

# Package ‘BMconcor’

May 6, 2026

**Title** CONCOR for Structural- And Regular-Equivalence Blockmodeling

**Version** 2.0.0

**Description** The four functions `svdcp()` ('cp' for column partitioned), `svdbip()` or `svdbip2()` ('bip' for bi-partitioned), and `svdbips()` ('s' for a simultaneous optimization of a set of 'r' solutions), correspond to a singular value decomposition (SVD) by blocks notion, by supposing each block depending on relative subspaces, rather than on two whole spaces as usual SVD does. The other functions, based on this notion, are relative to two column partitioned data matrices  $x$  and  $y$  defining two sets of subsets  $x_i$  and  $y_j$  of variables and amount to estimate a link between  $x_i$  and  $y_j$  for the pair  $(x_i, y_j)$  relatively to the links associated to all the other pairs. These methods were first presented in: Lafosse R. & Hanafi M.,(1997) <<https://eudml.org/doc/106424>> and Hanafi M. & Lafosse, R. (2001) <<https://eudml.org/doc/106494>>.

**License** GPL (>= 3)

**URL** <https://fatelarico.github.io/BMconcor/>

**Encoding** UTF-8

**RoxygenNote** 7.3.1

**NeedsCompilation** no

**Author** Roger Lafosse [aut],  
Fabio Ashtar Telarico [cre, aut] (ORCID:  
<<https://orcid.org/0000-0002-8740-7078>>)

**Maintainer** Fabio Ashtar Telarico <[Fabio-Ashtar.Telarico@fdv.uni-lj.si](mailto:Fabio-Ashtar.Telarico@fdv.uni-lj.si)>

**Repository** CRAN

**Date/Publication** 2024-05-02 13:22:42 UTC

## Contents

<code>concor</code> . . . . .	2
<code>concorcano</code> . . . . .	3
<code>concoreg</code> . . . . .	4
<code>concorgm</code> . . . . .	5
<code>concorgmcano</code> . . . . .	7
<code>concorgmreg</code> . . . . .	8

concors	9
concorscano	10
concorsreg	12
svdbip	13
svdbip2	14
svdbips	15
svdcp	16

<b>Index</b>	<b>18</b>
--------------	-----------

---

concor *Relative links of several subsets of variables*

---

## Description

Relative links of several subsets of variables  $Y_j$  with another set  $X$ . SUCCESSIVE SOLUTIONS

## Usage

concor(x, y, py, r)

## Arguments

x	are the n times p and n times q matrices of p and q centered column
y	See x
py	The partition vector of y. A row vector containing the numbers $q_i$ for $i = 1, \dots, k_y$ of the $k_y$ subsets $y_i$ of $y$ : $\sum(q_i) = \sum(py) = q$ .
r	The number of wanted successive solutions

## Details

The first solution calculates  $1+k_x$  normed vectors: the vector  $u[:, 1]$  of  $R_p$  associated to the  $k_y$  vectors  $v_i[:, 1]$ 's of  $R_{q_i}$ , by maximizing  $\sum_i \text{cov}(x * u[:, k], y_i * v_i[:, k])^2$ , with  $1+k_y$  norm constraints on the axes. A component  $(x)(u[:, k])$  is associated to  $k_y$  partial components  $(y_i)(v_i[:, k])$  and to a global component  $y * V[:, k]$ .  $\text{cov}((x)(u[:, k]), (y)(V[:, k]))^2 = \sum \text{cov}((x)(u[:, k]), (y_i)(v_i[:, k]))^2$ .  $(y)(V[:, k])$  is a global component of the components  $(y_i)(v_i[:, k])$ . The second solution is obtained from the same criterion, but after replacing each  $y_i$  by  $y_i - (y_i)(v_i[:, 1])(v_i[:, 1]')$ . And so on for the successive solutions 1,2,...,r. The biggest number of solutions may be  $r = \inf(n, p, q_i)$ , when the  $(x')(y_i')(s)$  are supposed with full rank; then  $r_{\max} = \min(c(\min(py), n, p))$ . For a set of r solutions, the matrix  $u'X'YV$  is diagonal and the matrices  $u'X'Y_j v_j$  are triangular (good partition of the link by the solutions). concor.m is the svdcp.m function applied to the matrix  $x'y$ .

**Value**

A list with following components:

u                    A p times r matrix of axes in  $R_p$  relative to x;  $(u^{\text{prime}})(u) = \text{Identity}$   
v                    A q times r matrix of  $k_y$  row blocks  $v_i$  ( $q_i \times r$ ) of axes in  $R_{q_i}$  relative to  $y_i$ ;  
 $v_i^{\text{prime}} * v_i = \text{Identity}$   
V                    A q times r matrix of axes in  $R_q$  relative to y;  $V^{\text{prime}} * V = \text{Identity}$   
cov2                A  $k_y$  times r matrix; each column k contains  $k_y$  squared covariances  $\text{cov}(x * u[, k], y_i * v_i[, k])^2$ , the partial measures of link

**Author(s)**

Lafosse, R.

**References**

Lafosse R. & Hanafi M.(1997) Concordance d'un tableau avec K tableaux: Definition de K+1 uples synthetiques. Revue de Statistique Appliquee vol.45,n.4.

**Examples**

```
# To make some "GPA" : so, by posing the compromise X = Y,
# "procrustes" rotations to the "compromise X" then are :
# Yj*(vj*u').
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
co <- concor(x,y,c(3,2,4),2)
```

---

concorcano

*Canonical analysis of several sets with another set*


---

**Description**

Relative proximities of several subsets of variables  $Y_j$  with another set X. SUCCESSIVE SOLUTIONS

**Usage**

```
concorcano(x, y, py, r)
```

**Arguments**

x                    are the n times p and n times q matrices of p and q centered column  
y                    See x  
py                   The partition vector of y. A row vector containing the numbers  $q_i$  for  $i = 1, \dots, k_y$  of the  $k_y$  subsets  $y_i$  of y :  $\sum(q_i) = \sum(py) = q$ .  
r                    The number of wanted successive solutions

**Details**

The first solution calculates a standardized canonical component  $cx[, 1]$  of  $x$  associated to  $ky$  standardized components  $cyi[, 1]$  of  $y_i$  by maximizing  $\sum_i \rho(cx[, 1], cy_i[, 1])^2$ . The second solution is obtained from the same criterion, with  $ky$  orthogonality constraints for having  $\rho(cy_i[, 1], cy_i[, 2])=0$  (that implies  $\rho(cx[, 1], cx[, 2])=0$ ). For each of the  $1+ky$  sets, the  $r$  canonical components are 2 by 2 zero correlated. The  $ky$  matrices  $(cx)'*cy_i$  are triangular. This function uses `concor` function.

**Value**

A list with following components:

<code>cx</code>	a $n$ times $r$ matrix of the $r$ canonical components of $x$
<code>cy</code>	a $n.ky$ times $r$ matrix. The $ky$ blocks $cy_i$ of the rows $n*(i-1)+1 : n*i$ contain the $r$ canonical components relative to $Y_i$
<code>rho2</code>	a $ky$ times $r$ matrix; each column $k$ contains $ky$ squared canonical correlations $\rho(cx[, k], cy_i[, k])^2$

**Author(s)**

Lafosse, R.

**References**

Hanafi & Lafosse (2001) Generalisation de la regression lineaire simple pour analyser la dependance de  $K$  ensembles de variables avec un  $K+1$  eme. *Revue de Statistique Appliquee* vol.49, n.1

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
ca <- concorcano(x,y,c(3,2,4),2)
```

---

concoreg

*Redundancy of sets  $y_j$  by one set  $x$*

---

**Description**

Regression of several subsets of variables  $Y_j$  by another set  $X$ . SUCCESSIVE SOLUTIONS

**Usage**

```
concoreg(x, y, py, r)
```

**Arguments**

x	are the n times p and n times q matrices of p and q centered column
y	See x
py	The partition vector of y. A row vector containing the numbers qi for i = 1, . . . , ky of the ky subsets yi of y : $\sum(q_i) = \sum(py) = q$ .
r	The number of wanted successive solutions

**Value**

A list with following components:

cx	a n times r matrix of the r explanatory components
v	is a $q \times r$ matrix of ky row blocks $v_i$ ( $q_i \times r$ ) of axes in $R_{q_i}$ relative to $y_i$ ; $v_i' * v_i = Id$
V	is a $q \times r$ matrix of axes in $R_q$ relative to y; $V' * V = Id$
varexp	is a $ky \times r$ matrix; each column k contains ky explained variances $\rho(cx[k], y_i * v_i[k])^2 \text{var}(y_i * v_i[k])$

**Author(s)**

Lafosse, R.

**References**

Lafosse R. & Hanafi M. (1997) Concordance d'un tableau avec K tableaux: Definition de K+1 uples synthetiques. *Revue de Statistique Appliquee* vol.45,n.4.

Chessel D. & Hanafi M. (1996) Analyses de la Co-inertie de K nuages de points. *Revue de Statistique Appliquee* vol.44, n.2. (this ACOM analysis of one multiset is obtained by the command : `concoreg(Y,Y,py,r)`)

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
co <- concoreg(x,y,c(3,2,4),2)
```

---

concorgm

*Analyzing a set of partial links between Xi and Yj*

---

**Description**

Analyzing a set of partial links between Xi and Yj, SUCCESSIVE SOLUTIONS

**Usage**

```
concorgm(x, px, y, py, r)
```

**Arguments**

x	are the n times p and n times q matrices of p and q centered column
px	A row vector which contains the numbers $p_i$ , $i=1, \dots, k_x$ , of the $k_x$ subsets $x_i$ of $x$ : $\sum(p_i)=\sum(px)=p$ . $px$ is the partition vector of $x$
y	See $x$
py	The partition vector of $y$ . A row vector containing the numbers $q_i$ for $i = 1, \dots, k_y$ of the $k_y$ subsets $y_i$ of $y$ : $\sum(q_i)=\sum(py)=q$ .
r	The number of wanted successive solutions $r_{max} \leq \min(\min(px), \min(py), n)$

**Details**

The first solution calculates  $1+k_x$  normed vectors: the vector  $u[:, 1]$  of  $R_p$  associated to the  $k_y$  vectors  $v_i[:, 1]$ 's of  $R_{q_i}$ , by maximizing  $\sum(\text{cov}((x)(u[, k]), (y_i)(v_i[, k]))^2)$ , with  $1+k_y$  norm constraints on the axes. A component  $(x)(u[, k])$  is associated to  $k_y$  partial components  $(y_i)(v_i)[, k]$  and to a global component  $y * v[, k]$ .  $\text{cov}((x)(u[, k]), (y)(v[, k]))^2 = \sum(\text{cov}((x)(u[, k]), (y_i)(v_i[, k]))^2) (y)(v[, k])$  is a global component of the components  $(y_i)(v_i[, k])$ . The second solution is obtained from the same criterion, but after replacing each  $y_i$  by  $y_i - (y_i)(v_i[, 1])(v_i[, 1])'$ . And so on for the successive solutions  $1, 2, \dots, r$ . The biggest number of solutions may be  $r = \inf(n, p, q_i)$ , when the  $(x')(y_i')(s)$  are supposed with full rank; then  $r_{max} = \min(c(\min(py), n, p))$ . For a set of  $r$  solutions, the matrix  $u'X'YV$  is diagonal and the matrices  $u'X'Y_j v_j$  are triangular (good partition of the link by the solutions). `concor.m` is the `svdcp.m` function applied to the matrix  $x'y$ .

**Value**

A list with following components:

u	a p times r matrix of axes in $R_p$ relative to $x$ ; $u^{\prime} * u = \text{Identity}$
v	a q times r matrix of $k_y$ row blocks $v_i$ ( $q_i \times r$ ) of axes in $R_{q_i}$ relative to $y_i$ ; $v_i^{\prime} * v_i = \text{Identity}$
cov2	a $k_y$ times r matrix; each column k contains $k_y$ squared covariances $\text{cov}((x)(u[, k]), (y_i)(v_i[, k]))^2$ , the partial measures of link

**Author(s)**

Lafosse, R.

**References**

Kissita, Cazes, Hanafi & Lafosse (2004) Deux methodes d'analyse factorielle du lien entre deux tableaux de variables partitionn?es. *Revue de Statistique Appliqu?e*, Vol 52, n. 3, 73-92.

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
cg <- concorgm(x,c(2,3),y,c(3,2,4),2)
cg$cov2[1,1,]
```

concorgmcano

*Canonical analysis of subsets Yj with subsets Xi***Description**

Canonical analysis of subsets Yj with subsets Xi. Relative valuations by squared correlations of the proximities of subsets Xi with subsets Yj. SUCCESSIVE SOLUTIONS

**Usage**

```
concorgmcano(x, px, y, py, r)
```

**Arguments**

x are the n times p and n times q matrices of p and q centered column  
 px The row vector which contains the numbers pi, i=1,...,kx, of the kx subsets xi of x :  $\sum_i p_i = \text{sum}(px) = p$ . px is the partition vector of x  
 y See x  
 py The partition vector of y. A row vector containing the numbers qi for i = 1, . . . , ky of the ky subsets yi of y :  $\text{sum}(q_i) = \text{sum}(py) = q$ .  
 r The number of wanted successive solutions  $r_{\max} \leq \min(\min(px), \min(py), n)$

**Details**

For the first solution,  $\text{sum}_i \text{sum}_j \text{rho2}(cx_i[, 1], cy_j[, 1])$  is the optimized criterion. The other solutions are calculated from the same criterion, but with orthogonalities for having two by two zero correlated the canonical components defined for each xi, and also for those defined for each yj. Each solution associates kx canonical components to ky canonical components. When kx = 1 (px=p), take concorcano function This function uses the concorgm function

**Value**

A list with following components:

cx is a n.kx times r matrix of kx row blocks cxi (n x r). Each row block contains r partial canonical components  
 cy is a n.ky times r matrix of ky row blocks cyj (n x r). Each row block contains r partial canonical components  
 rho2 is a kx time ky tims r array; for a fixed solution k, rho2[, , k] contains kxky squared correlations  $\text{rho2}(cx[n*(i-1)+1 : n*i, k], cy[n*(j-1)+1 : n*j, k])$ , simultaneously calculated between all the yj with all the xi

**Author(s)**

Lafosse, R.

**References**

Kissita G., Analyse canonique generalisee avec tableau de reference generalisee. Thesis, Ceremade Paris 9 Dauphine (2003).

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
cc <- concordmcano(x,c(2,3),y,c(3,2,4),2)
cc$rho2[1,1,]
```

concordmreg

*Regression of subsets Yj by subsets Xi***Description**

Regression of subsets Yj by subsets Xi for comparing all the explanatory-explained pairs (Xi,Yj).  
SUCCESSIVE SOLUTIONS

**Usage**

```
concordmreg(x, px, y, py, r)
```

**Arguments**

x	are the n times p and n times q matrices of p and q centered column
px	A row vector which contains the numbers pi, i = 1,...,kx, of the kx subsets xi of x : sum(pi)=sum(px)=p. px is the partition vector of x
y	See x
py	The partition vector of y. A row vector containing the numbers qi for i = 1, . . . ,ky of the ky subsets yi of y : sum(qi)=sum(py)=q.
r	The number of wanted successive solutions

**Details**

For the first solution,  $\sum_i \sum_j \rho_2(cx_i[, 1], y_j * v_j[, 1]) \text{var}(y_j * v_j[, 1])$  is the optimized criterion. The second solution is calculated from the same criterion, but with  $y_j - y_j * v_j[, 1] * v_j[, 1]'$  instead of the matrices yj and with orthogonalities for having two by two zero correlated the explanatory components defined for each matrix xi. And so on for the other solutions. One solution k associates kx explanatory components (in cx[, k]) to ky explained components. When kx = 1 (px = p), take concordmreg function This function uses the concordm function

**Value**

A list with following components:

`cx` a  $n$  times  $r$  matrix of the  $r$  explanatory components  
`v` is a  $q \times r$  matrix of  $k_y$  row blocks  $v_i$  ( $q_i \times r$ ) of axes in  $R_{qi}$  relative to  $y_i$ ;  $v_i' * v_i = Id$   
`varexp` is a  $k_x \times k_y \times r$  array; for a fixed solution  $k$ , the matrix `varexp[, , k]` contains  $k_x k_y$  explained variances obtained by a simultaneous regression of all the  $y_j$  by all the  $x_i$ , so the values  $\rho_2(cx[n*(i-1)+1 : n*i, k], y_j * v_j[, k]) \text{var}(y_j * v_j[, k])$

**Author(s)**

Lafosse, R.

**References**

Hanafi & Lafosse (2004) Regression of a multi-set by another based on an extension of the SVD. COMPSTAT'2004 Symposium

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
cr <- concormreg(x,c(2,3),y,c(3,2,4),2)
cr$varexp[1,1,]
```

---

concors

*simultaneous concorm*

---

**Description**

concorm with the set of  $r$  solutions simultaneously optimized

**Usage**

```
concors(x, px, y, py, r)
```

**Arguments**

`x` are the  $n$  times  $p$  and  $n$  times  $q$  matrices of  $p$  and  $q$  centered column  
`px` A row vector which contains the numbers  $p_i$ ,  $i=1, \dots, k_x$ , of the  $k_x$  subsets  $x_i$  of  $x$  :  $\sum(p_i)=\sum(px)=p$ . `px` is the partition vector of  $x$   
`y` See `x`  
`py` The partition vector of  $y$ . A row vector containing the numbers  $q_i$  for  $i = 1, \dots, k_y$  of the  $k_y$  subsets  $y_i$  of  $y$  :  $\sum(q_i)=\sum(py)=q$ .  
`r` The number of wanted successive solutions  $r_{max} \leq \min(\min(px), \min(py), n)$

**Details**

This function uses the svdbips function

**Value**

A list with following components:

u	a p times r matrix of axes in $R^p$ relative to x; $u^{\prime}u = \text{Identity}$
v	a q times r matrix of $k_y$ row blocks $v_i$ ( $q_i \times r$ ) of axes in $R^{q_i}$ relative to $y_i$ ; $v_i^{\prime}v_i = \text{Identity}$
cov2	a $k_y$ times r matrix; each column k contains $k_y$ squared covariances $\text{cov}(x * u[, k], y_i * v_i[, k])^2$ , the partial measures of link

**Author(s)**

Lafosse, R.

**References**

Lafosse R. & Hanafi M.(1997) Concordance d'un tableau avec K tableaux: Definition de K+1 uples synthetiques. Revue de Statistique Appliquee vol.45,n.4.

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
cs <- concors(x,c(2,3),y,c(3,2,4),2)
cs$cov2[1,1,]
```

---

concorscano

*simultaneous concorgmcano*

---

**Description**

concorgmcano with the set of r solutions simultaneously optimized

**Usage**

```
concorscano(x, px, y, py, r)
```

**Arguments**

<code>x</code>	are the $n$ times $p$ and $n$ times $q$ matrices of $p$ and $q$ centered column
<code>px</code>	A row vector which contains the numbers $p_i$ , $i=1,\dots,k_x$ , of the $k_x$ subsets $x_i$ of $x$ : $\sum(p_i)=\sum(px)=p$ . $px$ is the partition vector of $x$
<code>y</code>	See $x$
<code>py</code>	The partition vector of $y$ . A row vector containing the numbers $q_i$ for $i = 1, \dots, k_y$ of the $k_y$ subsets $y_i$ of $y$ : $\sum(q_i)=\sum(py)=q$ .
<code>r</code>	The number of wanted successive solutions $r_{max} \leq \min(\min(px), \min(py), n)$

**Details**

This function uses the `concors` function

**Value**

A list with following components:

<code>cx</code>	a $n$ times $r$ matrix of the $r$ canonical components of $x$
<code>cy</code>	a $n.k_y$ times $r$ matrix. The $k_y$ blocks $cy_i$ of the rows $n*(i-1)+1 : n*i$ contain the $r$ canonical components relative to $Y_i$
<code>cov2</code>	a $k_y$ times $r$ matrix; each column $k$ contains $k_y$ squared covariances $\text{cov}(x * u[, k], y_i * v_i[, k])^2$ , the partial measures of link

**Author(s)**

Lafosse, R.

**References**

Hanafi & Lafosse (2001) Generalisation de la regression lineaire simple pour analyser la dependance de  $K$  ensembles de variables avec un  $K+1$  eme. *Revue de Statistique Appliquee* vol.49, n.1

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
cca <- concorscano(x,c(2,3),y,c(3,2,4),2)
cca$rho2[1,1,]
```

concorsreg

*Redundancy of sets yj by one set x***Description**

Regression of several subsets of variables Yj by another set X. SUCCESSIVE SOLUTIONS

**Usage**

```
concorsreg(x, px, y, py, r)
```

**Arguments**

**x** are the n times p and n times q matrices of p and q centered column

**px** The row vector which contains the numbers  $p_i$ ,  $i = 1, \dots, k_x$ , of the  $k_x$  subsets  $x_i$  of  $x$  :  $\sum_i p_i = \text{sum}(px) = p$ .  $px$  is the partition vector of  $x$

**y** See  $x$

**py** The partition vector of  $y$ . A row vector containing the numbers  $q_i$  for  $i = 1, \dots, k_y$  of the  $k_y$  subsets  $y_i$  of  $y$  :  $\text{sum}(q_i) = \text{sum}(py) = q$ .

**r** The number of wanted successive solutions

**Value**

A list with following components:

**cx** a n times r matrix of the r explanatory components

**v** is a  $q \times r$  matrix of  $k_y$  row blocks  $v_i$  ( $q_i \times r$ ) of axes in  $R^{q_i}$  relative to  $y_i$ ;  
 $v_i' * v_i = \text{Id}$

**varexp** is a  $k_y \times r$  matrix; each column  $k$  contains  $k_y$  explained variances  $\rho(cx[, k], y_i * v_i[, k])^2 \text{var}(y_i * v_i[, k])$

**Author(s)**

Lafosse, R.

**Examples**

```
x <- matrix(runif(50),10,5);y <- matrix(runif(90),10,9)
x <- scale(x);y <- scale(y)
crs <- concorsreg(x,c(2,3),y,c(3,2,4),2)
crs$varexp[1,1,]
```

---

svdbip *SVD for one bipartitioned matrix x*

---

**Description**

SVD for bipartitioned matrix x. r successive Solutions

**Usage**

svdbip(x, K, H, r)

**Arguments**

x a p times q matrix  
 K is a row vector which contains the numbers pk, k=1,...,kx, of the partition of x with kx row blocks : sum(pk)=p  
 H is a row vector which contains the numbers qh, h=1,...,ky, of the partition of x with ky column blocks : sum(qh)=q  
 r The number of wanted successive solutions

**Details**

The first solution calculates kx+ky normed vectors: kx vectors  $u_k[:, 1]$  of  $R^{p_k}$  associated to ky vectors  $v_h[:, 1]$ 's of  $R^{q_h}$ , by maximizing  $\sum_k \sum_h (u_k[:, 1]^{prime} * x_{kh} * v_h[:, 1])^2$ , with kx+ky norm constraints. A value  $(u_k[:, 1]^{prime} * x_{kh} * v_h[:, 1])^2$  measures the relative link between  $R^{p_k}$  and  $R^{q_h}$  associated to the block xkh. The second solution is obtained from the same criterion, but after replacing each xkh by  $xkh - xkhv_hv_h' - ukuk'xkh + ukuk'xkhv_hv_h'$ . And so on for the successive solutions 1,2,...,r . The biggest number of solutions may be  $r = \inf(pk, qh)$ , when the xkh's are supposed with full rank; then  $r_{max} = \min([\min(K), \min(H)])$ . When K=p (or H=q, with t(x)), svdcp function is better. When H=q and K=p, it is the usual svd (with squared singular values). Convergence of algorithm may be not global. So the below proposed initialisation of the algorithm may be not very suitable for some data sets. Several different random initialisations with normed vectors might be considered and the best result then choosen.

**Value**

A list with following components:

u a p times r matrix of kx row blocks uk (pk x r);  $uk' * uk = Identity$ .  
 v a q times r matrix of ky row blocks vi (qi x r) of axes in Rqi relative to yi ;  $vi^{prime} * vi = Identity$   
 s a kx times ky times r array; with r fixed, each matrix contains kxky values  $(u_h' * x_{kh} * v_k)^2$ , the partial (squared) singular values relative to xkh.

**Author(s)**

Lafosse, R.

## References

Kissita G., Cazes P., Hanafi M. & Lafosse (2004) Deux methodes d'analyse factorielle du lien entre deux tableaux de variables partitionees. Revue de Statistique Appliquee.

## Examples

```
x <- matrix(runif(200),10,20)
s <- svdbip(x,c(3,4,3),c(5,15),3)
```

---

 svdbip2

*SVD for bipartitioned matrix x*


---

## Description

SVD for bipartitioned matrix x. r successive Solutions. As SVDBIP, but with another algorithm and another initialisation

## Usage

```
svdbip2(x, K, H, r)
```

## Arguments

x	a p times q matrix
K	is a row vector which contains the numbers pk, k=1,...,kx, of the partition of x with kx row blocks : sum(pk)=p
H	is a row vector which contains the numbers qh, h=1,...,ky, of the partition of x with ky column blocks : sum(qh)=q
r	The number of wanted successive solutions

## Details

The first solution calculates  $kx+ky$  normed vectors:  $kx$  vectors  $uk[:, 1]$  of  $R^{pk}$  associated to  $ky$  vectors  $vh[:, 1]$ 's of  $R^{qh}$ , by maximizing  $\sum_k \sum_h (u_k[:, 1]' * x_{kh} * v_h[:, 1])^2$ , with  $kx+ky$  norm constraints. A value  $(u_k[:, 1]' * x_{kh} * v_h[:, 1])^2$  measures the relative link between  $R^{pk}$  and  $R^{qh}$  associated to the block  $x_{kh}$ . The second solution is obtained from the same criterion, but after replacing each  $x_{kh}$  by  $x_{kh}-x_{kh}v_hv_h'-ukuk'x_{kh}+ukuk'x_{kh}v_hv_h'$ . And so on for the successive solutions 1,2,...,r . The biggest number of solutions may be  $r=\inf(pk,qh)$ , when the  $x_{kh}$ 's are supposed with full rank; then  $r_{\max}=\min([\min(K), \min(H)])$ . When  $K=p$  (or  $H=q$ , with  $t(x)$ ), svdcp function is better. When  $H=q$  and  $K=p$ , it is the usual svd (with squared singular values). Convergence of algorithm may be not global. So the below proposed initialisation of the algorithm may be not very suitable for some data sets. Several different random initialisations with normed vectors might be considered and the best result then choosen

**Value**

A list with following components:

- u a  $p$  times  $r$  matrix of  $k_x$  row blocks  $u_k$  ( $p_k \times r$ );  $u_k' * u_k = \text{Identity}$ .
- v a  $q$  times  $r$  matrix of  $k_y$  row blocks  $v_i$  ( $q_i \times r$ ) of axes in  $R_{q_i}$  relative to  $y_i$ ;  $v_i^{\text{prime}} * v_i = \text{Identity}$
- s a  $k_x$  times  $k_y$  times  $r$  array; with  $r$  fixed, each matrix contains  $k_x k_y$  values  $(u_h' * x_{kh} * v_k)^2$ , the partial (squared) singular values relative to  $x_{kh}$ .

**Author(s)**

Lafosse, R.

**References**

Kissita G., Analyse canonique generalisee avec tableau de reference generalisee. Thesis, Ceremade Paris 9 Dauphine (2003)

**Examples**

```
x <- matrix(runif(200),10,20)
s2 <- svdbip2(x,c(3,4,3),c(5,5,10),3);s2$s2
s1 <- svdbip(x,c(3,4,3),c(5,5,10),3);s1$s1
```

---

 svdbips

*SVD for bipartitioned matrix x*


---

**Description**

SVD for bipartitioned matrix x. SIMULTANEOUS SOLUTIONS. ("simultaneous svdbip")

**Usage**

```
svdbips(x, K, H, r)
```

**Arguments**

- x a  $p$  times  $q$  matrix
- K is a row vector which contains the numbers  $p_k$ ,  $k=1,\dots,k_x$ , of the partition of  $x$  with  $k_x$  row blocks :  $\text{sum}(p_k)=p$
- H is a row vector which contains the numbers  $q_h$ ,  $h=1,\dots,k_y$ , of the partition of  $x$  with  $k_y$  column blocks :  $\text{sum}(q_h)=q$
- r The number of wanted successive solutions

**Details**

One set of  $r$  solutions is calculated by maximizing  $\sum_i \sum_k \sum_h (u_k[,i]' * x_{kh} * v_h[,i])^2$ , with  $kx+ky$  orthonormality constraints (for each  $u_k$  and each  $v_h$ ). For each fixed  $r$  value, the solution is totally new (does'nt consist to complete a previous calculus of one set of  $r-1$  solutions).  $r_{max} = \min([\min(K), \min(H)])$ . When  $r=1$ , it is svdbip (thus it is svdcp when  $r=1$  and  $kx=1$ ). Convergence of algorithm may be not global. So the below proposed initialisation of the algorithm may be not very suitable for some data sets. Several different random initialisations with normed vectors might be considered and the best result then chosen....

**Value**

A list with following components:

- $u$  a  $p$  times  $r$  matrix of  $kx$  row blocks  $u_k$  ( $pk \times r$ );  $u_k' * u_k = \text{Identity}$ .
- $v$  a  $q$  times  $r$  matrix of  $ky$  row blocks  $v_i$  ( $qi \times r$ ) of axes in  $R_{qi}$  relative to  $y_i$ ;  $v_i' * v_i = \text{Identity}$
- $s$  a  $kx$  times  $ky$  times  $r$  array; with  $r$  fixed, each matrix contains  $kxky$  values  $(u_h' * x_{kh} * v_k)^2$ , the partial (squared) singular values relative to  $x_{kh}$ .

**Author(s)**

Lafosse, R.

**References**

Lafosse R. & Ten Berge J. A simultaneous CONCOR method for the analysis of two partitioned matrices. submitted.

**Examples**

```
x <- matrix(runif(200),10,20)
s1 <- svdbip(x,c(3,4,3),c(5,5,10),2);sum(sum(sum(s1$s2)))
ss <- svdbips(x,c(3,4,3),c(5,5,10),2);sum(sum(sum(ss$s2)))
```

---

 svdcp

*SVD for a Column Partitioned matrix x*


---

**Description**

SVD for a Column Partitioned matrix  $x$ .  $r$  global successive solutions

**Usage**

```
svdcp(x, H, r)
```

**Arguments**

x	a p times q matrix
H	is a row vector which contains the numbers $q_h$ , $h=1,\dots,k_y$ , of the partition of x with $k_y$ column blocks : $\sum(q_h)=q$
r	The number of wanted successive solutions

**Details**

The first solution calculates  $1+k_x$  normed vectors: the vector  $u[,1]$  of  $R^p$  associated to the  $k_x$  vectors  $v_i[,1]$ 's of  $R^{q_i}$ . by maximizing  $\sum_i (u[,1]' * x_i * v_i[,1])^2$ , with  $1+k_x$  norm constraints. A value  $(u[,1]' * x_i * v_i[,1])^2$  measures the relative link between  $R^p$  and  $R^{q_i}$  associated to  $x_i$ . It corresponds to a partial squared singular value notion, since  $\sum_i (u[,1]' * x_i * v_i[,1])^2 = s^2$ , where  $s$  is the usual first singular value of  $x$ . The second solution is obtained from the same criterion, but after replacing each  $x_i$  by  $x_i - x_i * v_i[,1] * v_i[,1]'$ . And so on for the successive solutions  $1,2,\dots,r$ . The biggest number of solutions may be  $r = \inf(p, q_i)$ , when the  $x_i$ 's are supposed with full rank; then  $r_{\max} = \min([\min(H), p])$ .

**Value**

A list with following components:

u	a p times r matrix of $k_x$ row blocks $u_k$ ( $p_k \times r$ ); $u_k' * u_k = \text{Identity}$ .
v	a q times r matrix of $k_y$ row blocks $v_i$ ( $q_i \times r$ ) of axes in $R^{q_i}$ relative to $y_i$ ; $v_i' * v_i = \text{Identity}$
s	a $k_x$ times $k_y$ times r array; with r fixed, each matrix contains $k_x k_y$ values $(u_h' * x_{kh} * v_k)^2$ , the partial (squared) singular values relative to $x_{kh}$ .

**Author(s)**

Lafosse, R.

**References**

Lafosse R. & Hanafi M.(1997) Concordance d'un tableau avec K tableaux: Definition de K+1 uples synthetiques. Revue de Statistique Appliquee vol.45,n.4.

**Examples**

```
x <- matrix(runif(200),10,20)
s <- svdcp(x,c(5,5,10),1)
ss <- svd(x);ss$d[1]^2
sum(ss$s2)
```

# Index

concor, [2](#)  
concorcano, [3](#)  
concoreg, [4](#)  
concorgm, [5](#)  
concorgmcano, [7](#)  
concorgmreg, [8](#)  
concors, [9](#)  
concorscano, [10](#)  
concorsreg, [12](#)

svdbip, [13](#)  
svdbip2, [14](#)  
svdbips, [15](#)  
svdcp, [16](#)