Package ‘MatrixCorrelation’

April 18, 2022

Encoding UTF-8
Type Package
Title Matrix Correlation Coefficients
Version 0.10.0
Date 2022-04-18
Author Kristian Hovde Liland
Maintainer Kristian Hovde Liland <kristian.liland@nmbu.no>
Description Computation and visualization of matrix correlation coefficients.
   The main method is the Similarity of Matrices Index, while various related
   measures like r1, r2, r3, r4, Yanai's GCD, RV, RV2, adjusted RV, Rozeboom's
   linear correlation and Coxhead's coefficient are included
   for comparison and flexibility.

URL https://github.com/khliland/MatrixCorrelation/
BugReports https://github.com/khliland/MatrixCorrelation/issues/
Depends R (>= 3.0.1)
Imports Rcpp, plotrix, pracma, progress, RSpectra
License GPL-2
LazyData TRUE
RoxygenNote 7.1.2
LinkingTo progress, Rcpp, ReppArmadillo
NeedsCompilation yes
Repository CRAN
Date/Publication 2022-04-18 19:24:29 UTC

R topics documented:

allCorrelations .................................................. 2
candy ............................................................. 4
cor.test_eq ...................................................... 4
allCorrelations

Description
Compare all correlation measures in the package (or a subset)

Usage
allCorrelations(
  X1,
  X2,
  ncomp1,
  ncomp2,
  methods = c("SMI", "RV", "RV2", "RVadj", "PSI", "r1", "r2", "r3", "r4", "GCD"),
  digits = 3,
  plot = TRUE,
  xlab = "",
  ylab = "",
  ...
)

Arguments
X1 first matrix to be compared (data.frames are also accepted).
X2 second matrix to be compared (data.frames are also accepted).
ncomp1 maximum number of subspace components from the first matrix.
ncomp2 maximum number of subspace components from the second matrix.
methods character vector containing a subset of the supported methods: "SMI", "RV", "RV2", "RVadj", "PSI", "r1", "r2", "r3", "r4", "GCD".
digits number of digits for numerical output.
plot logical indicating if plotting should be performed (default = TRUE).
allCorrelations

xlab          optional x axis label.
ylab          optional y axis label.
...           additional arguments for SMI or plot.

Details

For each of the coefficients a single scalar is computed to describe the similarity between the two input matrices. Note that some methods requires setting one or two numbers of components.

Value

A single value measuring the similarity of two matrices.

Author(s)

Kristian Hovde Liland

References


See Also

SMI, RV (RV2/RVadj), r1 (r2/r3/r4/GCD).

Examples

```r
X1 <- scale( matrix( rnorm(100*300), 100,300), scale = FALSE)
usv <- svd(X1)
# Remove third principal component from X1 to produce X2
X2 <- usv$s[,,-3] %*% diag(usv$d[-3]) %*% t(usv$v[,,-3])
allCorrelations(X1,X2, ncomp1 = 5,ncomp2 = 5)
```
**candy**  
*Candy data*

**Description**
Measurements from sensory analysis (professional tasting) on a number of candy products obtained by sensory labs. The two labs and the associated data sets are parts of a larger study described in Tomic et al. (2010).

**Usage**
data(candy)

**Format**
Two matrices of dimension 18 x 6.

**References**

---

**cor.test_eq**  
*Test for no correlation between paired samples*

**Description**
Permutation test for squared Pearson correlation between two vectors of samples.

**Usage**
cor.test_eq(x, y, B = 10000)

**Arguments**
x  
first vector to be compared (or two column matrix/data.frame).
y  
second vector to be compared (ommit if included in x).
B  
integer number of permutations, default = 10000.

**Details**
This is a convenience function combining SMI and significant for the special case of vector vs vector comparisons. The null hypothesis is that the correlation between the vectors is +/-1, while significance signifies a deviance toward 0.
Coxhead

Value
A value indicating if the two input vectors are significantly different.

Author(s)
Kristian Hovde Liland

References
Similarity of Matrices Index - Ulf Geir Indahl, Tormod Næs, Kristian Hovde Liland

See Also
plot.SMI (print.SMI/summary.SMI), RV (RV2/RVadj), r1 (r2/r3/r4/GCD), allCorrelations (matrix correlation comparison), PCAv (cross-validated PCA).

Examples
```r
a <- (1:5) + rnorm(5)
b <- (1:5) + rnorm(5)
cor.test_eq(a,b)
```

---

Coxhead

Coxhead's coefficient

Description
Coxhead's coefficient

Usage
```r
Coxhead(X1, X2, weighting = c("sqrt", "min"))
```

Arguments
- **X1**: first matrix to be compared (data.frames are also accepted).
- **X2**: second matrix to be compared (data.frames are also accepted).
- **weighting**: string indicating if weighting should be sqrt(p*q) or min(p, q) (default = 'sqrt').

Value
A single value measuring the similarity of two matrices. For diagnostic purposes it is accompanied by an attribute "canonical.correlation".
References


See Also

SMI, RV (RV2/RVadj), Rozeboom, r1 (r2/r3/r4/GCD).

Examples

```r
X <- matrix(rnorm(100*13),nrow=100)
X1 <- X[, 1:5]  # Random normal
X2 <- X[, 6:12] # Random normal
X2[,1] <- X2[,1] + X[,5] # Overlap in one variable
Coxhead(X1, X2)
```

Description

Computation and visualization of matrix correlation coefficients. The main method is the Similarity of Matrices Index, while various related measures like r1, r2, r3, r4, Yanai’s GCD, RV, RV2, adjusted RV, Rozeboom’s linear correlation and Coxhead’s coefficient are included for comparison and flexibility.

References

PCAcv

See Also

SMI, plot.SMI (print.SMI/summary.SMI), RV (RV2/RVadj), r1 (r2/r3/r4/GCD), Rozeboom, Coxhead,
allCorrelations (matrix correlation comparison).

| PCAcv | Principal Component Analysis cross-validation error |

Description

PRESS values for PCA as implemented by Eigenvector and described by Bro et al. (2008).

Usage

PCAcv(X, ncomp)

Arguments

X matrix object to perform PCA on.

ncomp integer number of components.

Details

For each number of components predicted residual sum of squares are calculated based on leave-one-out cross-validation. The implementation ensures no over-fitting or information bleeding.

Value

A vector of PRESS-values.

Author(s)

Kristian Hovde Liland

References


See Also

plot.SMI (print.SMI/summary.SMI), RV (RV2/RVadj), r1 (r2/r3/r4/GCD), allCorrelations (matrix correlation comparison).

Examples

X1 <- scale( matrix(rnorm(100*300), 100,300), scale = FALSE)
PCAcv(X1,10)
PCAimpute

Principal Component Analysis based imputation

Description

Imputation of missing data, NA, using Principal Component Analysis with iterative refitting and mean value updates. The chosen number of components and convergence parameters (iterations and tolerance) influence the precision of the imputation.

Usage

PCAimpute(X, ncomp, center = TRUE, max_iter = 20, tol = 10^-5)

Arguments

- **X**: matrix object to perform PCA on.
- **ncomp**: integer number of components.
- **center**: logical indicating if centering (default) should be performed.
- **max_iter**: integer number of iterations of PCA if sum of squared change in imputed values is above tol.
- **tol**: numeric tolerance for sum of squared change in imputed values.

Value

Final singular value decomposition, imputed X matrix and convergence metrics (sequence of sum of squared change and number of iterations).

Examples

```r
X <- matrix(rnorm(12), 3, 4)
X[c(2, 6, 10)] <- NA
PCAimpute(X, 3)
```

plot.SMI

Result functions for the Similarity of Matrices Index (SMI)

Description

Plotting, printing and summary functions for SMI, plus significance testing.
Usage

```r
## S3 method for class 'SMI'
plot(
  x,
  y = NULL,
  x1lab = attr(x, "mat.names")[[1]],
  x2lab = attr(x, "mat.names")[[2]],
  main = "SMI",
  signif = 0.05,
  ylim = c(0.5, (sum(pq) + 3)/2),
  B = 10000,
  cex = 1,
  cex.sym = 1,
  frame = NULL,
  frame.col = "red",
  frame.lwd = 2,
  replicates = NULL,
  ... )

## S3 method for class 'SMI'
print(x, ...)

## S3 method for class 'SMI'
summary(object, ...)

is.signif(x, signif = 0.05, B = 10000, ...)
```

Arguments

- **x**: object of class SMI.
- **y**: not used.
- **x1lab**: optional label for first matrix.
- **x2lab**: optional label for second matrix.
- **main**: optional heading (default = SMI).
- **signif**: significance level for testing (default=0.05).
- **xlim**: optional plotting limits.
- **ylim**: optional plotting limits.
- **B**: number of permutations (for significant, default=10000).
- **cex**: optional text scaling (default = 1)
- **cex.sym**: optional scaling for significance symbols (default = 1)
- **frame**: two element integer vector indicating framed components.
- **frame.col**: color for framed components.
frame.lwd: line width for framed components.
replicates: vector of replicates for significance testing.
...: additional arguments for `plot`.
object: object of class `SMI`.

Details

For plotting a diamonad plot is used. High SMI values are light and low SMI values are dark. If orthogonal projections have been used for calculating SMIs, significance symbols are included in the plot unless `signif=NULL`.

Value

`plot` silently returns `NULL`. `print` and `summary` return the printed matrix.

Author(s)

Kristian Hovde Liland

References

Similarity of Matrices Index - Ulf G. Indahl, Tormod Næs, Kristian Hovde Liland

See Also

`SMI`, `PCAcv` (cross-validated PCA).

Examples

```r
X1 <- scale( matrix( rnorm(100*300), 100,300), scale = FALSE)
usv <- svd(X1)
X2 <- usv$u[,3] %*% diag(usv$d[3]) %*% t(usv$v[,3])

smi <- SMI(X1,X2,5,5)
plot(smi, B = 1000) # default B = 10000
print(smi)
summary(smi)
is.signif(smi, B = 1000) # default B = 10000
```
PSI

*Procrustes Similarity Index*

**Description**

An index based on the RV coefficient with Procrustes rotation.

**Usage**

PSI(X1, X2, center = TRUE)

**Arguments**

- **X1**: first matrix to be compared (data.frames are also accepted).
- **X2**: second matrix to be compared (data.frames are also accepted).
- **center**: logical indicating if input matrices should be centered (default = TRUE).

**Value**

The Procrustes Similarity Index

**References**


**Examples**

```r
X1 <- scale( matrix( rnorm(100*300), 100,300), scale = FALSE)
usv <- svd(X1)
X2 <- usv$u[, -3] %*% diag(usv$d[-3]) %*% t(usv$v[, -3])
PSI(X1, X2)
```

---

**Correlational Measures for Matrices**

**Description**

Matrix similarity as described by Ramsey et al. (1984).
Usage

\begin{verbatim}
Usage
\begin{verbatim}
r1(X1, X2, center = TRUE, impute = FALSE)
r2(
  X1,
  X2,
  center = TRUE,
  impute = FALSE,
  impute_par = list(max_iter = 20, tol = 10^-5)
)

r3(
  X1,
  X2,
  center = TRUE,
  impute = FALSE,
  impute_par = list(max_iter = 20, tol = 10^-5)
)

r4(
  X1,
  X2,
  center = TRUE,
  impute = FALSE,
  impute_par = list(max_iter = 20, tol = 10^-5)
)

GCD(
  X1,
  X2,
  ncomp1 = min(dim(X1)),
  ncomp2 = min(dim(X2)),
  center = TRUE,
  impute = FALSE,
  impute_par = list(max_iter = 20, tol = 10^-5)
)
\end{verbatim}
\end{verbatim}
Arguments

\begin{verbatim}
Arguments
\begin{verbatim}
X1 first matrix to be compared (data.frames are also accepted).
X2 second matrix to be compared (data.frames are also accepted).
center logical indicating if input matrices should be centered (default = TRUE).
impute logical indicating if missing values are expected in X1 or X2.
impute_par named list of imputation parameters in case of NAs in X1/X2.
ncomp1 (GCD) number of subspace components from the first matrix (default: full subspace).
\end{verbatim}
\end{verbatim}
ncomp2 (GCD) number of subspace components from the second matrix (default: full subspace).

Details
Details can be found in Ramsey’s paper:

- r1: inner product correlation
- r2: orientation-independent inner product correlation
- r3: spectra-independent inner product correlations (including orientation)
- r4: Spectra-Independent inner product Correlations
- GCD: Yanai’s Generalized Coefficient of Determination (GCD) Measure. To reproduce the original GCD, use all components. When X1 and X2 are dummy variables, GCD is proportional with Pillai’s criterion: tr(W^-1(B+W)).

Value
A single value measuring the similarity of two matrices.

Author(s)
Kristian Hovde Liland

References

See Also
SMI, RV (RV2/RVadj), Rozeboom, Coxhead, allCorrelations (matrix correlation comparison), PCAcv (cross-validated PCA), PCAimpute (PCA based imputation).

Examples
```
X1 <- matrix(rnorm(100*300),100,300)
usv <- svd(X1)
X2 <- usv$u[, -3] %*% diag(usv$d[-3]) %*% t(usv$v[, -3])

r1(X1, X2)
r2(X1, X2)
r3(X1, X2)
r4(X1, X2)
GCD(X1, X2)
GCD(X1, X2, 5, 5)

# Missing data
X1[c(1, 50, 400, 900)] <- NA
X2[c(10, 200, 450, 1200)] <- NA
r1(X1, X2, impute = TRUE)
r2(X1, X2, impute = TRUE)
```
Rozeboom

Rozeboom’s squared vector correlation

Description

Rozeboom’s squared vector correlation

Usage

Rozeboom(X1, X2)

sqveccor(X1, X2)

Arguments

X1 first matrix to be compared (data.frames are also accepted).
X2 second matrix to be compared (data.frames are also accepted).

Value

A single value measuring the similarity of two matrices. For diagnostic purposes it is accompanied by an attribute "canonical.correlation".

Author(s)

Korbinian Strimmer and Kristian Hovde Liland

References


See Also

SMI, RV (RV2/RVadj), Coxhead, r1 (r2/r3/r4/GCD).
Examples

```r
X <- matrix(rnorm(100*13), nrow=100)
X1 <- X[, 1:5]  # Random normal
X2 <- X[, 6:12]  # Random normal
X2[,1] <- X2[,1] + X[,5]  # Overlap in one variable
Rozeboom(X1, X2)
```

<table>
<thead>
<tr>
<th>RV</th>
<th>RV coefficients</th>
</tr>
</thead>
</table>

Description

Three different RV coefficients: RV, RV2 and adjusted RV.

Usage

- `RV(X1, X2, center = TRUE, impute = FALSE)`
- `RV2(X1, X2, center = TRUE, impute = FALSE)`
- `RVadjMaye(X1, X2, center = TRUE)`
- `RVadjGhaziri(X1, X2, center = TRUE)`
- `RVadj(X1, X2, version = c("Maye", "Ghaziri"), center = TRUE)`

Arguments

- `X1` first matrix to be compared (data.frames are also accepted).
- `X2` second matrix to be compared (data.frames are also accepted).
- `center` logical indicating if input matrices should be centered (default = TRUE).
- `impute` logical indicating if missing values are expected in `X1` or `X2` (only for RV and RV2).
- `version` Which version of RV adjusted to apply: "Maye" (default) or "Ghaziri" RV adjusted is run using the `RVadj` function.

Details

For each of the four coefficients a single scalar is computed to describe the similarity between the two input matrices.

Value

A single value measuring the similarity of two matrices.
Author(s)
Kristian Hovde Liland, Benjamin Leutner (RV2)

References


See Also
SMI, r1 (r2/r3/r4/GCD), Rozeboom, Coxhead, allCorrelations (matrix correlation comparison), PCAcv (cross-validated PCA), PCAimpute (PCA based imputation).

Examples

```r
X1 <- matrix(rnorm(100*300),100,300)
usv <- svd(X1)
X2 <- usv$u[,,-3] %*% diag(usv$d[-3]) %*% t(usv$v[,,-3])

RV(X1,X2)
RV2(X1,X2)
RVadj(X1,X2)

# Missing data
X1[1, 50, 400, 900] <- NA
X2[10, 200, 450, 1200] <- NA
RV(X1,X2, impute = TRUE)
RV2(X1,X2, impute = TRUE)
```

significant

Significance estimation for Similarity of Matrices Index (SMI)

Description

Permutation based hypothesis testing for SMI. The null hypothesis is that a linear function of one matrix subspace is included in the subspace of another matrix.

Usage

```r
significant(smi, B = 10000, replicates = NULL)
```
SMI

Arguments

smi     smi object returned by call to SMI.
B       integer number of permutations, default = 10000.
replicates integer vector of replicates.

Details

For each combination of components significance is estimated by sampling from a null distribution of no similarity, i.e. when the rows of one matrix is permuted B times and corresponding SMI values are computed. If the vector replicates is included, replicates will be kept together through permutations.

Value

A matrix containing P-values for all combinations of components.

Author(s)

Kristian Hovde Liland

References

Similarity of Matrices Index - Ulf G. Indahl, Tormod Næs Kristian Hovde Liland

See Also

plot.SMI (print.SMI/summary.SMI), RV (RV2/RVadj), r1 (r2/r3/r4/GCD), allCorrelations (matrix correlation comparison).

Examples

X1 <- scale(matrix(rnorm(100*300), 100,300), scale = FALSE)
usv <- svd(X1)
X2 <- usv$u[,-3] %*% diag(usv$d[-3]) %*% t(usv$v[,-3])

(smi <- SMI(X1,X2,5,5))
significant(smi, B = 1000) # default B = 10000

SMI

Similarity of Matrices Index (SMI)

Description

A similarity index for comparing coupled data matrices.
Usage

SMI(
  X1,
  X2,
  ncomp1 = Rank(X1) - 1,
  ncomp2 = Rank(X2) - 1,
  projection = "Orthogonal",
  Scores1 = NULL,
  Scores2 = NULL,
  impute = FALSE,
  impute_par = list(max_iter = 20, tol = 10^-5)
)

Arguments

X1          first matrix to be compared (data.frames are also accepted).
X2          second matrix to be compared (data.frames are also accepted).
ncomp1      maximum number of subspace components from the first matrix.
ncomp2      maximum number of subspace components from the second matrix.
projection  type of projection to apply, defaults to "Orthogonal", alternatively "Procrustes".
Scores1     user supplied score-matrix to replace singular value decomposition of first matrix.
Scores2     user supplied score-matrix to replace singular value decomposition of second matrix.
impute      logical for activation of PCA based imputation for X1/X2.
impute_par  named list of imputation parameters in case of NAs in X1/X2.

Details

A two-step process starts with extraction of stable subspaces using Principal Component Analysis or some other method yielding two orthonormal bases. These bases are compared using Orthogonal Projection (OP / ordinary least squares) or Procrustes Rotation (PR). The result is a similarity measure that can be adjusted to various data sets and contexts and which includes explorative plotting and permutation based testing of matrix subspace equality.

Value

A matrix containing all combinations of components. Its class is "SMI" associated with print, plot, summary methods.

Author(s)

Kristian Hovde Liland

References

See Also

plot.SMI (print.SMI/summary.SMI), RV (RV2/RVadj), r1 (r2/r3/r4/GCD), Rozeboom, Coxhead, allCorrelations (matrix correlation comparison), PCAcv (cross-validated PCA), PCAimpute (PCA based imputation).

Examples

# Simulation
X1 <- scale(matrix(rnorm(100*300), 100,300), scale = FALSE)
usv <- svd(X1)
X2 <- usv$u[,3] %*% diag(usv$d[,3]) %*% t(usv$v[,3])

(smi <- SMI(X1,X2,5,5))
plot(smi, B = 1000 ) # default B = 10000

# Sensory analysis
data(candy)
plot( SMI(candy$Panel1, candy$Panel2, 3,3, projection = "Procrustes"),
     frame = c(2,2), B = 1000, x1lab = "Panel1", x2lab = "Panel2" ) # default B = 10000

# Missing data (100 missing completely at random points each)
X1[sort(round(runif(100)*29999+1))] <- NA
X2[sort(round(runif(100)*29999+1))] <- NA
(smi <- SMI(X1,X2,5,5, impute = TRUE))
Index

allCorrelations, 2, 5, 7, 13, 16, 17, 19

candy, 4
cor.test_eq, 4
Coxhead, 5, 7, 13, 14, 16, 19

GCD (r1), 11

is.signif (plot.SMI), 8

MatrixCorrelation, 6
MatrixCorrelation-package (MatrixCorrelation), 6

PCAcv, 5, 7, 10, 13, 16, 19
PCAimpute, 8, 13, 16, 19
plot.SMI, 5, 7, 8, 17, 19
print.SMI (plot.SMI), 8
PSI, 11

r1, 3, 5–7, 11, 14, 16, 17, 19
r2 (r1), 11
r3 (r1), 11
r4 (r1), 11
Rozeboom, 6, 7, 13, 14, 16, 19
RV, 3, 5–7, 13, 14, 15, 17, 19
RV2 (RV), 15
RVadj (RV), 15
RVadjGhaziri (RV), 15
RVadjMaye (RV), 15

significant, 16
SMI, 3, 6, 7, 10, 13, 14, 16, 17
sqveccor (Rozeboom), 14
summary.SMI (plot.SMI), 8