Package 'ERP'

December 13, 2019

Type Package

Title Significance Analysis of Event-Related Potentials Data

Version 2.2

Date 2019-12-10

Author David Causeur, Ching-Fan Sheu, Mei-Chen Chu, Flavia Rufini

Maintainer David Causeur <david.causeur@agrocampus-ouest.fr>

Depends R (>= 3.0.0)

Imports graphics,grDevices,stats,corpcor,irlba,fdrtool,splines,mnormt,pacman

Description Functions for signal detection and identification designed for Event-Related Potentials (ERP) data in a linear model framework. The functional F-test proposed in Causeur, Sheu, Perthame, Rufini (2018, submitted) for analysis of variance issues in ERP designs is implemented for signal detection (tests for mean difference among groups of curves in One-way ANOVA designs for example). Once an experimental effect is declared significant, identification of significant intervals is achieved by the multiple testing procedures reviewed and compared in Sheu, Perthame, Lee and Causeur (2016, <DOI:10.1214/15-AOAS888>). Some of the methods gathered in the package are the classical FDR- and FWER-controlling procedures, also available using function p.adjust. The package also implements the Guthrie-Buchwald procedure (Guthrie and Buchwald, 1991 <DOI:10.1111/j.1469-8986.1991.tb00417.x>), which accounts for the auto-correlation among t-tests to control erroneous detection of short intervals. The Adaptive Factor-Adjustment method is an extension of the method described in Causeur, Chu, Hsieh and Sheu (2012, <DOI:10.3758/s13428-012-0230-0>). It assumes a factor model for the correlation among tests and combines adaptively the estimation of the signal and the updating of the dependence modelling (see Sheu et al., 2016, <DOI:10.1214/15-AOAS888> for further details).

License GPL (>= 2)

URL http://erpinr.org

Suggests prettydoc,knitr,rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2019-12-13 11:30:06 UTC

R topics documented:

ERP-package | *Significance Analysis of Event-Related Potentials Data: Significance Analysis of Event-Related Potentials Data*

Description

Functions for signal detection and identification designed for Event-Related Potentials (ERP) data in a linear model framework. The functional F-test proposed in Causeur, Sheu, Perthame, Rufini (2018, submitted) for analysis of variance issues in ERP designs is implemented for signal detection (tests for mean difference among groups of curves in One-way ANOVA designs for example). Once an experimental effect is declared significant, identification of significant intervals is achieved by the multiple testing procedures reviewed and compared in Sheu, Perthame, Lee and Causeur (2016, <DOI:10.1214/15-AOAS888>). Some of the methods gathered in the package are the classical FDR- and FWER-controlling procedures, also available using function p.adjust. The package also implements the Guthrie-Buchwald procedure (Guthrie and Buchwald, 1991 <DOI:10.1111/j.1469-8986.1991.tb00417.x>), which accounts for the auto-correlation among t-tests to control erroneous detection of short intervals. The Adaptive Factor-Adjustment method is an extension of the method described in Causeur, Chu, Hsieh and Sheu (2012, <DOI:10.3758/s13428-012-0230-0>). It assumes a factor model for the correlation among tests and combines adaptively the estimation of the signal and the updating of the dependence modelling (see Sheu et al., 2016, <DOI:10.1214/15-AOAS888> for further details).: This package provides testing procedures designed for Event-Related Potentials (ERP) data in a linear model framework. The functional F-test proposed in Causeur, Sheu, Perthame, Rufini (2018) for analysis of variance issues in ERP designs is implemented for signal detection (tests for mean difference among groups of curves in One-way ANOVA designs for example). Once an experimental effect is declared significant, identification of significant intervals is achieved by the multiple testing procedures reviewed and compared in Sheu, Perthame, Lee and Causeur (2016, <DOI: 10.1214/15-AOAS888>). Some of the methods gathered in the package are the classical FDR- and FWER-controlling procedures, also available using function p.adjust. The package also implements the Guthrie-Buchwald procedure (Guthrie and Buchwald, 1991, <DOI: 10.1111/j.1469-8986.1991.tb00417.x>), which accounts for the auto-correlation among t-tests to control erroneous detection of short intervals. The Adaptive Factor-Adjustment

method is an extension of the method described in Causeur, Chu, Hsieh and Sheu (2012, <DOI: 10.3758/s13428-012-0230-0>). It assumes a factor model for the correlation among tests and combines adaptively the estimation of the signal and the updating of the dependence modelling (see Sheu et al., 2016, <DOI: 10.1214/15-AOAS888> for further details).

Details

The DESCRIPTION file:

Package:	ERP
Type:	Package
Title:	Significance Analysis of Event-Related Potentials Data
Version:	2.2
Date:	2019-12-10
Author:	David Causeur, Ching-Fan Sheu, Mei-Chen Chu, Flavia Rufini
Maintainer:	David Causeur <david.causeur@agrocampus-ouest.fr>
Depends:	R (>= 3.0.0)
Imports:	graphics,grDevices,stats,corpcor,irlba,fdrtool,splines,mnormt,pacman
Description:	Functions for signal detection and identification designed for Event-Related Potentials (ERP) data in a line
License:	GPL (>=2)
URL:	http://erpinr.org
Suggests:	prettydoc,knitr,rmarkdown
VignetteBuilder:	knitr

Index of help topics:

ERP-package	Significance Analysis of Event-Related Potentials Data: Significance Analysis of Event-Related Potentials Data
emfa	Expectation-Maximization (EM) estimation of a factor model.
erpFtest	Functional Analysis-of-Variance (Anova) testing of Event-Related Potentials (ERP) data
erpavetest	Significance testing of averaged ERPs.
erpfatest	Adaptive Factor-Adjustement for multiple testing of ERP data
erpplot	Plot of ERP curves or effect curves (difference curve for example) with confidence intervals.
erptest	FDR- and FWER-controlling Multiple testing of ERP data
gbtest	The Guthrie-Buchwald procedure for significance analysis of ERP data
ifa	Inverse of a matrix based on its factor decomposition.
impulsivity	Event-Related Potentials data from a study conducted by Shen et al. (2014) to investigate neural correlates of impulsive behavior.
isqrtfa	Inverse square-root of a matrix based on its

```
                              factor decomposition.
nbfactors                     Determination of the number of factors in high
                              dimensional factor models.
```

This package is designed to help making decision about the position on the brain and the time interval after the onset where an effect is significant. The function erpplot can be used to display the ERP curves in an exploratory purpose or to plot effect curves (difference curve among experimental groups for example), with or without smoothing options and confidendence intervals. All the other functions implement testing procedures of ERP data in a linear model framework (F-tests for the comparison of two nested models). The function gbtest implements the Guthrie-Buchwald procedure (see Guthrie and Buchwald, 1991). The function erpavetest first partitions the entire interval of ERP observations into a predetermined number of equally-spaced intervals before performing siginificance testing using the averaged ERPs. The function erptest can be used for the classical FDR- and FWER-controlling multiple testing of ERP data: especially the Benjamini-Hochberg (see Benjamini and Hochberg, 1995) and Benjamini-Yekutieli (see Benjamini and Yekutieli, 2001) procedures, with the possible extension proposed by Storey and Tibshirani (2003) including a non-parametric estimation of the proportion of true nulls. Finally, erpFtest is designed for functional F-tests in ERP designs using the generalized likelihood-ratio test proposed by Causeur et al. (2018) and erpfatest for the identification of significant intervals with FDR control using the adaptive factor-adjustment (AFA) multiple testing procedure proposed by Sheu et al. (2016).

Author(s)

David Causeur, Ching-Fan Sheu, Mei-Chen Chu, Flavia Rufini David Causeur, (Agrocampus, Rennes, France), Ching-Fan Sheu (National Cheng Kung University, Tainan, Taiwan), Mei-Chen Chu (National Cheng Kung University, Tainan, Taiwan), Flavia Rufini (Agrocampus, Rennes, France) Maintainer: David Causeur <david.causeur@agrocampus-ouest.fr> Maintainer: David Causeur, http://math.agrocampus-ouest.fr/infoglueDeliverLive/membres/david.causeur, mailto: david.causeur@agrocampus-ouest.fr

References

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society Series B, 289-300.

Benjamini, Y., and Yekutieli, D. (2001). The control of the false discovery rate in multiple testing under dependency. Annals of Statistics 29, 1165-1188.

Causeur, D., Chu, M.-C., Hsieh, S. and Sheu, C.-F. (2012). A factor-adjusted multiple testing procedure for ERP data analysis. Behavior Research Methods, 44, 635-643.

Causeur, D., Sheu, C.-F., Perthame, E. and Rufini, F. (2018). A functional generalized F-test for signal detection with applications to event-related potentials significance analysis. Submitted.

Friguet, C., Kloareg, M. and Causeur, D. (2009). A factor model approach to multiple testing under dependence. Journal of the American Statistical Association, 104, 1406-1415.

Guthrie, D. and Buchwald, J.S. (1991). Significance testing of difference potentials. Psychophysiology, 28, 240-244.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. Scandinavian Journal of Statistics, 6, 65-70.

Hommel, G. (1988). A stagewise rejective multiple test procedure based on a modified Bonferroni test. Biometrika, 75, 383-386.

Hochberg, Y. (1988). A sharper Bonferroni procedure for multiple tests of significance. Biometrika, 75, 800-803.

Shaffer, J. P. (1995). Multiple hypothesis testing. Annual Review of Psychology, 46, 561-576.

Sarkar, S. (1998). Some probability inequalities for ordered MTP2 random variables: a proof of Simes conjecture. Annals of Statistics, 26, 494-504.

Sarkar, S., and Chang, C. K. (1997). Simes' method for multiple hypothesis testing with positively dependent test statistics. Journal of the American Statistical Association, 92, 1601-1608.

Shen, I. H., Lee, D. S. and Chen, C. L. (2014). The role of trait impulsivity in response inhibition: event-related potentials in a stop-signal task. International Journal of Psychophysiology 91.

Sheu, C-.F., Perthame, E., Lee, Y-.S., Causeur, D. (2016). Accounting for time dependence in large-scale multiple testing of event-related potential data. Annals of Applied Statistics. 10(1), 219-245.

Wright, S. P. (1992). Adjusted P-values for simultaneous inference. Biometrics, 48, 1005-1013.

Examples

```
## Not run:

data(impulsivity)

# Paired t-tests for the comparison of the ERP curves in the two conditions,
# within experimental group High, at channel CPZ

erpdta.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),5:505]
   # ERP curves for subjects in group 'High'
covariates.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),1:4]
covariates.highCPZ = droplevels(covariates.highCPZ)
   # Experimental covariates for subjects in group High

design = model.matrix(~C(Subject,sum)+Condition,data=covariates.highCPZ)
   # Design matrix to compare ERP curves in the two conditions
design0 = model.matrix(~C(Subject,sum),data=covariates.highCPZ)
   # Design matrix for the null model (no condition effect)

tests = erpfatest(erpdta.highCPZ,design,design0,nbf=NULL,wantplot=TRUE,significance="none")
   # with significance="none", just to choose a number of factors
Ftest = erpFtest(erpdta.highCPZ,design,design0,nbf=6)
Ftest$pval
tests = erpfatest(erpdta.highCPZ,design,design0,nbf=6)
  # with nbf=6 (approximate conservative recommendation based on the variance inflation plot)

time_pt = seq(0,1000,2)    # Sequence of time points (1 time point every 2ms in [0,1000])
nbs = 20                    # Number of B-splines for the plot of the effect curve
effect=which(colnames(design)=="ConditionSuccess")
erpplot(erpdta.highCPZ,design=design,frames=time_pt,effect=effect,xlab="Time (ms)",
        ylab=expression(Effect~curve~(mu~V)),bty="l",ylim=c(-3,3),nbs=nbs,
```

```
        cex.axis=1.25,cex.lab=1.25,interval="simultaneous")
  # with interval="simultaneous", both the pointwise and the simultaneous confidence bands
   # are plotted
points(time_pt[tests$significant],rep(0,length(tests$significant)),pch=16,col="blue")
   # Identifies significant time points by blue dots
title("Success-Failure effect curve with 95 percent C.I.",cex.main=1.25)
mtext(paste("12 subjects - Group High - ",nbs," B-splines",sep=""),cex=1.25)

## End(Not run)
```

---

emfa   *Expectation-Maximization (EM) estimation of a factor model.*

Description

This function implements the EM algorithm by Thayer and Rubin (1982) for the ML estimation of a factor model. It is an internal function used to estimate the factor model parameters in the factor-adjustment methods.

Usage

```
emfa(dta, nbf, min.err = 1e-06, verbose = FALSE, svd.method = c("fast.svd", "irlba"))
```

Arguments

dta	n x m matrix which rows are multivariate m-profiles for n individuals. n can be much smaller than m.
nbf	number of factors. It has to be a positive integer, smaller than the rank of dta.
min.err	stopping criterion for the iterative algorithm. Maximum difference between the estimated parameters in the last two iterations.
verbose	logical value. If verbose=TRUE, then some information is printed along the calculation.
svd.method	the EM algorithm starts from an SVD estimation of the factor model parameters. The default option to implement this SVD is fast.svd. An alternative option is an approximate but faster SVD by function irlba.

Details

Data are centered but not scaled. If a factor model for a correlation matrix is to be estimated, then a scaled dataset is required as an input of the function.

Value

B	m x nbf matrix of loadings
Psi	m-vector of specific variances
Factors	n x nbf matrix of factor scores
Objective	Final value of the stopping criterion, after convergence.

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

References

Friguet, C., Kloareg, M. and Causeur, D. (2009). A factor model approach to multiple testing under dependence. Journal of the American Statistical Association. 104 (488), 1406-1415.

Rubin, D. B., and Thayer, D. T. (1982), EM Algorithms for ML Factor Analysis, Psychometrika, 47 (1), 69-76.

See Also

factanal

Examples

```
data(impulsivity)

erpdta = as.matrix(impulsivity[,5:505]) # erpdta contains the whole set of ERP curves

fa = emfa(erpdta,nbf=1) # 1-factor modelling of the ERP curves
fa$Objective            # Final difference between the last two iterations
Semp = var(erpdta)      # Sample estimation of the variance of ERP curves
Sfa = diag(fa$Psi)+tcrossprod(fa$B) # Factorial estimation of the variance
max(abs(Semp-Sfa))      # Distance between the two estimates

fa = emfa(erpdta,nbf=20) # 20-factor modelling of the ERP curves in erpdta
fa$Objective             # Final difference between the last two iterations
Semp = var(erpdta)       # Sample estimation of the variance of ERP curves
Sfa = diag(fa$Psi)+tcrossprod(fa$B) # Factorial estimation of the variance
max(abs(Semp-Sfa))       # Distance between the two estimates
```

| erpavetest | *Significance testing of averaged ERPs.* |

Description

The function first calculates averaged ERP values within a predetermined number of equally-spaced intervals then tests for significance of the relationship between averaged ERPs and covariates in a linear model framework.

Usage

```
erpavetest(dta, design, design0 = NULL, nintervals = 10,
method = c("none","BH","holm","hochberg","hommel","bonferroni","BY","fdr"),alpha = 0.05)
```

Arguments

`dta`	Data frame containing the ERP curves: each column corresponds to a time frame and each row to a curve.
`design`	Design matrix of the nonnull model for the relationship between the ERP and the experimental variables. Typically the output of the function model.matrix
`design0`	Design matrix of the null model. Typically a submodel of the nonnull model, obtained by removing columns from design. Default is NULL, corresponding to the model with no covariates.
`nintervals`	Number of intervals in the partition of the whole interval of observation. Default is 10.
`method`	FDR- or FWER- controlling multiple testing procedures as available in the function p.adjust. Default is "none", for no multiplicity correction.
`alpha`	The FDR or FWER control level. Default is 0.05

Value

`pval`	p-values of the tests.
`correctedpval`	Corrected p-values, for the multiplicity of tests. Depends on the multiple testing method (see function p.adjust).
`significant`	Indices of the time points for which the test is positive.
`segments`	Factor giving the membership of timepoints to each interval in the partition.
`breaks`	Breakpoints of the partition.
`test`	Pointwise F-statistics if p>1, where p is the difference between the numbers of parameters in the nonnull and null models. Otherwise, if p=1, the function returns pointwise t-statistics (signed square-roots of F-statistics).
`df1`	Residual degrees of freedom for the nonnull model.
`df0`	Residual degrees of freedom for the null model.
`signal`	Estimated signal: a pxT matrix, where T the number of frames.
`r2`	R-squared values for each of the T linear models.

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

See Also

[erptest](), [erpfatest](), [gbtest](), [p.adjust]()

Examples

```
data(impulsivity)

# Paired t-tests for the comparison of the ERP curves in the two conditions,
# within experimental group High, at channel CPZ
```

```
erpdta.high = impulsivity[impulsivity$Group=="High",5:505]
   # ERP curves for subjects in group 'High'
covariates.high = impulsivity[impulsivity$Group=="High",1:4]
   # Experimental covariates for subjects in group 'High'

design = model.matrix(~C(Subject,sum)+Condition,data=covariates.high)
   # Design matrix to compare ERP curves in the two conditions
design0 = model.matrix(~C(Subject,sum),data=covariates.high)
   # Design matrix for the null model (no condition effect)

tests = erpavetest(erpdta.high,design,design0)

time_pt = seq(0,1000,2)     # sequence of time points (1 time point every 2ms in [0,1000])
nbs = 20                     # Number of B-splines for the plot of the effect curve
effect=which(colnames(design)=="ConditionSuccess")
erpplot(erpdta.high,design=design,frames=time_pt,effect=effect,xlab="Time (ms)",
        ylab=expression(Effect~curve~(mu~V)),bty="l",ylim=c(-3,3),nbs=nbs,
        cex.axis=1.25,cex.lab=1.25,interval="simultaneous")
  # with interval="simultaneous", both the pointwise and the simultaneous confidence bands
   # are plotted
abline(v=time_pt[tests$breaks],lty=2,col="darkgray")
   # Add a grid to show breakpoints
points(time_pt[tests$significant],rep(0,length(tests$significant)),pch=16,col="blue")
   # Identifies significant time points by blue dots
title("Success-Failure effect curve with 95 percent C.I.",cex.main=1.25)
mtext(paste("12 subjects - Group 'High' - ",nbs," B-splines",sep=""),cex=1.25)
```

---

erpfatest   *Adaptive Factor-Adjustement for multiple testing of ERP data*

Description

Adaptive factor-adjusted FDR- and FWER-controlling multiple testing procedures for ERP data.
The procedure is described in detail in Sheu, Perthame, Lee, and Causeur (2016).

Usage

```
erpfatest(dta, design, design0 = NULL,
method = c("BH","holm","hochberg","hommel","bonferroni","BY","fdr","none"),nbf = NULL,
nsamples = 200, significance = c("Satterthwaite", "none"),
nbfmax = min(c(nsamples, nrow(design))) - ncol(design) - 1, alpha = 0.05, pi0 = 1,
wantplot = ifelse(is.null(nbf), TRUE, FALSE), s0 = NULL, min.err = 0.01, maxiter = 5,
verbose = FALSE, svd.method = c("fast.svd", "irlba"))
```

Arguments

dta	Data frame containing the ERP measurements: each column corresponds to a time frame and each row to a curve.
design	Design matrix of the nonnull model for the relationship between the ERP and the experimental variables. Typically the output of the function model.matrix
design0	Design matrix of the null model. Typically a submodel of the nonnull model, obtained by removing columns from design. Default is NULL, corresponding to the model with no covariates.
method	FDR- or FWER- controlling multiple testing procedures as available in the function p.adjust. Default is "BH", for the Benjamini-Hochberg procedure (see Benjamini and Hochberg, 1995).
nbf	Number of factors in the residual covariance model. Default is NULL: the number of factors is determined by minimization of the variance inflation criterion as suggested in Friguet et al. (2009).
nsamples	Number of Monte-Carlo samples for the estimation of the null distribution. Default is nsamples=200, recommended when the Satterthwaite approximation is used.
significance	If "Satterthwaite", then the Monte-Carlo p-value is calculated with a Satterthwaite approximation of the null distribution. The other possible value is "none", for no calculation of the p-value.
nbfmax	Only required if nbf=NULL. The largest possible number of factors.
alpha	The FDR or FWER control level. Default is 0.05
pi0	An estimate of the proportion of true null hypotheses, which can be plugged into an FDR controlling multiple testing procedure to improve its efficiency. Default is 1, corresponding to the classical FDR controlling procedures. If NULL, the proportion is estimated using the function pval.estimate.eta0 of package fdrtool, with the method proposed by Storey and Tibshirani (2003).
wantplot	If TRUE, a diagnostic plot is produced to help choosing the number of factors. Only active if nbf=NULL.
s0	Prior knowledge of the time frames for which no signal is expected. For example, s0=c(1:50, 226:251) specifies that the first 50 time frames and time frames between 226 and 251 are known not to contain ERP signal. s0 can also be specified by giving the lower and upper fraction of the entire time interval in which the signal is to be searched for. For example: s0=c(0.2, 0.9) means that ERP signals are not expected for for the first 20 percent and last 10 percent of the time frames measured. Defaul is NULL and it initiates a data-driven determination of s0.
min.err	Control parameter for convergence of the iterative algorithm. Default is 1e-03.
maxiter	Maximum number of iterations in algorithms. Default is 5.
verbose	If TRUE, details are printed along the iterations of the algorithm. Default is FALSE.
svd.method	the EM algorithm starts from an SVD estimation of the factor model parameters. The default option to implement this SVD is [fast.svd](). An alternative option is an approximate but faster SVD by function [irlba]().

Details

The method is described in Sheu et al. (2016). It combines a decorrelation step based on a regression factor model as in Leek and Storey (2008), Friguet et al. (2009) or Sun et al. (2012) with an adaptive estimation of the ERP signal. The multiple testing corrections of the p-values are described in the help documentation of the function `p.adjust`.

Value

`pval`	p-values of the Adaptive Factor-Adjusted tests.
`correctedpval`	Corrected p-values, for the multiplicity of tests. Depends on the multiple testing method (see function p.adjust).
`significant`	Indices of the time points for which the test is positive.
`pi0`	Value for pi0: if the input argument pi0 is NULL, the output is the estimated proportion of true null hypotheses using the method by Storey and Tibshirani (2003).
`test`	Pointwise factor-adjusted F-statistics if $p>1$, where p is the difference between the numbers of parameters in the nonnull and null models. Otherwise, if $p=1$, the function returns pointwise factor adjusted t-statistics (signed square-roots of F-statistics).
`df1`	Residual degrees of freedom for the nonnull model.
`df0`	Residual degrees of freedom for the null model.
`nbf`	Number of factors for the residual covariance model.
`signal`	Estimated signal: a pxT matrix, where p is the difference between the numbers of parameters in the nonnull and null models and T the number of frames.
`r2`	R-squared values for each of the T linear models.

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

References

Causeur, D., Chu, M.-C., Hsieh, S., Sheu, C.-F. 2012. A factor-adjusted multiple testing procedure for ERP data analysis. Behavior Research Methods, 44, 635-643.

Friguet, C., Kloareg, M., Causeur, D. 2009. A factor model approach to multiple testing under dependence. Journal of the American Statistical Association, 104, 1406-1415.

Leek, J.T., Storey, J.D. 2008. A general framework for multiple testing dependence. Proceedings of the National Academy of Sciences of the United States of America, 105, 18718-18723.

Sheu, C.-F., Perthame, E., Lee Y.-S. and Causeur, D. 2016. Accounting for time dependence in large-scale multiple testing of event-related potential data. Annals of Applied Statistics. 10(1), 219-245.

Storey, J. D., Tibshirani, R. 2003. Statistical significance for genome-wide experiments. Proceedings of the National Academy of Sciences of the United States of America, 100, 9440-9445.

Sun, Y., Zhang, N.R., Owen, A.B. 2012. Multiple hypothesis testing adjusted for latent variables, with an application to the AGEMAP gene expression data. The Annals of Applied Statistics, 6, no. 4, 1664-1688.

See Also

erpavetest, erptest, gbtest, p.adjust, pval.estimate.eta0

Examples

```
## Not run:

data(impulsivity)

# Paired t-tests for the comparison of the ERP curves in the two conditions,
# within experimental group High, at channel CPZ

erpdta.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),5:505]
   # ERP curves for subjects in group 'High', at channel CPZ
covariates.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),1:4]
covariates.highCPZ = droplevels(covariates.highCPZ)
   # Experimental covariates for subjects in group 'High', at channel CPZ

design = model.matrix(~C(Subject,sum)+Condition,data=covariates.highCPZ)
   # Design matrix to compare ERP curves in the two conditions
design0 = model.matrix(~C(Subject,sum),data=covariates.highCPZ)
   # Design matrix for the null model (no condition effect)

tests = erpfatest(erpdta.highCPZ,design,design0,nbf=NULL,wantplot=TRUE,significance="none")
   # with nbf=NULL and significance="none", just to choose a number of factors
Ftest = erpFtest(erpdta.highCPZ,design,design0,nbf=6)
  # with nbf=6 (approximate conservative recommendation based on the variance inflation plot)
   # Signal detection: test for an eventual condition effect.
Ftest$pval

tests = erpfatest(erpdta.highCPZ,design,design0,nbf=6)
   # Signal identification: which are the significant intervals?

time_pt = seq(0,1000,2)      # sequence of time points (1 time point every 2ms in [0,1000])
nbs = 20                       # Number of B-splines for the plot of the effect curve
effect=which(colnames(design)=="ConditionSuccess")
erpplot(erpdta.highCPZ,design=design,frames=time_pt,effect=effect,xlab="Time (ms)",
        ylab=expression(Effect~curve~(mu~V)),bty="l",ylim=c(-3,3),nbs=nbs,
        cex.axis=1.25,cex.lab=1.25,interval="simultaneous")
  # with interval="simultaneous", both the pointwise and the simultaneous confidence bands
   # are plotted
points(time_pt[tests$significant],rep(0,length(tests$significant)),pch=16,col="blue")
   # Identifies significant time points by blue dots
title("Success-Failure effect curve with 95 percent C.I.",cex.main=1.25)
mtext(paste("12 subjects - Group 'High' - ",nbs," B-splines",sep=""),cex=1.25)


## End(Not run)
```

---

erpFtest   *Functional Analysis-of-Variance (Anova) testing of Event-Related Potentials (ERP) data*

Description

Factor-based generalized Likelihood-Ratio Test for functional ANOVA in ERP designs. The procedure is described in details in Causeur, Sheu, Perthame and Rufini (2018).

Usage

```
erpFtest(dta,design,design0=NULL,nbf=NULL,pvalue=c("Satterthwaite","MC","none"),
nsamples=200,min.err=0.01,verbose=FALSE,
nbfmax=min(c(nsamples,nrow(design)))-ncol(design)-1,
wantplot=ifelse(is.null(nbf),TRUE,FALSE),svd.method=c("fast.svd","irlba"))
```

Arguments

dta	Data frame containing the ERP measurements: each column corresponds to a time frame and each row to a curve.
design	Design matrix of the nonnull model for the relationship between the ERP and the experimental variables. Typically the output of the function model.matrix
design0	Design matrix of the null model. Typically a submodel of the nonnull model, obtained by removing columns from design. Default is NULL, corresponding to the model with no covariates.
nbf	Number of factors in the residual covariance model. Default is NULL: the number of factors is determined by minimization of the variance inflation criterion as suggested in Friguet et al. (2009).
pvalue	If "Satterthwaite", then the Monte-Carlo p-value is calculated with a Satterthwaite approximation of the null distribution. The other possible value is "none", for no calculation of the p-value or "MC" for the Monte-Carlo p-value.
nsamples	Number of Monte-Carlo samples for the estimation of the null distribution. Default is nsamples=200, recommended when the Satterthwaite approximation is used.
min.err	Control parameter for convergence of the iterative algorithm. Default is 1e-02.
verbose	If TRUE, details are printed along the iterations of the algorithm. Default is FALSE.
nbfmax	Only required if nbf=NULL. The largest possible number of factors.
wantplot	If TRUE, a diagnostic plot is produced to help choosing the number of factors. Only active if nbf=NULL.
svd.method	the EM algorithm starts from an SVD estimation of the factor model parameters. The default option to implement this SVD is [fast.svd](). An alternative option is an approximate but faster SVD by function [irlba]().

Details

The method is described in Causeur et al. (2018). It can be viewed as a Hotelling's T-squared multivariate ANOVA test based on a factor decomposition of the time-dependence across point-wise F-tests. For 1-way functional ANOVA, it can be compared to function `anova.onefactor` in package `fda.usc`.

Value

Fgls	The generalized LRT statistics comparing the nonnull model with design matrix `design` and the null model with design matrix `design0`.
Fols	The LRT statistics ignoring time-dependence across pointwise F-tests.
pval	p-value of the generalized LRT statistics.
pval.Fols	p-value of the LRT statistics ignoring time-dependence across pointwise F-tests.

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France. Flavia Rufini, Agrocampus Ouest, Rennes, France.

References

Causeur, D., Sheu, C.-F., Perthame, E. and Rufini, F. (2018). A functional generalized F-test for signal detection with applications to event-related potentials significance analysis. Submitted.

Examples

```
data(impulsivity)

# Comparison of ERP curves in the two conditions, within experimental group 'High', at channel CPZ

erpdta.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),5:505]
   # ERP curves for subjects in group 'High'
covariates.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),1:4]
covariates.highCPZ = droplevels(covariates.highCPZ)
   # Experimental covariates for subjects in group 'High'

design = model.matrix(~C(Subject,sum)+Condition,data=covariates.highCPZ)
   # Design matrix to compare ERP curves in the two conditions
design0 = model.matrix(~C(Subject,sum),data=covariates.highCPZ)
   # Design matrix for the null model (no condition effect)

Ftest = erpFtest(erpdta.highCPZ,design,design0,nbf=NULL,wantplot=TRUE,pvalue="none")
   # with pvalue="none", just to choose a number of factors
Ftest = erpFtest(erpdta.highCPZ,design,design0,nbf=6)
Ftest$pval        # p-value of the Generalized LRT statistic
Ftest$pval.Fols   # p-value of the LRT statistic when time-dependence is ignored

time_pt = seq(0,1000,2)    # sequence of time points (1 time point every 2ms in [0,1000])
nbs = 20                    # Number of B-splines for the plot of the effect curve
effect=which(colnames(design)=="ConditionSuccess")
```

```
erpplot(erpdta.highCPZ,design=design,frames=time_pt,effect=effect,xlab="Time (ms)",
        ylab=expression(Effect~curve~(mu~V)),bty="l",ylim=c(-3,3),nbs=nbs,
        cex.axis=1.25,cex.lab=1.25,interval="simultaneous")
  # with interval="simultaneous", both the pointwise and the simultaneous confidence bands
   # are plotted
title("Success-Failure effect curve with 95 percent C.I.",cex.main=1.25)
mtext(paste("12 subjects - Group 'High' - ",nbs," B-splines",sep=""),cex=1.25)
```

---

erpplot	*Plot of ERP curves or effect curves (difference curve for example) with confidence intervals.*

Description

The function is designed to plot either raw ERP curves or estimated effect curves. In the case an effect curve is displayed, then spline smoothing can be used to obtained more regular curves. Pointwise or simultaneous confidence intervals can be added to the plot.

Usage

```
erpplot(dta,design=NULL,effect=1,interval=c("none","pointwise","simultaneous"),
alpha=0.05,frames=NULL,ylim=NULL,nbs=NULL, ...)
```

Arguments

dta	Data frame containing the ERP curves: each column corresponds to a time frame and each row to a curve.
design	Design matrix of the nonnull model for the relationship between the ERP and the experimental variables. Typically the output of the function model.matrix. Only needed if an effect curve is to be plotted.
effect	Integer value giving the column number of the design matrix corresponding to the effect curve to be plotted
interval	Type of confidence interval to be added to the plot of an effect curve. Default is "none" for no confidence interval. Other possibilities are "pointwise" for a pointwise confidence interval or "simultaneous" for both a pointwise and a simultaneous confidence interval.
alpha	The confidence level for the confidence interval is 1-alpha.
frames	Sequence of time frames. Default is NULL, in which case frames is just the sequence of intergers between one and the total number of frames.
ylim	Limits for the y-axis. Default is NULL (set into the function).
nbs	Number of B-spline basis for an eventual spline smoothing of the effect curve. Default is NULL for no smoothing.
...	Graphical parameters (see par) and any further arguments of plot, typically plot.default, may also be supplied as arguments to this function. Hence, the high-level graphics control arguments described under par and the arguments to title may be supplied to this function.

Value

The function generates a plot, but does not return any other numerical outputs.

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

See Also

matplot

Examples

```
data(impulsivity)

# Comparison of ERP curves in the two conditions, within experimental group 'High', at channel CPZ

erpdta.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),5:505]
   # ERP curves for subjects in group 'High'
covariates.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),1:4]
covariates.highCPZ = droplevels(covariates.highCPZ)
   # Experimental covariates for subjects in group 'High'

colors = ifelse(covariates.highCPZ$Condition=="Success","darkgray","orange")
   # Gives a color code for each condition
time_pt = seq(0,1000,2)     # sequence of time points (1 time point every 2ms in [0,1000])
erpplot(erpdta.highCPZ,frames=time_pt,xlab="Time (ms)",lty=1,col=colors,
        ylab=expression(ERPs~(mu~V)),bty="l",
        cex.axis=1.25,cex.lab=1.25)
   # Displays the ERP curves in group 'High', at channel CPZ
legend("topright",bty="n",lty=1,col=c("darkgray","orange"),legend=c("Success","Failure"))
title("ERP curves",cex.main=1.25)
mtext(paste("12 subjects - Group 'High'",sep=""),cex=1.25)


design = model.matrix(~C(Subject,sum)+Condition,data=covariates.highCPZ)
   # Design matrix to compare ERP curves in the two conditions
design0 = model.matrix(~C(Subject,sum),data=covariates.highCPZ)
   # Design matrix for the null model (no condition effect)

nbs = 20                    # Number of B-splines for the plot of the effect curve
effect=which(colnames(design)=="ConditionSuccess")
erpplot(erpdta.highCPZ,design=design,frames=time_pt,effect=effect,xlab="Time (ms)",
        ylab=expression(Effect~curve~(mu~V)),bty="l",ylim=c(-3,3),nbs=nbs,
        cex.axis=1.25,cex.lab=1.25,interval="simultaneous")
  # with interval="simultaneous", both the pointwise and the simultaneous confidence bands
   # are plotted
title("Success-Failure effect curve with 95 percent C.I.",cex.main=1.25)
mtext(paste("12 subjects - Group 'High' - ",nbs," B-splines",sep=""),cex=1.25)
```

---

erptest   *FDR- and FWER-controlling Multiple testing of ERP data*

Description

Classical FDR- and FWER-controlling multiple testing procedures for ERP data in a linear model framework.

Usage

```
erptest(dta,design,design0=NULL,
method=c("BH","holm","hochberg","hommel","bonferroni","BY","fdr","none"),alpha=0.05,
pi0 = 1, nbs = NULL)
```

Arguments

dta	Data frame containing the ERP curves: each column corresponds to a time frame and each row to a curve.
design	Design matrix of the nonnull model for the relationship between the ERP and the experimental variables. Typically the output of the function model.matrix
design0	Design matrix of the null model. Typically a submodel of the nonnull model, obtained by removing columns from design. Default is NULL, corresponding to the model with no covariates.
method	FDR- or FWER- controlling multiple testing procedures as available in the function p.adjust. Default is "BH", for the Benjamini-Hochberg procedure (see Benjamini and Hochberg, 1995).
alpha	The FDR or FWER control level. Default is 0.05
pi0	An estimate of the proportion of true null hypotheses, which can be plugged into an FDR controlling multiple testing procedure to improve its efficiency. Default is 1, corresponding to the classical FDR controlling procedures. If NULL, the proportion is estimated using the function pval.estimate.eta0 of package fdrtool, with the method proposed by Storey and Tibshirani (2003).
nbs	Number of B-spline basis for an eventual spline smoothing of the effect curve. Default is NULL for no smoothing.

Value

pval	p-values of the tests.
correctedpval	Corrected p-values, for the multiplicity of tests. Depends on the multiple testing method (see function p.adjust).
significant	Indices of the time points for which the test is positive.
pi0	Value for pi0: if the input argument pi0 is NULL, the output is the estimated proportion of true null hypotheses using the method by Storey and Tibshirani (2003).

test	Pointwise F-statistics if p>1, where p is the difference between the numbers of parameters in the nonnull and null models. Otherwise, if p=1, the function returns pointwise t-statistics (signed square-roots of F-statistics).
df1	Residual degrees of freedom for the nonnull model.
df0	Residual degrees of freedom for the null model.
signal	Estimated signal: a p x T matrix, where p is the difference between the numbers of parameters in the nonnull and null models and T the number of time points.
sd	T-vector of estimated residual standard deviations where T is the number of time points.
r2	R-squared values for each of the T linear models.
sdsignal	Standard deviations of the estimated signal: a p x T matrix, where p is the difference between the numbers of parameters in the nonnull and null models and T the number of time points.
residuals	n x T matrix of residuals of the fit of the nonnull model.
coef	Estimated regression coefficients: a q x T matrix, where q is the number of parameters in the nonnull model and T the number of time points.

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

See Also

[erpavetest](), [erpfatest](), [gbtest](), [p.adjust]()

Examples

```
data(impulsivity)

# Paired t-tests for the comparison of the ERP curves in the two conditions,
# within experimental group High, at channel CPZ

erpdta.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),5:505]
   # ERP curves for subjects in group 'High'
covariates.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),1:4]
covariates.highCPZ = droplevels(covariates.highCPZ)
   # Experimental covariates for subjects in group 'High'

design = model.matrix(~C(Subject,sum)+Condition,data=covariates.highCPZ)
   # Design matrix to compare ERP curves in the two conditions
design0 = model.matrix(~C(Subject,sum),data=covariates.highCPZ)
   # Design matrix for the null model (no condition effect)

tests = erptest(erpdta.highCPZ,design,design0)

time_pt = seq(0,1000,2)     # sequence of time points (1 time point every 2ms in [0,1000])
nbs = 20                      # Number of B-splines for the plot of the effect curve
effect=which(colnames(design)=="ConditionSuccess")
erpplot(erpdta.highCPZ,design=design,frames=time_pt,effect=effect,xlab="Time (ms)",
```

```
        ylab=expression(Effect~curve~(mu~V)),bty="l",ylim=c(-3,3),nbs=nbs,
        cex.axis=1.25,cex.lab=1.25,interval="simultaneous")
  # with interval="simultaneous", both the pointwise and the simultaneous confidence bands
   # are plotted
points(time_pt[tests$significant],rep(0,length(tests$significant)),pch=16,col="blue")
   # Identifies significant time points by blue dots
title("Success-Failure effect curve with 95 percent C.I.",cex.main=1.25)
mtext(paste("12 subjects - Group 'High' - ",nbs," B-splines",sep=""),cex=1.25)
```

---

gbtest	*The Guthrie-Buchwald procedure for significance analysis of ERP* *data*

Description

Monte-Carlo implementation of the Guthrie-Buchwald procedure (see Guthrie and Buchwald, 1991) which accounts for the auto-correlation among test statistics to control erroneous detections of short intervals.

Usage

```
gbtest(dta, design, design0 = NULL, graphthresh = 0.05, nsamples = 1000)
```

Arguments

dta	Data frame containing the ERP curves: each column corresponds to a time frame and each row to a curve.
design	Design matrix of the nonnull model for the relationship between the ERP and the experimental variables. Typically the output of the function model.matrix
design0	Design matrix of the null model. Typically a submodel of the nonnull model, obtained by removing columns from design. Default is NULL, corresponding to the model with no covariates.
graphthresh	Graphical threshold (see Guthrie and Buchwald, 1991). Default is 0.05. As the FDR control level, the smaller is the graphical threshold, the more conservative is the procedure.
nsamples	Number of samples in the Monte-Carlo method to estimate the residual covariance. Default is 1000.

Details

The Guthrie-Buchwald method starts from a preliminary estimation of r, the lag-1 autocorrelation, among test statistics. Then, the null distribution of the lengths of the intervals I_alpha = t : pvalue_t <= alpha , where alpha is the so-called graphical threshold parameter of the method, is obtained using simulations of p-values p_t associated to auto-regressive t-test process of order 1 with mean 0 and auto-correlation r. Such an interval I_alpha is declared significant if its length exceeds the (1-alpha)-quantile of the null distribution. Note that the former method is designed to control erroneous detections of short significant intervals but not to control any type-I error rate.

Value

nbsignifintervals
: Number of significant intervals.

intervals
: List of length nbsignifintervals which components give the indices of each significant intervals.

significant
: Indices of the time points for which the test is positive.

signal
: Estimated signal: a pxT matrix, where p is the difference between the numbers of parameters in the nonnull and null models and T the number of frames.

rho
: Estimated lag-1 auto-correlation.

r2
: R-squared values for each of the T linear models.

Author(s)

David Causeur - david.causeur@agrocampus-ouest.fr and Mei-Chen Chu (National Cheng-Kung University, Tainan, Taiwan)

References

Guthrie, D. and Buchwald, J.S. (1991). Significance testing of difference potentials. Psychophysiology, 28, 240-244.

Sheu, C.-F., Perthame, E., Lee Y.-S. and Causeur, D. (2016). Accounting for time dependence in large-scale multiple testing of event-related potentials data. Annals of Applied Statistics. 10(1), 219-245.

See Also

[erpavetest](), [erpfatest](), [erptest]()

Examples

```
data(impulsivity)

# Paired t-tests for the comparison of the ERP curves in the two conditions,
# within experimental group High, at channel CPZ

erpdta.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),5:505]
   # ERP curves for subjects in group 'High'
covariates.highCPZ = impulsivity[(impulsivity$Group=="High")&(impulsivity$Channel=="CPZ"),1:4]
covariates.highCPZ = droplevels(covariates.highCPZ)
   # Experimental covariates for subjects in group 'High'

design = model.matrix(~C(Subject,sum)+Condition,data=covariates.highCPZ)
   # Design matrix to compare ERP curves in the two conditions
design0 = model.matrix(~C(Subject,sum),data=covariates.highCPZ)
   # Design matrix for the null model (no condition effect)

tests = gbtest(erpdta.highCPZ,design,design0)

time_pt = seq(0,1000,2)     # sequence of time points (1 time point every 2ms in [0,1000])
```

```
nbs = 20                    # Number of B-splines for the plot of the effect curve
effect=which(colnames(design)=="ConditionSuccess")
erpplot(erpdta.highCPZ,design=design,frames=time_pt,effect=effect,xlab="Time (ms)",
        ylab=expression(Effect~curve~(mu~V)),bty="l",ylim=c(-3,3),nbs=nbs,
        cex.axis=1.25,cex.lab=1.25,interval="simultaneous")
  # with interval="simultaneous", both the pointwise and the simultaneous confidence bands
   # are plotted
points(time_pt[tests$significant],rep(0,length(tests$significant)),pch=16,col="blue")
   # Identifies significant time points by blue dots
title("Success-Failure effect curve with 95 percent C.I.",cex.main=1.25)
mtext(paste("12 subjects - Group 'High' - ",nbs," B-splines",sep=""),cex=1.25)
```

---

ifa   *Inverse of a matrix based on its factor decomposition.*

Description

This function is not supposed to be called directly by users. It is needed in function emfa implementing an EM algorithm for a factor model.

Usage

```
ifa(Psi, B)
```

Arguments

Psi	m-vector of specific variances, where m is the number of rows and columns of the matrix.
B	m x q matrix of loadings, where q is the number of factors

Details

If Sigma has the q-factor decomposition Sigma=diag(Psi)+BB', then Sigma^-1 has the corresponding q-factor decomposition Sigma^-1=diag(phi)(I-theta theta')diag(phi), where theta is a mxq matrix and phi the vector of inverse specific standard deviations.

Value

iS	m x m inverse of diag(Psi)+BB'.
iSB	m x q matrix (diag(Psi)+BB')^-1B.
Phi	m-vector of inverse specific standard deviations.
Theta	mxq matrix of loadings for the inverse factor model (see the details Section).

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

References

Woodbury, M.A. (1949) The Stability of Out-Input Matrices. Chicago, Ill., 5 pp

See Also

emfa, isqrtfa

Examples

```
data(impulsivity)
erpdta = as.matrix(impulsivity[,5:505])   # erpdta contains the whole set of ERP curves
fa = emfa(erpdta,nbf=20)          # 20-factor modelling of the ERP curves in erpdta
Sfa = diag(fa$Psi)+tcrossprod(fa$B) # Factorial estimation of the variance
iSfa = ifa(fa$Psi,fa$B)$iS        # Matrix inversion
max(abs(crossprod(Sfa,iSfa)-diag(ncol(erpdta)))) # Checks that Sfa x iSfa = diag(ncol(erpdta))
```

---

impulsivity	*Event-Related Potentials data from a study conducted by Shen et al. (2014) to investigate neural correlates of impulsive behavior.*

Description

In the study, adult males who scored at the top 25% and bottom 25% on a scale measuring trait impulsivity were recruited to participate in an ERP experiment. There were twelve subjects in each impulsivity trait group (high or low). Specifically, the influence of trait impulsivity in response inhibition was examined in a two-choice discrimination task using the stop-signal paradigm (Logan et al, 1984). Participants were asked to withhold their responses when a 'stop' signal appeared after the 'go' stimulus. ERP amplitudes from 32 scalp locations (channels) were recorded from 200 ms before signal onset till 1000 ms after with one reading per 2 ms. Two of the 32 channels were excluded in the analysis by the authors. The response inhibition is a within-subject factor with two levels (successful or failed).

Usage

```
data("impulsivity")
```

Format

A data frame with 144 observations on the following 505 variables. The first four variables are categorical: Channel with levels CPZ CZ FCZ giving the electrode position, Subject with levels S1 S2 ... to identify the subjects, Group with levels High Low for the membership to an impulsivity group and Condition with levels Failure Success for the response inhibition condition. The remaining 501 variables are the ERP values at time 0 (onset), 2ms, 4ms, ..., 1000ms.

Source

Shen, I., Lee, D., and Chen, C. (2014), The role of trait impulsivity in response inhibition: event-related potentials in a stop-signal task. International Journal of Psychophysiology, 91(2).

References

Causeur, D., Sheu, C.-F., Perthame, E. and Rufini, F. (2018). A functional generalized F-test for signal detection with applications to event-related potentials significance analysis. Submitted.

Logan, G. D., Cowan, W., and Davis, K. (1984), On the ability to inhibit simple and choice reaction time responses: a model and a method. Journal of Experimental Psychology: Human Perception and Performance, 10(2), 276-291.

Examples

```
data(impulsivity)

head(impulsivity[,1:10]) # each row of impulsivity contains an ERP curve (from column 5 to 505)
                  # Column 1: electrode position (channel)
                  # Column 2: subject id
                  # Column 3: impulsivity group (High/Low)
                  # Column 4: response inhibition condition (Success/Failure)

time_pt = seq(0, 1000, 2)    # sequence of time points (1 time point every 2ms in [0,1000])
erpdta = impulsivity[,5:505]         # erpdta contains the ERP curves
T = length(time_pt)          # number of time points

covariates = impulsivity[,1:4]       # contains the experimental covariates of interest

# Impulsivity ERP design

with(data=covariates,table(Channel,Condition,Group))
   # 48 ERP curves for each of the 3 channels
   # Within each channel, 12 ERP curves in each Condition x Group
   # Within each channel, 12 subjects in group High, 12 in group Low
```

---

isqrtfa   *Inverse square-root of a matrix based on its factor decomposition.*

Description

This function is not supposed to be called directly by users. It is needed in functions `erpfatest` and `erpFtest` implementing factor-adjusted testing methods.

Usage

```
isqrtfa(Psi, B)
```

Arguments

Psi   m-vector of specific variances, where m is the number of rows and columns of
               the matrix.

B   mxq matrix of loadings, where q is the number of factors

Details

If Sigma has the q-factor decomposition Sigma=diag(Psi)+BB', then the function returns a matrix Omega such that Sigma^-1 = Omega Omega'. Equivalently, Omega' Sigma Omega = I.

Value

The mxm matrix Omega (see the details Section).

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

References

Woodbury, M.A. (1949) The Stability of Out-Input Matrices. Chicago, Ill., 5 pp

See Also

emfa, [ifa](ifa)

Examples

```
data(impulsivity)
erpdta = as.matrix(impulsivity[,5:505])   # erpdta contains the whole set of ERP curves
fa = emfa(erpdta,nbf=20)           # 20-factor modelling of the ERP curves in erpdta
Sfa = diag(fa$Psi)+tcrossprod(fa$B) # Factorial estimation of the variance
iSfa = ifa(fa$Psi,fa$B)$iS         # Matrix inversion
isqrtSfa = isqrtfa(fa$Psi,fa$B)   # Inverse square-root of Sfa
max(abs(tcrossprod(isqrtSfa)-iSfa)) # Checks that isqrtSfa x t(isqrtSfa) = iSfa
```

---

nbfactors   *Determination of the number of factors in high dimensional factor models.*

Description

This function is not supposed to be called directly by users. It implements the variance inflation method proposed by Friguet et al. (2009) to choose the number of factors in high dimensional factor models.

Usage

```
nbfactors(dta,maxnbfactors=15,diagnostic.plot=FALSE,min.err=0.001,verbose=FALSE,
svd.method=c("fast.svd","irlba"))
```

Arguments

dta	n x m scaled matrix which rows are multivariate m-profiles for n individuals. n can be much smaller than m. Columns are all centered with standard deviations 1.
maxnbfactors	Maximum number of factors. It has to be a positive integer, smaller than the rank of dta.
diagnostic.plot	
	Logical value. If TRUE, a plot dispalying the variance inflation curve is produced, with recommendations for the optimal number of factors.
min.err	stopping criterion for the iterative algorithm. Maximum difference between the estimated parameters in the last two iterations.
verbose	logical value. If verbose=TRUE, then some information is printed along the calculation.
svd.method	the EM algorithm starts from an SVD estimation of the factor model parameters. The default option to implement this SVD is `fast.svd`. An alternative option is an approximate but faster SVD by function `irlba`.

Details

It is highly recommended to run the function first with diagnostic.plot=TRUE and to choose the number of factors based on the plot that will be produced. Two recommendations are provided: a conservative one obtaned using a kind of elbow criterion, and a liberal one that minimizes the variance inflation curve.

Value

sdt	Variance inflation values for a number of factors going from 0 to maxnbfactors.
nbf	Recommendation for an optimal number of factors (the more liberal, see the details Section for more information).

Author(s)

David Causeur, IRMAR, UMR 6625 CNRS, Agrocampus Ouest, Rennes, France.

References

Friguet, C., Kloareg, M. and Causeur, D. (2009). A factor model approach to multiple testing under dependence. Journal of the American Statistical Association. 104 (488), 1406-1415.

See Also

emfa

Examples

```
data(impulsivity)
erpdta = as.matrix(impulsivity[,5:505])   # erpdta contains the whole set of ERP curves
nbf = nbfactors(scale(erpdta),maxnbfactors=10,diagnostic.plot=TRUE)
```

Index