRxy` fits a low-rank approximation to a between-set correlation, while allowing for adjustment by a scalar or column and/or row effects.

Usage

```
FitRxy(Rxy, R, C, ndim = 2, itmax = 1000, eps = 1e-08, verbose = TRUE,
adjust = "row", alpha = 1, lambda.eps = 1e-12)
```

Arguments

`Rxy`	The between-set correlation matrix.
`R`	The GLS weight matrix for the rows.
`C`	The GLS weight matrix for the columns.
`ndim`	The rank of the approximation (two by default).
`itmax`	The maximum number of iterations.
`eps`	The numerical criterion for convergence (1e-08 by default).
`verbose`	Print the iteration history (`verbose=TRUE`) or not.
`adjust`	The type of adjustment. Should be: "delta" (only a scalar adjustment), "col" (only adjument of the columns), "row" (only adjustment of the rows) or "both" (row and column adjustments).
`alpha`	Scaling factor for the biplot coordinates (1 = principal coordinates, 0 = standard coordinates, 0.5 = symmetric coordinates).
`lambda.eps`	The numerical criterion for considering small negative eigenvalues zero or not.

Details

Function `FitRxy` finds a low-rank approximation to the between-set correlation matrix while allowing for scalar, row and/or column adjustments. It implements an alternating least squares algorithm.

Value

`y`	The low-rank approximation to the correlation matrix.
`Fc`	Biplot coordinates for the rows of Rxy.
`Gc`	Biplot coordinates for the columns of Rxy.
`itel`	Number of iterations until convergence.
`re`	Estimated row adjustments.
`ce`	Estimated column adjustments.
`delta`	Estimated scalar adjustment.
`loss`	Value of the loss function upon convergence.
`rmse.approximation`	The root-mean-squared-error of the low-rank approximation to Rxy.
`convergence`	`TRUE` if the decrease in the loss function drops below `eps`.

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman (2026) On the approximation of the between-set correlation matrix. Preprint.

See Also

wAddPCA

Examples

```
data(achievement)
X <- achievement[,1:3]
Y <- achievement[,4:ncol(achievement)]
Rxy  <- cor(X,Y)
Rxx  <- cor(X)
Ryy  <- cor(Y)
out.delta <- FitRxy(Rxy,solve(Rxx),solve(Ryy),
                adjust="delta",eps=1e-08,
                verbose=FALSE)
```

---

fit_angles *Fit angles to a correlation matrix*

Description

`fit_angles` finds a set of optimal angles for representing a particular correlation matrix by angles between vectors

Usage

```
fit_angles(R, ifun = "cos", ntrials = 10, verbose = FALSE)
```

Arguments

`R`	a correlation matrix.
`ifun`	an angle interpretation function (cosine, by default).
`ntrials`	number of trials for optimization routine `nlminb`
`verbose`	be silent (FALSE), or produce more output (TRUE)

Value

a vector of angles (in radians)

Author(s)

anonymous

References

Trosset, M.W. (2005) Visualizing correlation. Journal of Computational and Graphical Statistics 14(1), pp. 1–19

See Also

nlminb

Examples

```
X <- matrix(rnorm(90),ncol=3)
R <- cor(X)
angles <- fit_angles(R)
print(angles)
```

---

fysiologyR — *Correlations between thirtheen fysiological variables*

Description

Correlations of 13 fysiological variables (sys, dia, p.p., pul, cort, u.v., tot/100, adr/100, nor/100, adr/tot, tot/hr, adr/hr, nor/hr) obtained from 48 medical students

Usage

```
data(fysiologyR)
```

Format

A matrix of correlations

Source

Hills (1969), Table 1.

References

Hills, M (1969) On looking at large correlation matices *Biometrika* 56(2): pp. 249.

ggbplot *Create a biplot with ggplot2*

Description

Function ggbiplot creates a biplot of a matrix with ggplot2 graphics.

Usage

```
ggbplot(A, B, main = "", circle = TRUE, xlab = "", ylab = "", main.size = 8,
xlim = c(-1, 1), ylim = c(-1, 1), rowcolor = "red", rowch = 1, colcolor = "blue",
colch = 1, rowarrow = FALSE, colarrow = TRUE, linewidth = 0.25, size = 1.5,
onedimensional = FALSE)
```

Arguments

A	A dataframe with coordinates and names for the biplot row markers.
B	A dataframe with coordinates and names for the biplot column markers.
main	A title for the biplot.
circle	Draw a unit circle (circle=TRUE) or not (circle=FALSE).
xlab	The label for the x axis.
ylab	The label for the y axis.
main.size	Size of the main title.
xlim	Limits for the horizontal axis.
ylim	Limits for the vertical axis.
rowcolor	Color used for the row markers.
rowch	Symbol used for the row markers.
colcolor	Color used for the column markers.
colch	Symbol used for the column markers.
rowarrow	Draw arrows from the origin to the row markers (rowarrow=TRUE) or not.
colarrow	Draw arrows from the origin to the column markers (colarrow=TRUE) or not.
linewidth	Width of the vectors in the biplot.
size	Size of the labels in the plot.
onedimensional	With onedimensional=TRUE a one dimensional biplot will be created that separates the biplot vectors in the second dimension.

Details

Dataframes A and B must consists of dataframes with three columns labeled "PA1", "PA2" (coordinates of the first and second principal axis) and a column "strings" with the labels for the coordinates. Optionally, these dataframes can contain two columns with labels "ve" and "ho" containing the vertical and horizontal adjustments for the label positions of the variables in the biplot.

Dataframe B is optional. If it is not specified, a biplot with a single set of markers is constructed, for which the row settings must be specified.

Value

A ggplot2 object

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician 77(4): 432-442. doi:10.1080/00031305.2023.2186952

See Also

bplot,ggtally,biplot

Examples

```
data("HeartAttack")
X <- as.matrix(HeartAttack[,1:7])
n <- nrow(X)
Xt <- scale(X)/sqrt(n-1)
res.svd <- svd(Xt)
Fs <- sqrt(n)*res.svd$u # standardized principal components
Gp <- crossprod(t(res.svd$v),diag(res.svd$d)) # biplot coordinates for variables
rows.df <- data.frame(Fs[,1:2],as.character(1:n))
colnames(rows.df) <- c("PA1","PA2","strings")
cols.df <- data.frame(Gp[,1:2],colnames(X))
colnames(cols.df) <- c("PA1","PA2","strings")
ggbplot(rows.df,cols.df,xlab="PA1",ylab="PA2",main="PCA")
```

---

ggcorrelogram *Create a correlogram as a ggplot object.*

Description

Function `ggcorrelogram` creates a correlogram of a correlation matrix using ggplot graphics.

Usage

```
ggcorrelogram(R, labs = colnames(R), ifun = "cos", cex = 1, main = "", ntrials = 50,
           xlim = c(-1.2, 1.2), ylim = c(-1.2, 1.2), hjust = 1, vjust = 2, size = 2,
      main.size = 8)
```

Arguments

R	a correlation matrix
labs	a vector of labels for the variables
ifun	the interpretation function ("cos" or "lincos")
cex	character expansion factor for the variable labels
main	a title for the correlogram
ntrials	number of starting points for the optimization routine
xlim	limits for the x axis (e.g. c(-1.2,1.2))
ylim	limits for the y axis (e.g. c(-1.2,1.2))
hjust	horizontal adjustment of variable labels (by default 1 for all variables)
vjust	vertical adjustment of variable labels (by default 2 for all variables)
size	font size for the labels of the variables
main.size	font size of the main title of the correlogram

Details

ggcorrelogram makes a correlogram on the basis of a set of angles. All angles are given w.r.t the positive x-axis. Variables are represented by unit vectors emanating from the origin.

Value

A ggplot object. Field theta of the output contains the angles for the variables.

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Trosset, M.W. (2005) Visualizing correlation. Journal of Computational and Graphical Statistics 14(1), pp. 1–19

See Also

correlogram,fit_angles,nlminb

Examples

```
set.seed(123)
X <- matrix(rnorm(90),ncol=3)
R <- cor(X)
angles <- ggcorrelogram(R)
```

ggtally *Create a correlation tally stick on a biplot vector*

Description

Function ggtally puts a series of dots along a biplot vector of a correlation matrix, so marking the change in correlation along the vector with specified values.

Usage

```
ggtally(p1, A, B, R, ind = 1:nrow(B), adj = 0, values = seq(-1,1,by=0.2), dotsize = 0.10,
        dotcolour = "black", dp = FALSE, linewidth = 0.1,
W = diag(nrow(A)), xlim = c(-1, 1), ylim = c(-1, 1), verbose = FALSE,
onedimensional = FALSE)
```

Arguments

p1	A ggplot2 object with an existing biplot.
A	Biplot markers of the rows.
B	Biplot markers of the columns (typically the biplot vector to be calibrated).
R	The original matrix (e.g., the correlatin matrix) to be represented.
ind	The indices (row numbers in matrix B) of the biplot vector(s) to be calibrated. If not specified, all row vectors of B will be calibrated.
adj	A scalar adjustment for the correlations.
values	Values of the correlations to be marked off by dots.
dotsize	Size of the dot.
dotcolour	Colour of the dot.
dp	Drops perpendiculars (dp=TRUE) onto the biplot vector that is calibrated. Only applies to two-dimensional biplots (onedimensional=FALSE.
linewidth	The width of the biplot vector(s)
W	Weight matrix used in the calibration.
xlim	Limits for the horizontal axis. These should coincide with those used in ggbplot.
ylim	Limits for the vertical axis. These should coincide with those used in ggbplot.
verbose	Prints coordinates of tick marks if verbose=TRUE.
onedimensional	For one-dimensional biplots. This should coincide with onedimensional used in ggbplot.

Details

Any set of values for the correlation to be marked off can be used, though a standard scale with 0.2 increments is recommmended.

Value

A ggplot2 object with the updated biplot

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician, 77(4), 432-442. doi:10.1080/00031305.2023.2186952

See Also

ggbplot

Examples

```
library(calibrate)
data(goblets)
R <- cor(goblets)
out.sd <- eigen(R)
V  <- out.sd$vectors[,1:2]
Dl <- diag(out.sd$values[1:2])
Gp <- crossprod(t(V),sqrt(Dl))
pca.df <- data.frame(Gp)
pca.df$strings <- colnames(R)
colnames(pca.df) <- c("PA1","PA2","strings")
p1 <- ggbplot(pca.df,pca.df,main="PCA correlation biplot",xlab="",ylab="",rowarrow=TRUE,
              rowcolor="blue",rowch="",colch="")
p1 <- ggtally(p1,Gp,Gp,R,values=seq(-0.2,0.6,by=0.2),dotsize=0.1)
```

---

gobletsR — *Correlations between size measurements of archeological goblets*

Description

Correlations between 6 size measurements of archeological goblets

Usage

```
data(gobletsR)
```

Format

A matrix of correlations

Source

Manly (1989)

References

Manly, B.F.J. (1989) *Multivariate statistical methods: a primer*. Chapman and Hall, London.

`HeartAttack` *Myocardial infarction or Heart attack data*

Description

Data set consisting of 101 observations of patients who suffered a heart attack.

Usage

```
data("HeartAttack")
```

Format

A data frame with 101 observations on the following 8 variables.

- `Pulse` Pulse
- `CI` Cardiac index
- `SI` Systolic index
- `DBP` Diastolic blood pressure
- `PA` Pulmonary artery pressure
- `VP` Ventricular pressure
- `PR` Pulmonary resistance
- `Status` Deceased or survived

Source

Table 18.1, (Saporta 1990, pp. 452–454)

References

Saporta, G. (1990) Probabilites analyse des donnees et statistique. Paris, Editions technip

Examples

```
data(HeartAttack)
str(HeartAttack)
```

`ipSymLS` *Function for obtaining a weighted least squares low-rank approximation of a symmetric matrix*

Description

Function `ipSymLS` implements an alternating least squares algorithm that uses both decomposition and block relaxation to find the optimal positive semidefinite approxation of given rank p to a known symmetric matrix of order n.

Usage

```
ipSymLS(target, w = matrix(1, dim(target)[1], dim(target)[2]), ndim = 2,
        init = FALSE, itmax = 100, eps = 1e-06, verbose = FALSE)
```

Arguments

`target`	Symmetric matrix to be approximated
`w`	Matrix of weights
`ndim`	Number of dimensions extracted (2 by default)
`init`	Initial value for the solution (optional; if supplied should be a matrix of dimensions `nrow(target)` by `ndim`)
`itmax`	Maximum number of iterations
`eps`	Tolerance criterion for convergence
`verbose`	Show the iteration history (`verbose=TRUE`) or not (`verbose=FALSE`)

Value

A matrix with the coordinates for the variables

Author(s)

deleeuw@stat.ucla.edu

References

De Leeuw, J. (2006) A decomposition method for weighted least squares low-rank approximation of symmetric matrices. Department of Statistics, UCLA. Retrieved from https://escholarship.org/uc/item/1wh197mh

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician 77(4): 432-442. doi:10.1080/00031305.2023.2186952

Examples

```
data(banknotes)
R <- cor(banknotes)
W <- matrix(1,nrow(R),nrow(R))
diag(W) <- 0
Fp.als <- ipSymLS(R,w=W,verbose=TRUE,eps=1e-15)
Rhat.als <- Fp.als%*%t(Fp.als)
```

---

jointlim *Establish limits for x and y axis*

Description

jointlim computes a sensible range for x and y axis if two sets of points are to be plotted simultaneously

Usage

```
jointlim(X, Y)
```

Arguments

X	Matrix of coordinates
Y	Matrix of coordinates

Value

xlim	minimum and maximum for x-range
ylim	minimum and maximum for y-range

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

Examples

```
X <- matrix(runif(20),ncol=2)
Y <- matrix(runif(20),ncol=2)
print(jointlim(X,Y)$xlim)
```

Keller — *Program* `Keller` *calculates a rank p approximation to a correlation matrix according to Keller's method.*

Description

Keller's method is based on iterated eigenvalue decompositions that are used to adjust the diagonal of the correlation matrix.

Usage

```
Keller(R, eps = 1e-06, nd = 2, itmax = 10)
```

Arguments

`R`	A correlation matrix
`eps`	Numerical criterion for convergence (default `eps=1e-06`)
`nd`	Number of dimensions used in the spectral decomposition (default `nd=2`)
`itmax`	The maximum number of iterations

Value

A matrix containing the approximation to the correlation matrix-

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Keller, J.B. (1962) Factorization of Matrices by Least-Squares. Biometrika, 49(1 and 2) pp. 239–242.

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician, 77(4), 432-442. doi:10.1080/00031305.2023.2186952

See Also

`ipSymLS`

Examples

```
data(Kernels)
R <- cor(Kernels)
Rhat <- Keller(R)
```

`Kernels` *Wheat kernel data*

Description

Wheat kernel data set taken from the UCI Machine Learning Repository

Usage

```
data("Kernels")
```

Format

A data frame with 210 observations on the following 8 variables.

`area` Area of the kernel

`perimeter` Perimeter of the kernel

`compactness` Compactness (C = 4*pi*A/P^2)

`length` Length of the kernel

`width` Width of the kernel

`asymmetry` Asymmetry coefficient

`groove` Length of the groove of the kernel

`variety` Variety (1=Kama, 2=Rosa, 3=Canadian)

Source

https://archive.ics.uci.edu/ml/datasets/seeds

References

M. Charytanowicz, J. Niewczas, P. Kulczycki, P.A. Kowalski, S. Lukasik, S. Zak, A Complete Gradient Clustering Algorithm for Features Analysis of X-ray Images. in: Information Technologies in Biomedicine, Ewa Pietka, Jacek Kawa (eds.), Springer-Verlag, Berlin-Heidelberg, 2010, pp. 15-24.

Examples

```
data(Kernels)
```

linangplot — *Linang plot*

Description

`linangplot` produces a plot of two variables, such that the correlation between the two variables is linear in the angle.

Usage

```
linangplot(x, y, tmx = NULL, tmy = NULL, ...)
```

Arguments

`x`	x variable
`y`	y variable
`tmx`	vector of tickmarks for the x variable
`tmy`	vector of tickmarks for the y variable
`...`	additional arguments for the plot routine

Value

`Xt`	coordinates of the points
`B`	axes for the plot
`r`	correlation coefficient
`angledegrees`	angle between axes in degrees
`angleradians`	angle between axes in radians
`r`	correlation coefficient

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

See Also

`plotcorrelogram`

Examples

```
x <- runif(10)
y <- rnorm(10)
linangplot(x,y)
```

`lincos` *Linearized cosine function*

Description

Function `lincos` linearizes the cosine function over the interval [0,2pi]. The function returns -2/pi*x + 1 over [0,pi] and 2/pi*x - 3 over [pi,2pi]

Usage

```
lincos(x)
```

Arguments

`x`	angle in radians

Value

a real number in [-1,1].

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. (2012) Linear-angle correlation plots: new graphs for revealing correlation structure. Journal of Computational and Graphical Statistics. 22(1): 92-106.

See Also

`cos`

Examples

```
angle <- pi
y <- lincos(angle)
print(y)
```

`pco` *Principal Coordinate Analysis*

Description

`pco` is a program for Principal Coordinate Analysis.

Usage

```
pco(Dis)
```

Arguments

`Dis`	A distance or dissimilarity matrix

Details

The program `pco` does a principal coordinates analysis of a dissimilarity (or distance) matrix (Dij) where the diagonal elements, Dii, are zero.

Note that when we dispose of a similarity matrix rather that a distance matrix, a transformation is needed before calling coorprincipal. For instance, if Sij is a similarity matrix, Dij might be obtained as Dij = 1 - Sij/diag(Sij)

Goodness of fit calculations need to be revised such as to deal (in different ways) with negative eigenvalues.

Value

`PC`	the principal coordinates
`Dl`	all eigenvalues of the solution
`Dk`	the positive eigenvalues of the solution
`B`	double centred matrix for the eigenvalue decomposition
`decom`	the goodness of fit table

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

See Also

`cmdscale`

Examples

```
citynames <- c("Aberystwyth","Brighton","Carlisle","Dover","Exeter","Glasgow","Hull",
"Inverness","Leeds","London","Newcastle", "Norwich")
A <-matrix(c(
0,244,218,284,197,312,215,469,166,212,253,270,
244,0,350,77,167,444,221,583,242,53,325,168,
218,350,0,369,347,94,150,251,116,298,57,284,
284,77,369,0,242,463,236,598,257,72,340,164,
197,167,347,242,0,441,279,598,269,170,359,277,
312,444,94,463,441,0,245,169,210,392,143,378,
215,221,150,236,279,245,0,380,55,168,117,143,
469,583,251,598,598,169,380,0,349,531,264,514,
166,242,116,257,269,210,55,349,0,190,91,173,
212,53,298,72,170,392,168,531,190,0,273,111,
253,325,57,340,359,143,117,264,91,273,0,256,
270,168,284,164,277,378,143,514,173,111,256,0),ncol=12)
rownames(A) <- citynames
colnames(A) <- citynames
out <- pco(A)
plot(out$PC[,2],-out$PC[,1],pch=19,asp=1)
textxy(out$PC[,2],-out$PC[,1],rownames(A))
```

`PearsonLee` *Heights of mothers and daughters*

Description

Heights of 1375 mothers and daughters (in cm) in the UK in 1893-1898.

Usage

```
data(PearsonLee)
```

Format

dataframe with Mheight and Dheight

Source

Weisberg, Chapter 1

References

Weisberg, S. (2005) *Applied Linear Regression*, John Wiley & Sons, New Jersey

pfa *Principal factor analysis*

Description

Program pfa performs (iterative) principal factor analysis, which is based on the computation of eigenvalues of the reduced correlation matrix.

Usage

```
pfa(X, option = "data", m = 2, initial.communality = "R2", crit = 0.001, verbose = FALSE)
```

Arguments

X	A data matrix or correlation matrix
option	Specifies the type of matrix supplied by argument X. Values for option are data, cor or cov. data is the default.
m	The number of factors to extract (2 by default)
initial.communality	Method for computing initial communalites. Possibilities are R2 or maxcor.
crit	The criterion for convergence. The default is 0.001. A smaller value will require more iterations before convergence is reached.
verbose	When set to TRUE, additional numerical output is shown.

Value

Res	Matrix of residuals
Psi	Diagonal matrix with specific variances
La	Matrix of loadings
Shat	Estimated correlation matrix
Fs	Factor scores

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Mardia, K.V., Kent, J.T. and Bibby, J.M. (1979) Multivariate analysis.

Rencher, A.C. (1995) Methods of multivariate analysis.

Satorra, A. and Neudecker, H. (1998) Least-Squares Approximation of off-Diagonal Elements of a Variance Matrix in the Context of Factor Analysis. Econometric Theory 14(1) pp. 156–157.

See Also

`princomp`

Examples

```
   X <- matrix(rnorm(100),ncol=2)
   out.pfa <- pfa(X)
#  based on a correlation matrix
   R <- cor(X)
   out.pfa <- pfa(R,option="cor")
```

`proteinR` *Correlations between sources of protein*

Description

Correlations between sources of protein for a number of countries (Red meat, White meat, Eggs, Milk, Fish, Cereals, Starchy food, Nuts, Fruits and vegetables.

Usage

`data(proteinR)`

Format

A matrix of correlations

Source

Manly (1989)

References

Manly, B.F.J. (1989) *Multivariate statistical methods: a primer*. Chapman and Hall, London.

proteinsR *Correlations between sources of protein*

Description

Correlations between sources of protein for a number of countries (Red meat, White meat, Eggs, Milk, Fish, Cereals, Starchy food, Nuts, Fruits and vegetables.

Usage

```
data(proteinR)
```

Format

A matrix of correlations

Source

Manly (1989)

References

Manly, B.F.J. (1989) *Multivariate statistical methods: a primer*. Chapman and Hall, London.

recordsR *Correlations between national track records for men*

Description

Correlations between national track records for men (100m,200m,400m,800m,1500m,5000m,10.000m and Marathon

Usage

```
data(recordsR)
```

Format

A matrix of correlations

Source

Johnson and Wichern, Table 8.6

References

Johnson, R.A. and Wichern, D.W. (2002) *Applied Multivariate Statistical Analysis.* Fifth edition. New Jersey: Prentice Hall.

`rmse` *Calculate the root mean squared error*

Description

Program `rmse` calculates the RMSE for a matrix approximation.

Usage

```
rmse(R, Rhat, W = matrix(1, nrow(R), ncol(R)) - diag(nrow(R)),
     verbose = FALSE, per.variable = FALSE)
```

Arguments

`R`	The original matrix
`Rhat`	The approximating matrix
`W`	A symmetric matrix of weights
`verbose`	Print output (`verbose=TRUE`) or not (`verbose=FALSE`)
`per.variable`	Calculate the RMSE for the whole matrix (`per.variable=FALSE`) or for each variable seperately (`per.variable=TRUE`)

Details

By default, function `rmse` assumes a symmetric correlation matrix as input, together with its approximation. The approximation does not need to be symmetric. Weight matrix W has to be symmetric. By default, the diagonal is excluded from RMSE calcuations (W = J - I). To include it, specify W = J, that is set W = matrix(1, nrow(R), ncol(R))

Value

the calculated rmse

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician 77(4): 432-442. doi:10.1080/00031305.2023.2186952

Examples

```
data(banknotes)
X <- as.matrix(banknotes[,1:6])
p <- ncol(X)
J <- matrix(1,p,p)
R <- cor(X)
out.sd <- eigen(R)
V <- out.sd$vectors
Dl <- diag(out.sd$values)
V2 <- V[,1:2]
D2 <- Dl[1:2,1:2]
Rhat <- V2%*%D2%*%t(V2)
rmse(R,Rhat,W=J)
```

---

rmse.rxy *Calculate RMSE of a Low-rank Approximation to the Between-set Correlation Matrix*

Description

Function `rmse.rxy` calculates the root-mean-squared error (RMSE) of a low-rank approximation to the between-set correlation matrix.

Usage

```
rmse.rxy(Rxy, Rhat, R, C)
```

Arguments

Rxy	The between-set correlation matrix.
Rhat	The low-rank approximation to the between-set correlation matrix.
R	The weight matrix for the rows.
C	The weight matrix for the columns.

Details

By default, weighting by generalised least squares is assumed, and weight matrices R and C must be supplied. The RMSE according to an ordinary least squares criterion can be obtained by setting `R = diag(nrow(Rxy))` and `C = diag(ncol(Rxy))`.

Value

The RMSE

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician 77(4): 432-442. doi:10.1080/00031305.2023.2186952

See Also

rmse

Examples

```
data(achievement)
X <- achievement[,1:3]
Y <- achievement[,4:ncol(achievement)]
Rxy  <- cor(X,Y)
Rxx  <- cor(X)
Ryy  <- cor(Y)
out.delta <- FitRxy(Rxy,solve(Rxx),solve(Ryy),
                adjust="delta",eps=1e-08,
                verbose=FALSE)
Rxy.hat <- out.delta$delta
rmse.rxy(Rxy,Rxy.hat,R=solve(Rxx),C=solve(Ryy))
```

---

rmsePCAandWALS — *Generate a table of root mean square error (RMSE) statistics for principal component analysis (PCA) and weighted alternating least squares (WALS).*

Description

Function `rmsePCAandWALS` creates table with the RMSE for each variable, for a low-rank approximation to the correlation matrix obtained by PCA or WALS.

Usage

```
rmsePCAandWALS(R, output, digits = 4, omit.diagonals = c(FALSE,FALSE,TRUE,TRUE))
```

Arguments

`R`	The correlation matrix
`output`	A list object with four approximationst to the correlation matrix
`digits`	The number of digits used in the output
`omit.diagonals`	Vector of four logicals for omitting the diagonal of the correlation matrix for RMSE calculations. Defaults to c(FALSE,FALSE,TRUE,TRUE), to include the diagonal for PCA and exclude it for WALS

Value

A matrix with one row per variable and four columns for RMSE statistics.

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician 77(4): 432-442. doi:10.1080/00031305.2023.2186952

See Also

FitRwithPCAandWALS

Examples

```
data(HeartAttack)
X <- HeartAttack[,1:7]
X[,7] <- log(X[,7])
colnames(X)[7] <- "logPR"
R <- cor(X)
## Not run:
out <- FitRwithPCAandWALS(R)
Results <- rmsePCAandWALS(R,out)

## End(Not run)
```

---

storksR *Correlations between three variables*

Description

Danish data from 1953-1977 giving the correlations between nesting storks, human birth rate and per capita electricity consumption.

Usage

```
data(storksR)
```

Format

A matrix of correlations

Source

Gabriel and Odoroff, Table 1.

References

Gabriel, K. R. and Odoroff, C. L. (1990) Biplots in biomedical research. *Statistics in Medicine* 9(5): pp. 469-485.

`students` *Marks for 5 student exams*

Description

Matrix of marks for five exams, two with closed books and three with open books (Mechanics (C), Vectors (C), Algebra (O), Analysis (O) and Statistics (O)).

Usage

```
data(students)
```

Format

A data matrix

Source

Mardia et al., Table 1.2.1

References

Mardia, K.V., Kent, J.T. and Bibby, J.M. (1979) *Multivariate Analysis*, Academic Press London.

`studentsR` *Correlations between marks for 5 exams*

Description

Correlation matrix of marks for five exams, two with closed books and three with open books (Mechanics (C), Vectors (C), Algebra (O), Analysis (O) and Statistics (O)).

Usage

```
data(studentsR)
```

Format

A matrix of correlations

Source

Mardia et al., Table 1.2.1

References

Mardia, K.V., Kent, J.T. and Bibby, J.M. (1979) *Multivariate Analysis*, Academic Press London.

tally *Create a tally on a biplot vector*

Description

Function `tally` marks of a set of dots on a biplot vector. It is thought for biplot vectors representing correlations, such that their correlation scale becomes visible, without doing a full calibration with tick marks and tick mark labels.

Usage

```
tally(G, adj = 0, values = seq(-1, 1, by = 0.2), pch = 19, dotcolor = "black", cex = 0.5,
      color.negative = "red", color.positive = "blue")
```

Arguments

`G`	Matrix with biplot coordinates of the variables
`adj`	A scalar adjustment for the correlations
`values`	The values of the correlations to be marked off by dots
`pch`	The character code used for marking off correlations
`dotcolor`	The colour of the dots that are marked off
`cex`	The character expansion factor for a dot.
`color.negative`	The colour of the segments of the negative part of the correlation scale
`color.positive`	The colour of the segments of the positive part of the correlation scale

Value

NULL

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician, 77(4), 432-442. doi:10.1080/00031305.2023.2186952

See Also

`bplot`, `calibrate`

Examples

```
data(goblets)
R <- cor(goblets)
results <- eigen(R)
V  <- results$vectors
Dl <- diag(results$values)
#
# Calculate correlation biplot coordinates
#
G  <- crossprod(t(V[,1:2]),sqrt(Dl[1:2,1:2]))
#
# Make the biplot
#
bplot(G,G,rowch=NA,colch=NA,collab=colnames(R),
      xl=c(-1.1,1.1),yl=c(-1.1,1.1))
#
# Create a correlation tally stick for variable X1
#
tally(G[1,])
```

---

tr — *Compute the trace of a matrix*

Description

`tr` computes the trace of a matrix.

Usage

```
tr(X)
```

Arguments

X	a (square) matrix

Value

the trace (a scalar)

Author(s)

Jan Graffelman (jan.graffelman@upc.edu)

Examples

```
X <- matrix(runif(25),ncol=5)
print(X)
print(tr(X))
```

wAddPCA *Low-rank matrix approximation by weighted alternating least squares*

Description

Function wAddPCA calculates a weighted least squares approximation of low rank to a given matrix.

Usage

```
wAddPCA(x, w = matrix(1, nrow(x), ncol(x)), p = 2, add = "all", bnd = "opt",
        itmaxout = 1000, itmaxin = 1000, epsout = 1e-06, epsin = 1e-06,
verboseout = TRUE, verbosein = FALSE)
```

Arguments

x	The data matrix to be approximated
w	The weight matrix
p	The dimensionality of the low-rank solution (2 by default)
add	The additive adjustment to be employed. Can be "all" (default), "nul" (no adjustment), "one" (adjustment by a single scalar), "row" (adjustment by a row) or "col" (adjustment by a column).
bnd	Can be "opt" (default), "all", "row" or "col".
itmaxout	Maximum number of iterations for the outer loop of the algorithm
itmaxin	Maximum number of iterations for the inner loop of the algorithm
epsout	Numerical criterion for convergence of the outer loop
epsin	Numerical criterion for convergence of the inner loop
verboseout	Be verbose on the outer loop iterations
verbosein	Be verbose on the inner loop iterations

Value

A list object with fields:

a	The left matrix (A) of the factorization X = AB’
b	The right matrix (B) of the factorization X = AB’
z	The product AB’
f	The final value of the loss function
u	Vector for rows used to construct rank 1 weights
v	Vector for columns used to construct rank 1 weights
p	The vector with row adjustments
q	The vector with column adjustments
itel	Iterations needed for convergence
delta	The additive adjustment
y	The low-rank approximation to x

Author(s)

jan@deleeuwpdx.net

References

Graffelman, J. and De Leeuw, J. (2023) Improved approximation and visualization of the correlation matrix. The American Statistician, 77(4), 432-442. doi:10.1080/00031305.2023.2186952

https://jansweb.netlify

See Also

ipSymLS

Examples

```
data(HeartAttack)
X <- HeartAttack[,1:7]
X[,7] <- log(X[,7])
colnames(X)[7] <- "logPR"
R <- cor(X)
W <- matrix(1, 7, 7)
diag(W) <- 0
Wals.out <- wAddPCA(R, W, add = "nul", verboseout = FALSE)
Rhat <- Wals.out$y
```

Index