Package 'BSSasymp'

September 12, 2017

Type Package

Title Asymptotic Covariance Matrices of Some BSS Mixing and Unmixing
Matrix Estimates

Version 1.2-1

Date 2017-09-11

Author Jari Miettinen, Klaus Nordhausen, Hannu Oja, Sara Taskinen

Maintainer Jari Miettinen <jari.p.miettinen@aalto.fi>

Imports fICA (>= 1.0-2), JADE

Description

Functions to compute the asymptotic covariance matrices of mixing and unmixing matrix esti-
mates of the following blind source separation (BSS) methods: symmetric and squared symmet-
ric FastICA, regular and adaptive deflation-based FastICA, FOBI, JADE, AMUSE and deflation-
based and symmetric SOBI. Also functions to estimate these covariances based on data are avail-
able. For details, see Miettinen et al. (2015) <doi:10.1214/15-STS520>, Mietti-
nen et al. (2016) <doi:10.1111/jtsa.12159>, Mietti-
nen et al. (2017) <doi:10.1016/j.sigpro.2016.08.028>, Mietti-
nen et al. (2017) <doi:10.18637/jss.v076.i02>, and references therein.

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2017-09-12 11:47:55 UTC

R topics documented:

BSSasymp-package	*Asymptotic Covariance Matrices of Some BSS Mixing and Unmixing Matrix Estimates.*

Description

Description: Functions to compute the asymptotic covariance matrices of mixing and unmixing matrix estimates of the following blind source separation (BSS) methods: symmetric and squared symmetric FastICA, regular and adaptive deflation-based FastICA, FOBI, JADE, AMUSE and deflation-based and symmetric SOBI. Also functions to estimate these covariances based on data are available. For details, see Miettinen et al. (2015) <doi:10.1214/15-STS520>, Miettinen et al. (2016) <doi:10.1111/jtsa.12159>, Miettinen et al. (2017) <doi:10.1016/j.sigpro.2016.08.028>, Miettinen et al. (2017) <doi:10.18637/jss.v076.i02>, and references therein.

Details

Package:	BSSasymp
Type:	Package
Version:	1.2-0
Date:	2017-09-11
License:	GPL (>= 2)

Author(s)

Jari Miettinen, Klaus Nordhausen, Hannu Oja, Sara Taskinen

Maintainer: Jari Miettinen <jari.p.miettinen@aalto.fi>

References

Miettinen, J., Nordhausen, K. and Taskinen, S. (2017), Blind Source Separation Based on Joint Diagonalization in R: The Packages JADE and BSSasymp, Journal of Statistical Software, **76**, 1-31, <doi:10.18637/jss.v076.i02>.

---

alphas   *Asymptotic variances of the deflation-based FastICA estimate*

Description

Deflation-based FastICA solves the blind source separation problem in the case of independent components. These function computes some interesting theoretic quantities related to the deflation-based FastICA unmixing matrix estimate, see details.

Usage

```
alphas(sdf, gs, dgs, name=NULL, supp=NULL,...)
```

Arguments

sdf	a list of density functions of the sources scaled so that the mean is 0 and variance is 1.
gs	a list of nonlinearity functions.
dgs	the first derivative functions of the nonlinearity functions.
name	a list of strings, which labels the nonlinearities.
supp	a two column matrix, where each row gives the lower and the upper limit used in numerical integration for the corresponding source component which is done using `integrate`. Can be NULL if each support is the whole real line.
...	arguments to be passed to `integrate`.

Details

When the mixing matrix is the identity matrix, the asymptotic variances of the first row elements of the deflation-based FastICA estimate depend only on the corresponding source component and the chosen nonlinearity function g. Furthermore, the asymptotic variances of the off-diagonal elements of the first row are equal, let us call this value alpha. Also the other asymptotic variances depend straightforwardly on alphas corresponding to different components and to (possibly) different nonlinearities. Alphas indicate which nonlinearities should be used and in which order the components should be separated. Reloaded (Nordhausen et al., 2011) and adaptive (Miettinen et al., 2014) deflation-based FastICA estimators are based on the estimation of alphas from the data.

Value

A matrix where the ith row gives the alphas for the ith nonlinearity and the jth column corresponds to the jth density in sdf.

Author(s)

Jari Miettinen

References

Miettinen, J., Nordhausen, K., Oja, H. and Taskinen, S. (2014), Deflation-based FastICA with adaptive choices of nonlinearities, IEEE Transactions on Signal Processing, *62(21)*, 5716–5724.

Nordhausen, K., Ilmonen, P., Mandal, A., Oja, H. and Ollila, E. (2011), Deflation-based Fas-tICA reloaded, in Proc. "19th European Signal Processing Conference 2011 (EUSIPCO 2011)", *Barcelona, 1854–1858.*

See Also

ASCOV_FastICAdefl, adapt_fICA

Examples

```
# source components have uniform- and exponential(1)- distribution
fu <- function(x){1/(sqrt(3)*2)}
fe <- function(x){exp(-x-1)}

supp <- matrix(c(-sqrt(3),sqrt(3),-1,Inf), nrow=2, ncol=2, byrow=TRUE)

# couple of nonlinearities
g_pow3 <- function(x){x^3}
dg_pow3 <- function(x){3*x^2}

g_gaus <- function(x){x*exp(-x^2/2)}
dg_gaus <- function(x){exp(-x^2/2)-x^2*exp(-x^2/2)}

gs <- c(g_pow3,g_gaus)
dgs <- c(dg_pow3,dg_gaus)

alphas(sdf=c(fu,fe), gs=c(g_pow3), dgs=c(dg_pow3), supp=supp)

alphas(sdf=c(fu,fe), gs=gs, dgs=dgs, supp=supp)
```

ASCOV_FastICAdefl	*Asymptotic covariance matrices of different deflation-based FastICA estimates*

Description

The regular deflation-based FastICA finds the independent components one by one using a non-linearity function. The adaptive deflation-based FastICA chooses, for each component separately, the best nonlinearity from a set of nonlinearities. This function computes asymptotic covariance matrices of the different deflation-based FastICA mixing and unmixing matrix estimates.

Usage

```
ASCOV_FastICAdefl(sdf, gs, dgs, Gs=NULL, method="adapt",
                  name=NULL, supp=NULL, A=NULL, ...)
```

Arguments

sdf	a list of density functions of the sources scaled so that the mean is 0 and variance is 1.
gs	a list of nonlinearity functions.
dgs	the first derivative functions of the nonlinearity functions.
Gs	the integral function of the nonlinearity function. Is needed only when method="G".
method	"adapt", "G" or "regular", see details.
name	a list of strings, which labels the nonlinearities.
supp	a two column matrix, where each row gives the lower and the upper limit used in numerical integration for the corresponding source component which is done using integrate. Can be NULL if each support is the whole real line.
A	the mixing matrix, identity matrix as default.
...	arguments to be passed to integrate.

Details

Depending on the argument method, the function computes the asymptotic covariance matrices for three different extraction orders of the independent components. The choice method="adapt" picks the adaptive deflation-based FastICA estimate, which extracts the components in asymptotically optimal order and uses the best nonlinearity from the set of nonlinearities gs. The other two methods use only one nonlinearity, and if gs and dgs contain more than one function, the first one is taken. When method="G", the order is based on the deviance from normality measured by Gs. When method="regular", the order is that of sdf.

The signs of the components are fixed so that the sum of the elements of each row of the unmixing matrix is positive.

Since the unmixing matrix has asymptotic normal distribution, we have a connection between the asymptotic variances and the minimum distance index, which is defined as

$$MD(\hat{W}, A) = \frac{1}{\sqrt{p-1}} \inf_{PD} ||PD\hat{W}A - I||,$$

where $\hat{W}$ is the unmixing matrix estimate, A is the mixing matrix, P is a permutation matrix and D a diagonal matrix with nonzero diagonal entries. If $\hat{W}A$ converges to the identity matrix, the limiting expected value of $n(p-1)MD^2$ is the sum of the asymptotic variances of the off-diagonal elements of $\hat{W}A$. Here n is the sample size and p is the number of components.

Value

A list with the following components:

W	mean of the unmixing matrix estimate.
COV_W	asymptotic covariance matrix of the unmixing matrix estimate.
A	mean of the mixing matrix estimate.
COV_A	asymptotic covariance matrix of the mixing matrix estimate.
EMD	the limiting expected value of $n(p-1)MD^2$, see details.
used_gs	indicates which nonlinearity is used in estimation of each rows of the unmixing matrix.

Author(s)

Jari Miettinen

References

Ilmonen, P., Nordhausen, K., Oja, H. and Ollila, E. (2010): A New Performance Index for ICA: Properties, Computation and Asymptotic Analysis. In Vigneron, V., Zarzoso, V., Moreau, E., Gribonval, R. and Vincent, E. (editors) Latent Variable Analysis and Signal Separation, *229–236, Springer.*

Miettinen, J., Nordhausen, K., Oja, H. and Taskinen, S. (2014), Deflation-based FastICA with adaptive choices of nonlinearities, IEEE Transactions on Signal Processing, *62(21), 5716–5724.*

Nordhausen, K., Ilmonen, P., Mandal, A., Oja, H. and Ollila, E. (2011), Deflation-based FastICA reloaded, in Proc. "19th European Signal Processing Conference 2011 (EUSIPCO 2011)", *Barcelona, 1854–1858.*

See Also

ASCOV_FastICAdefl_est, adapt_fICA, integrate

Examples

```
# source components have uniform- and exponential(1)- distribution
fu <- function(x){1/(sqrt(3)*2)}
fe <- function(x){exp(-x-1)}

supp <- matrix(c(-sqrt(3),sqrt(3),-1,Inf), nrow=2, ncol=2, byrow=TRUE)

# couple of nonlinearities
g_pow3 <- function(x){x^3}
dg_pow3 <- function(x){3*x^2}
G_pow3 <- function(x){x^4/4}

g_gaus <- function(x){x*exp(-x^2/2)}
dg_gaus <- function(x){exp(-x^2/2)-x^2*exp(-x^2/2)}

gs <- c(g_pow3,g_gaus)
dgs <- c(dg_pow3,dg_gaus)

A <- matrix(rnorm(4),2,2)

res1 <- ASCOV_FastICAdefl(sdf=c(fu,fe), gs=gs, dgs=dgs, supp=supp, A=A)

round(res1$COV_W, 2)
res1$EMD
res1$used_gs

res2 <- ASCOV_FastICAdefl(sdf=c(fu,fe), gs=c(g_pow3), dgs=c(dg_pow3),
Gs=c(G_pow3), method="G", supp=supp, A=A)

res2$EMD
```

ASCOV_FastICAdefl_est   *Asymptotic covariance matrices of deflation-based FastICA estimates*

Description

The regular deflation-based FastICA finds the independent components one by one using a nonlinearity function. The adaptive deflation-based FastICA chooses, for each component separately, the best nonlinearity from a set of nonlinearities. This function computes estimates of the covariance matrices of the different deflation-based FastICA mixing and unmixing matrix estimates.

Usage

```
ASCOV_FastICAdefl_est(X, gs, dgs, Gs=NULL, method="adapt",
                      name=NULL, mixed=TRUE)
```

Arguments

X	a numeric data matrix.
gs	a list of nonlinearity functions.
dgs	the first derivative functions of the nonlinearity functions.
Gs	the integral function of the nonlinearity function. Is needed only when `method="G"`.
method	"adapt" or "G", see details.
name	a list of strings, which labels the nonlinearities.
mixed	logical, see details.

Details

Depending on the argument `method`, the function computes the asymptotic covariance matrices for two different extraction orders of the independent components. The choice `method="adapt"` picks the adaptive deflation-based FastICA estimate, which extracts the components in asymptotically optimal order and uses the best nonlinearity from the set of nonlinearities `gs`. When `method="G"`, the order is based on the deviance from normality measured by `Gs`. This method uses only one nonlinearity, and if `gs` and `dgs` contain more than one function, the first one is taken.

If `mixed` is TRUE, then X will be transformed by the adaptive FastICA estimate. The option FALSE can be used, for example, to estimate the covariance when X are source estimates given by some other method than FastICA.

Value

A list with the following components:

W	estimated mean of the unmixing matrix estimate.
COV_W	estimated covariance matrix of the unmixing matrix estimate.
A	estimated mean of the mixing matrix estimate.
COV_A	estimated covariance matrix of the mixing matrix estimate.
used_gs	indicates which nonlinearity is used in estimation of each rows of the unmixing matrix.

Author(s)

Jari Miettinen

References

Miettinen, J., Nordhausen, K., Oja, H. and Taskinen, S. (2014), Deflation-based FastICA with adaptive choices of nonlinearities, IEEE Transactions on Signal Processing, *62(21), 5716–5724.*

Nordhausen, K., Ilmonen, P., Mandal, A., Oja, H. and Ollila, E. (2011), Deflation-based FastICA reloaded, in Proc. "19th European Signal Processing Conference 2011 (EUSIPCO 2011)", *Barcelona, 1854–1858.*

See Also

[ASCOV_FastICAdefl](#), [adapt_fICA](#)

Examples

```
# source components have uniform- and exponential(1)- distribution
s1 <- runif(1000,-sqrt(3),sqrt(3))
s2 <- rexp(1000)
S <- cbind(s1,s2)

# couple of nonlinearities
g_pow3 <- function(x){x^3}
dg_pow3 <- function(x){3*x^2}
G_pow3 <- function(x){x^4/4}

g_gaus <- function(x){x*exp(-x^2/2)}
dg_gaus <- function(x){exp(-x^2/2)-x^2*exp(-x^2/2)}

gs <- c(g_pow3,g_gaus)
dgs <- c(dg_pow3,dg_gaus)

A<-matrix(rnorm(4),2,2)

X <- S %*% t(A)

round(1000*ASCOV_FastICAdefl_est(X, gs=gs, dgs=dgs)$COV_W,2)

round(1000*ASCOV_FastICAdefl_est(X, gs=c(g_pow3), dgs=c(dg_pow3),
Gs=c(G_pow3), method="G")$COV_W,2)
```

ASCOV_FastICAsym   *Asymptotic covariance matrix of symmetric FastICA estimates*

Description

Symmetric FastICA estimators solves the blind source separation problem in the case of independent components. These functions compute the asymptotic covariance matrices of the regular and the squared symmetric FastICA mixing and unmixing matrix estimates.

Usage

```
ASCOV_FastICAsym(sdf, G, g, dg, supp=NULL, A=NULL, ...)

ASCOV_FastICAsym2(sdf, G, g, dg, supp=NULL, A=NULL, ...)
```

Arguments

sdf	a list of density functions of the sources scaled so that the mean is 0 and variance is 1.
G	the integral function of the nonlinearity function, see details.
g	the nonlinearity function.
dg	the first derivative function of the nonlinearity function.
supp	a two column matrix, where each row gives the lower and the upper limit used in numerical integration for the corresponding source component which is done using integrate. Can be NULL if each support is the whole real line.
A	the mixing matrix, identity matrix as default.
...	arguments to be passed to integrate.

Details

The signs of the components are fixed so that the sum of the elements of each row of the unmixing matrix is positive.

Since the unmixing matrix has asymptotic normal distribution, we have a connection between the asymptotic variances and the minimum distance index, which is defined as

$$MD(\hat{W}, A) = \frac{1}{\sqrt{p-1}} \inf_{PD} ||PD\hat{W}A - I||,$$

where $\hat{W}$ is the unmixing matrix estimate, A is the mixing matrix, P is a permutation matrix and D a diagonal matrix with nonzero diagonal entries. If $\hat{W}A$ converges to the identity matrix, the limiting expected value of $n(p-1)MD^2$ is the sum of the asymptotic variances of the off-diagonal elements of $\hat{W}A$. Here n is the sample size and p is the number of components.

Value

A list with the following components:

W	mean of the unmixing matrix estimate.
COV_W	asymptotic covariance matrix of the unmixing matrix estimate.
A	mean of the mixing matrix estimate.

COV_A   asymptotic covariance matrix of the mixing matrix estimate.

EMD   the limiting expected value of $n(p-1)MD^2$, see details.

Author(s)

Jari Miettinen

References

Hyv\"arinen, A. (1999), Fast and robust fixed-point algorithms for independent component analysis, IEEE Transactions on Neural Networks, *10, 626-634.*

Wei, T. (2014), The convergence and asymptotic analysis of the generalized symmetric FastICA algorithm, http://arxiv.org/abs/1408.0145.

Miettinen, J., Nordhausen, K., Oja, H., Taskinen, S. and Virta, J. (2015), The squared symmetric FastICA estimator, submitted.

See Also

ASCOV_FastICAsym_est, fICA, integrate

Examples

```
# source components have uniform- and exponential(1)- distribution
fu <- function(x){1/(sqrt(3)*2)}
fe <- function(x){exp(-x-1)}

supp <- matrix(c(-sqrt(3),sqrt(3),-1,Inf), nrow=2, ncol=2, byrow=TRUE)

# couple of nonlinearities
g_pow3 <- function(x){x^3}
dg_pow3 <- function(x){3*x^2}
G_pow3 <- function(x){x^4/4}

g_gaus <- function(x){x*exp(-x^2/2)}
dg_gaus <- function(x){exp(-x^2/2)-x^2*exp(-x^2/2)}
G_gaus <- function(x){-exp(-x^2/2)}

A <- matrix(rnorm(4),2,2)

res1 <- ASCOV_FastICAsym(sdf=c(fu,fe), G=G_pow3, g=g_pow3, dg=dg_pow3,
supp=supp, A=A)
res2 <- ASCOV_FastICAsym(sdf=c(fu,fe), G=G_gaus, g=g_gaus, dg=dg_gaus,
supp=supp, A=A)

round(res1$COV_W, 2)
res1$EMD

round(res2$COV_W, 2)
res2$EMD
```

ASCOV_FastICAsym_est   *Asymptotic covariance matrix of symmetric FastICA estimate*

Description

Symmetric FastICA solves the blind source separation problem in the case of independent compo-
nents. This function computes an estimate of the covariance matrix of symmetric FastICA mixing
and unmixing matrix estimates.

Usage

```
ASCOV_FastICAsym_est(X, G, g, dg, mixed=TRUE)

ASCOV_FastICAsym2_est(X, G, g, dg, mixed=TRUE)
```

Arguments

X	a numeric data matrix.
G	the integral function of the nonlinearity function.
g	the nonlinearity function.
dg	the first derivative function of the nonlinearity function.
mixed	logical, see details.

Details

If mixed is TRUE, then X will be transformed by the symmetric FastICA estimate. The option
FALSE can be used, for example, to estimate the covariance when X are source estimates given by
some other method than FastICA.

Value

A list with the following components:

W	estimated mean of the unmixing matrix estimate.
COV_W	estimated covariance matrix of the unmixing matrix estimate.
A	estimated mean of the mixing matrix estimate.
COV_A	estimated covariance matrix of the mixing matrix estimate.

Author(s)

Jari Miettinen

References

Hyv\"arinen, A. (1999), Fast and robust fixed-point algorithms for independent component analysis, IEEE Transactions on Neural Networks, *10, 626-634.*

Wei, T. (2014), The convergence and asymptotic analysis of the generalized symmetric FastICA algorithm, http://arxiv.org/abs/1408.0145.

See Also

[ASCOV_FastICAsym, fICA](#)

Examples

```
# source components have uniform- and exponential(1)- distribution
n <- 1000
s1 <- runif(n,-sqrt(3),sqrt(3))
s2 <- rexp(n)
S <- cbind(s1,s2)

# couple of nonlinearities
g_pow3 <- function(x){x^3}
dg_pow3 <- function(x){3*x^2}
G_pow3 <- function(x){x^4/4}

g_gaus <- function(x){x*exp(-x^2/2)}
dg_gaus <- function(x){exp(-x^2/2)-x^2*exp(-x^2/2)}
G_gaus <- function(x){-exp(-x^2/2)}

A<-matrix(rnorm(4),2,2)

X <- S %*% t(A)

round(n*ASCOV_FastICAsym_est(X,G=G_pow3,g=g_pow3,dg=dg_pow3)$COV_W,2)
round(n*ASCOV_FastICAsym_est(X,G=G_gaus,g=g_gaus,dg=dg_gaus)$COV_W,2)
```

ASCOV_JADE   *Asymptotic covariance matrix of JADE and FOBI estimates*

Description

JADE solves the blind source separation problem in the case of independent components with at most one component having kurtosis values zero, while FOBI requires distinct kurtosis values. The functions compute the asymptotic covariance matrices of JADE and FOBI estimates for the mixing or the unmixing matrices.

Usage

```
ASCOV_JADE(sdf, supp=NULL, A=NULL, ...)

ASCOV_FOBI(sdf, supp=NULL, A=NULL, ...)
```

Arguments

sdf	a vector of density functions of the sources scaled so that the mean is 0 and variance is 1.
supp	a two column matrix, where each row gives the lower and the upper limit used in numerical integration for the corresponding source component which is done using integrate.
A	the mixing matrix, identity matrix as default.
...	arguments to be passed to integrate.

Details

The order of the estimated components is fixed so that their kurtosis values are in a decreasing order. The signs of the components is fixed so that the sum of the elements of each row of the unmixing matrix is positive.

Since the unmixing matrix has asymptotic normal distribution, we have a connection between the asymptotic variances and the minimum distance index, which is defined as

$$MD(\hat{W}, A) = \frac{1}{\sqrt{p-1}} \inf_{PD} ||PD\hat{W}A - I||,$$

where $\hat{W}$ is the unmixing matrix estimate, A is the mixing matrix, P is a permutation matrix and D a diagonal matrix with nonzero diagonal entries. If $\hat{W}A$ converges to the identity matrix, the limiting expected value of $n(p-1)MD^2$ is the sum of the asymptotic variances of the off-diagonal elements of $\hat{W}A$. Here n is the sample size and p is the number of components.

Value

A list with the following components:

W	mean of the unmixing matrix estimate.
COV_W	asymptotic covariance matrix of the unmixing matrix estimate.
A	mean of the mixing matrix estimate.
COV_A	asymptotic covariance matrix of the mixing matrix estimate.
EMD	The limiting expected value of $n(p-1)MD^2$, see details.

Author(s)

Jari Miettinen

References

Ilmonen, P., Nevalainen, J. and Oja, H. (2010), Characteristics of multivariate distributions and the invariant coordinate system, Statistics and Probability Letters, *80, 1844–1853.*

Ilmonen, P., Nordhausen, K., Oja, H. and Ollila, E. (2010), A New Performance Index for ICA: Properties, Computation and Asymptotic Analysis. In Vigneron, V., Zarzoso, V., Moreau, E., Gribonval, R. and Vincent, E. (editors) Latent Variable Analysis and Signal Separation, *229–236, Springer.*

Miettinen, J., Taskinen S., Nordhausen, K. and Oja, H. (2015), Fourth Moments and Independent Component Analysis, Statistical Science, *30, 372–390.*

See Also

[ASCOV_JADE_est](#), [ASCOV_FOBI_est](#), [JADE](#), [FOBI](#)

Examples

```
# source components have uniform- and exponential(1)- distribution
fu <- function(x){1/(sqrt(3)*2)}
fe <- function(x){exp(-x-1)}

supp <- matrix(c(-sqrt(3),sqrt(3),-1,Inf), nrow=2, ncol=2, byrow=TRUE)
A<-matrix(rnorm(4),2,2)

jade <- ASCOV_JADE(sdf=c(fu,fe), supp=supp, A=A)
fobi <- ASCOV_FOBI(sdf=c(fu,fe), supp=supp, A=A)

round(jade$COV_W,2)
round(fobi$COV_W,2)

jade$EMD
fobi$EMD
```

ASCOV_JADE_est   *Asymptotic covariance matrix of JADE and FOBI estimates*

Description

JADE solves the blind source separation problem in the case of independent components with at most one component having kurtosis values zero, while FOBI requires distinct kurtosis values. The functions compute the asymptotic covariance matrices of JADE and FOBI estimates for the mixing or the unmixing matrix.

Usage

```
ASCOV_JADE_est(X, mixed=TRUE)

ASCOV_FOBI_est(X, mixed=TRUE)
```

Arguments

X   a numeric data matrixx.

mixed   logical, see details.

Details

If `mixed` is TRUE, then `X` will be transformed by the corresponding estimate. The option FALSE can be used, for example, to estimate the covariance when `X` are source estimates given by some other method than JADE or FOBI.

Value

A list with the following components:

W	estimated mean of the unmixing matrix estimate.
COV_W	estimated covariance matrix of the unmixing matrix estimate.
A	estimated mean of the mixing matrix estimate.
COV_A	estimated covariance matrix of the mixing matrix estimate.

Author(s)

Jari Miettinen

References

Ilmonen, P., Nevalainen, J. and Oja, H. (2010), Characteristics of multivariate distributions and the invariant coordinate system, Statistics and Probability Letters, *80, 1844–1853.*

Miettinen, J., Taskinen S., Nordhausen, K. and Oja, H. (2015), Fourth Moments and Independent Component Analysis, Statistical Science, *30, 372–390.*

See Also

ASCOV_JADE, ASCOV_FOBI, JADE, FOBI

Examples

```
# source components have t-10-, uniform- and gaussian distribution
s1 <- rt(1000,10)/sqrt(10/8)
s2 <- runif(1000,-sqrt(3),sqrt(3))
s3 <- rnorm(1000)
S <- cbind(s1,s2,s3)
A <- matrix(rnorm(9),3,3)

X <- S %*% t(A)
round(1000*ASCOV_JADE_est(X)$COV_W,2)
round(1000*ASCOV_FOBI_est(X)$COV_W,2)
```

ASCOV_SOBI	*Asymptotic covariance matrix of symmetric and deflation-based SOBI estimates*

Description

The symmetric and deflation-based SOBI methods solve the blind source separation problem in the case of second order stationary time series sources by jointly diagonalizing the covariance matrix and several autocovariance matrices at different lags. The functions compute the asymptotic covariance matrices of a SOBI estimates for the mixing or the unmixing matrices, when the sources are $MA(\infty)$ time series. Notice that, since AMUSE method is a special case of SOBI, also the asymptotic covariance matrix of an AMUSE estimate can be computed using these functions.

Usage

```
ASCOV_SOBI(psi, taus, a=2, Beta=NULL, A=NULL)

ASCOV_SOBIdefl(psi, taus, Beta=NULL, A=NULL)
```

Arguments

psi   a numeric matrix containing the MA-coefficients of the time series, see details.

taus   a vector of integers for the lags.

a   numeric, see details.

Beta   a matrix of fourth moments of the innovations, see details.

A   the mixing matrix, identity matrix as default.

Details

Naturally, the function can deal with only finite number of coefficients. The larger is the number of the rows of psi, the more accurate is the result, but also the longer is the computation time.

AR or ARMA coefficients can be transformed to MA coefficients by using [ARMAtoMA](#).

The ijth entry of Beta is $E[Z_{it}^2 Z_{jt}^2]$, where Z_{it} are the innovations of ith source component such that $E[Z_{it}] = 0$ and $E[Z_{it}^2] = 1$.

The order of the estimated components is fixed so that the sums of their squared autocovariances over taus are in a decreasing order. The signs of the components are fixed so that the sum of the elements of each row of the unmixing matrix is positive.

Since the unmixing matrix has asymptotic normal distribution, we have a connection between the asymptotic variances and the minimum distance index, which is defined as

$$MD(\hat{W}, A) = \frac{1}{\sqrt{p-1}} \inf_{PD} ||PD\hat{W}A - I||,$$

where $\hat{W}$ is the unmixing matrix estimate, A is the mixing matrix, P is a permutation matrix and D a diagonal matrix with nonzero diagonal entries. If $\hat{W}A$ converges to the identity matrix, the limiting expected value of $n(p-1)MD^2$ is the sum of the asymptotic variances of the off-diagonal elements of $\hat{W}A$. Here n is the sample size and p is the number of components.

The symmetric SOBI estimator maximizes the sum of squares of the diagonal elements of the autocovariance matrices. Different SOBI estimators are obtained when the diagonality of $p \times p$ matrices $B_1, \ldots, B_K$ is measured by

$$\sum_{k=1}^{K} \sum_{i=1}^{p} |(B_k)_{ii}|^a$$

with $a > 1$. The diagonality measure can be selected using the argument a.

Value

A list with the following components:

W	mean of the unmixing matrix estimate.
COV_W	asymptotic covariance matrix of the unmixing matrix estimate.
A	mean of the mixing matrix estimate.
COV_A	asymptotic covariance matrix of the mixing matrix estimate.
EMD	The limiting expected value of $n(p-1)MD^2$, see details.

Author(s)

Jari Miettinen

References

Ilmonen, P., Nordhausen, K., Oja, H. and Ollila, E. (2010): A New Performance Index for ICA: Properties, Computation and Asymptotic Analysis. In Vigneron, V., Zarzoso, V., Moreau, E., Gribonval, R. and Vincent, E. (editors) Latent Variable Analysis and Signal Separation, *229–236, Springer.*

Miettinen, J. (2015): Alternative diagonality criteria for SOBI. In Nordhausen, K. and Taskinen, S. (editors) Modern Nonparametric, Robust and Multivariate methods, Festschrift in Honour of Hannu Oja, *455–469, Springer.*

Miettinen, J., Nordhausen, K., Oja, H. and Taskinen, S. (2012), Statistical properties of a blind source separation estimator for stationary time series, Statistics and Probability Letters, *82, 1865–1873.*

Miettinen, J., Nordhausen, K., Oja, H. and Taskinen, S. (2014), Deflation-based separation of uncorrelated stationary time series, Journal of Multivariate Analysis, *123, 214–227.*

Miettinen, J., Illner, K., Nordhausen, K., Oja, H., Taskinen, S. and Theis, F. (2016), Separation of uncorrelated stationary time series using autocovariance matrices, Journal of Time Series Analysis, *37, 337–354.*

See Also

ASCOV_SOBI_est, SOBI, AMUSE, ARMAtoMA

Examples

```
A<- matrix(rnorm(9),3,3)
psi1 <- ARMAtoMA(ar=0.6, lag.max=100)
psi2 <- ARMAtoMA(ar=c(0.2,0.3,-0.3),lag.max=100)
psi3 <- ARMAtoMA(ar=-0.2, ma=c(0.5,-0.1,0.4), lag.max=100)
psi <- cbind(psi1,psi2,psi3)

sym <- ASCOV_SOBI(psi=psi, taus=1:10, A=A)
defl <- ASCOV_SOBIdefl(psi=psi, taus=1:10, A=A)
```

```
round(sym$COV_W,2)
round(defl$COV_W,2)

sym$EMD
defl$EMD
```

ASCOV_SOBI_est   *Asymptotic covariance matrix of symmetric and deflation-based SOBI estimates*

Description

The symmetric and deflation-based SOBI methods solve the blind source separation problem in the case of second order stationary time series sources by jointly diagonalizing the covariance matrix and several autocovariance matrices at different lags. The functions compute an estimate of the covariance matrix of a SOBI estimate for the mixing or the unmixing matrix, under the assumption that the sources are $MA(\infty)$ time series. Notice that, since AMUSE method is a special case of SOBI, also an estimate of the covariance matrix of an AMUSE estimate can be computed using these functions.

Usage

```
ASCOV_SOBI_estN(X, taus, mixed=TRUE, M=100, a=2)

ASCOV_SOBI_est(X, taus, arp=NULL, maq=NULL, mixed=TRUE,
               M=100, a=2, ...)

ASCOV_SOBIdefl_estN(X, taus, mixed=TRUE,  M=100)

ASCOV_SOBIdefl_est(X, taus, arp=NULL, maq=NULL, mixed=TRUE,
                   M=100, ...)
```

Arguments

X	a numeric data matrix or a multivariate time series object of class `ts`.
taus	a vector of integers for the lags.
arp	a vector containing the AR orders used for the estimation of ARMA coefficients.
maq	a vector containing the MA orders used for the estimation of ARMA coefficients.
mixed	logical, see details.
M	the number of autocovariance matrices used for the estimation of the covariance matrices, see details.
a	numeric, see details
...	arguments to be passed to `arima`.

Details

Functions ASCOV_SOBI_estN and ASCOV_SOBIdefl_estN assume that the innovations of the components are gaussian. Therefore, they are faster than ASCOV_SOBI_est and ASCOV_SOBI-defl_est, which estimate the fourth moments of the innovations by estimating the ARMA coefficients of the time series. Fitting the univariate ARMA coefficients is done using the function `arima` based on the orders provided by `arp` and `maq`.

The estimation is mostly based on autocovariance matrices and all non-zero matrices should be included. On the other hand, too large value of `M` increases the computation time and it may even reduce the estimation accuracy.

If `mixed` is TRUE, then `X` will be transformed by the corresponding SOBI estimate. The option FALSE can be used, for example, to estimate the covariance when `X` are source estimates given by some other method than SOBI.

The symmetric SOBI estimator maximizes the sum of squares of the diagonal elements of the autocovariance matrices. Different SOBI estimators are obtained when the diagonality of $p \times p$ matrices $B_1, \ldots, B_K$ is measured by

$$\sum_{k=1}^{K} \sum_{i=1}^{p} |(B_k)_{ii}|^a$$

with $a > 1$. The diagonality measure can be selected using the argument `a`.

Value

A list with the following components:

W	estimated mean of the unmixing matrix estimate.
COV_W	estimated covariance matrix of the unmixing matrix estimate.
A	estimated mean of the mixing matrix estimate.
COV_A	estimated covariance matrix of the mixing matrix estimate.

Author(s)

Jari Miettinen

References

Miettinen, J. (2015): *Alternative diagonality criteria for SOBI. In Nordhausen, K. and Taskinen, S. (editors)*, Modern Nonparametric, Robust and Multivariate methods, Festschrift in Honour of Hannu Oja, *455–469, Springer.*

Miettinen, J., Nordhausen, K., Oja, H. and Taskinen, S. (2012), *Statistical properties of a blind source separation estimator for stationary time series*, Statistics and Probability Letters, *82, 1865–1873.*

Miettinen, J., Nordhausen, K., Oja, H. and Taskinen, S. (2014), *Deflation-based separation of uncorrelated stationary time series*, Journal of Multivariate Analysis, *123, 214–227.*

Miettinen, J., Illner, K., Nordhausen, K., Oja, H., Taskinen, S. and Theis, F. (2015), *Separation of uncorrelated stationary time series using autocovariance matrices*, Journal of Time Series Analysis, *in print, DOI: 10.1111/jtsa.12159.*

See Also

ASCOV_SOBI, SOBI, AMUSE, arima

Examples

```
A<- matrix(rnorm(9),3,3)
s1 <- arima.sim(list(ar=0.6),1000)
s2 <- arima.sim(list(ma=c(0.2,0.3,-0.3)),1000)
s3 <- arima.sim(list(ar=-0.2,ma=c(0.5,-0.1,0.4)),1000)

S <- cbind(s1,s2,s3)
X <- S %*% t(A)

round(1000*ASCOV_SOBI_estN(X, taus=1:10)$COV_W,2)
round(1000*ASCOV_SOBIdefl_estN(X, taus=1:10)$COV_W,2)
```

---

aSOBI   *Alternative SOBI estimators*

Description

The SOBI method solves the blind source separation problem in the case of second order station-
ary time series sources by jointly diagonalizing the covariance matrix and several autocovariance
matrices. In the classical SOBI method, the sum of squares of the diagonal elements is used as
diagonality measure. This function computes the SOBI estimate, when a choice from a family of
alternative diagonality criteria is used.

Usage

```
aSOBI(X, k=12, a=4, eps=1e-06, maxiter=1000)
```

Arguments

X	a numeric data matrix or a multivariate time series object of class `ts`.
k	if a single integer, then the lags 1:k are used, if an integer vector, then these are used as the lags.
a	numeric, determines the diagonality criterion, see details.
eps	convergence tolerance.
maxiter	maximum number of iterations.

Details

The classical SOBI estimator maximizes the sum of squares of the diagonal elements of the auto-covariance matrices. A family of alternative SOBI estimators is obtained when the diagonality of $p \times p$ matrices $B_1, \ldots, B_K$ is measured by

$$\sum_{k=1}^{K} \sum_{i=1}^{p} |(B_k)_{ii}|^a$$

with $a > 1$.

Value

A list with class 'bss' containing the following components:

W	estimated unmixing matrix.
S	estimated sources standardized to have mean 0 and unit variances.
k	lags used.
a	value of the diagonality criterion parameter used.

Author(s)

Jari Miettinen

References

Miettinen, J. (2015): Alternative diagonality criteria for SOBI. In Nordhausen, K. and Taskinen, S. (editors), Modern Nonparametric, Robust and Multivariate methods, Festschrift in Honour of Hannu Oja, *455–469, Springer.*

See Also

[SOBI]

Examples

```
A <- matrix(rnorm(9),3,3)
s1 <- arima.sim(list(ar=0.6),1000)
s2 <- arima.sim(list(ma=c(0.2,0.3,-0.3)),1000)
s3 <- arima.sim(list(ar=-0.2,ma=c(0.5,-0.1,0.4)),1000)

S <- cbind(s1,s2,s3)
X <- S %*% t(A)

aSOBI(X, a=3)
```

CRB	*Cramer-Rao bound for the unmixing matrix estimate in the independent component model.*

Description

Cramer-Rao bound for the unmixing matrix estimate in the independent component model.

Usage

```
CRB(sdf,supp=NULL,A=NULL,eps=1e-04,...)
```

Arguments

sdf	a list of density functions of the sources scaled so that the mean is 0 and variance is 1.
supp	a two column matrix, where each row gives the lower and the upper limit used in numerical integration for the corresponding source component which is done using [integrate](). Can be NULL if each support is the whole real line.
A	the mixing matrix, identity by default.
eps	a value which is used when the derivative functions of the density functions are approximated.
...	arguments to be passed to [integrate]().

Details

Let $\hat{W}$ denote an unmixing matrix estimate. If the estimate is affine equivariant, then the matrix $\hat{G} = \hat{W}A$ does not depend on the mixing matrix A and the estimated independent components are $\hat{S} = \hat{G}S$, where S is the matrix of the true independent components.

Value

A list containing the following components:

CRLB	A matrix whose elements give the Cramer-Rao lower bounds for the asymptotic variances of the corresponding elements of $\hat{G}$.
FIM	The Fisher information matrix.
EMD	The sum of the Cramer-Rao lower bounds of the off-diagonal elements of $\hat{G}$.

Author(s)

Jari Miettinen

References

Ollila, E., Kim, H. J. and Koivunen, V. (2008), Compact Cramer-Rao bound expression for independent component analysis. IEEE Transactions on Signal Processing, *56(4), 1421–1428.*

Examples

```
# source components have t(9)- and Gaussian distribution

f1<-function(x)
{
 gamma(5)*(1+(x*sqrt(9/7))^2/9)^(-5)/
 (sqrt(9*pi/(9/7))*gamma(9/2))
}

f2<-function(x)
{
 exp(-(x)^2/2)/sqrt(2*pi)
}

CRB(sdf=c(f1,f2))
```

---

eSOBI   *The efficient SOBI estimator*

Description

The SOBI method solves the blind source separation problem in the case of second order stationary time series sources by jointly diagonalizing the covariance matrix and several autocovariance matrices. The separation performance depends on the lag set. The efficient SOBI estimator uses asymptotic results to estimate the variances of the elements of the SOBI unmixing matrices obtained by given lag sets. The unmixing matrix corresponding to the lag set which minimizes the sum of the variances is the efficient SOBI estimate.

Usage

```
eSOBI(X, taus=taus_def, M=200, fast=TRUE, eps=1e-06, maxiter=1000)
```

Arguments

X	a numeric data matrix or a multivariate time series object of class `ts`.
taus	a list whose components are vectors of integers. The list gives the set of lag sets. The default set is `taus_def`.
M	the number of autocovariance matrices used for the estimation of the variance estimates, see `ASCOV_SOBI_estN`.
fast	logical, see details.
eps	convergence tolerance.
maxiter	maximum number of iterations.

Details

The function includes two versions of the efficient SOBI algorithm. The faster version uses only one SOBI estimate given by the first component in taus to estimate the sum of the limiting variances for all lag sets. In the other version, which is obtained by fast=FALSE, the sum of the limiting variances of each lag set is estimated using that particular lag set for the initial SOBI estimate. When the length of the time series is sufficient, say 5000 or more, the two versions yield equally good estimates and the use of the faster version is recommended. Otherwise we recommend the use of the slower version. The variance estimates are based on asymptotic results which assume that the time series are multivariate linear processes. Such processes include a wide class of stationary time series, for example all causal ARMA processes. It is also assumed that the innovations are Gaussian. This simplifies the computations and has practically no effect on which lag set is chosen. If the user does not want to make the Gaussianity assumption, the slower version of the function can be easily modified by replacing the function ASCOV_SOBI_estN in the code by ASCOV_SOBI_est. If the SOBI algorithm fails to converge for some lag set, the corresponding value of the estimated sum of variances in sum_var is Inf.

Value

A list with class 'bss' containing the following components:

W	estimated unmixing matrix.
S	estimated sources standardized to have mean 0 and unit variances.
taus_used	the lag set which is considered best and used for the estimation of the unmixing matrix.
sum_var	estimated sum of variances of the unmixing matrix estimates for all lag sets given in taus.

Author(s)

Jari Miettinen

References

Miettinen, J., Illner, K., Nordhausen, K., Oja, H., Taskinen, S. and Theis, F. (2015), Separation of uncorrelated stationary time series using autocovariance matrices, Journal of Time Series Analysis, *in print, http://arxiv.org/abs/1405.3388.*

Taskinen, S., Miettinen, J. and Nordhausen, K. (2016), Efficient second order blind identification method for separation of uncorrelated stationary time series, submitted.

See Also

ASCOV_SOBI_est, SOBI

Examples

```
A <- matrix(rnorm(9),3,3)
s1 <- arima.sim(list(ar=0.6),1000)
s2 <- arima.sim(list(ma=c(0.2,0.3,-0.3)),1000)
```

```
s3 <- arima.sim(list(ar=-0.2,ma=c(0.5,-0.1,0.4)),1000)

S <- cbind(s1,s2,s3)
X <- S %*% t(A)

taus <- list(1,1:2,1:5,1:12)
eSOBI(X, taus=taus)
```

---

eSOBI_lags   *The default set of lag sets for the efficient SOBI estimator*

Description

The set of lag sets to be used as the default set in function [eSOBI](#) .

Usage

```
taus_def
```

Details

The object `taus_def` contains the following lag sets:

taus_def[[1]]	1,2,...,12
taus_def[[2]]	1
taus_def[[3]]	1,2
taus_def[[4]]	1,2,3
taus_def[[5]]	1,2,3,4,5
taus_def[[6]]	1,2,...,8
taus_def[[7]]	1,2,...,20
taus_def[[8]]	1,2,...,50
taus_def[[9]]	1,2,...,10,12,...,20
taus_def[[10]]	5,6,...,10,12,...,20,25,...,50
taus_def[[11]]	2,4,...,20
taus_def[[12]]	1,2,3,5,7,11,13,17,19

Author(s)

Jari Miettinen

References

Taskinen, S., Miettinen, J. and Nordhausen, K. (2016), Efficient second order blind identification method for separation of uncorrelated stationary time series, submitted.

See Also

[eSOBI](#)

Index