Package 'FAMT'

January 20, 2025

Type Package

Title Factor Analysis for Multiple Testing (FAMT) : Simultaneous Tests under Dependence in High-Dimensional Data

Version 2.6

Date 2022-05-06

Author David Causeur, Chloe Friguet, Magalie Houee-Bigot, Maela Kloareg

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

Depends R (>= 3.5.0)

Imports mnormt, impute

Description

The method proposed in this package takes into account the impact of dependence on the multiple testing procedures for high-throughput data as proposed by Friguet et al. (2009). The common information shared by all the variables is modeled by a factor analysis structure. The number of factors considered in the model is chosen to reduce the false discoveries variance in multiple tests. The model parameters are estimated thanks to an EM algorithm. Adjusted tests statistics are derived, as well as the associated p-values. The proportion of true null hypotheses (an important parameter when controlling the false discovery rate) is also estimated from the FAMT model. Graphics are proposed to interpret and describe the factors.

LazyLoad yes

License GPL (>= 2)

URL http://famt.free.fr/

NeedsCompilation no

Repository CRAN

Date/Publication 2022-05-09 10:20:02 UTC

Contents

FAMT-package | *Factor Analysis for Multiple Testing (FAMT) : simultaneous tests under dependence in high-dimensional data*

Description

The method proposed in this package takes into account the impact of dependence on multiple testing procedures for high-throughput data as proposed by Friguet et al. (2009). The common information shared by all the variables is modeled by a factor analysis structure. The number of factors considered in the model is chosen to reduce the variance of the number of false discoveries. The model parameters are estimated thanks to an EM algorithm. Factor-adjusted tests statistics are derived, as well as the associated p-values. The proportion of true null hypotheses (an important parameter when controlling the false discovery rate) is also estimated from the FAMT model. Diagnostic plots are proposed to interpret and describe the factors.

Details

Package:	FAMT
Type:	Package
Version:	1.0
Date:	2010-05-03
License:	GPL
LazyLoad:	yes

The `as.FAMTdata` function creates a single R object containing the data stored: - in one mandatory data-frame: the 'expression' dataset with m rows (if m tests) and n columns (n is the sample size) containing the observations of the responses. - and two optional data-frames: the 'covariates' dataset with n rows and at least 2 columns, one giving the specification to match 'expression' and 'covariates' and the other one containing the observations of at least one covariate. The optional dataset, 'annotations' can be provided to help interpreting the factors: with m rows and at least one column to identify the variables (ID).

The whole multiple testing procedure is provided in a single function, `modelFAMT`, but you can also choose to apply the procedure step by step, using the functions :

nbfactors (Estimation of the optimal number of factors) emfa (EM fitting of the Factor Analysis model).

The modelFAMT also provides the individual test statistics and corresponding p-values like the raw.pvalues function.

A function summaryFAMT provides some key elements of classical summaries either on 'FAMTdata' or 'FAMTmodel'.

The estimation of the proportion of true null hypotheses from a 'FAMTmodel' is done by the function pi0FAMT.

The defacto function provides diagnostic plots to interpret and describe the factors.

Author(s)

David Causeur, Chloe Friguet, Magalie Houee-Bigot, Maela Kloareg.

Maintainer: <David.Causeur@agrocampus-ouest.fr>

References

Causeur D., Friguet C., Houee-Bigot M., Kloareg M. (2011). Factor Analysis for Multiple Testing (FAMT): An R Package for Large-Scale Significance Testing Under Dependence. Journal of Statistical Software, 40(14),1-19. https://www.jstatsoft.org/v40/i14

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, p.1406-1415

http://famt.free.fr/

---

annotations *Gene annotations data frame*

Description

A data frame with 6 columns describing the 9893 genes, which expressions are stored in the 'expression' dataset, in terms of functional categories, oligonucleotide size and location on the microarray. See also expression, covariates.

Usage

```
data(annotations)
```

Format

A data frame with 9893 observations on the following 6 variables.

ID Gene identification

Name Gene annotation (functional categories) (character)

Block Location on the microarray(factor)

Column Location on the microarray (factor)

Row Location on the microarray (factor)

Length Oligonucleotide size (numeric vector)

Source

UMR Genetique Animale - INRA/AGROCAMPUS OUEST - Rennes, France.

References

Blum Y., Le Mignon G., Lagarrigue S. and Causeur S. (2010) - A factor model to analyze heterogeneity in gene expression, BMC Bioinformatics, 11:368.

Le Mignon, G. and Desert, C. and Pitel, F. and Leroux, S. and Demeure, O. and Guernec, G. and Abasht, B. and Douaire, M. and Le Roy, P. and Lagarrigue S. (2009) - Using trancriptome profiling to characterize QTL regions on chicken chromosome 5. BMC Genomics, 10:575.

Examples

```
data(annotations)
dim(annotations)
summary(annotations)
```

---

as.FAMTdata   *Create a 'FAMTdata' object from an expression, covariates and annotations dataset*

Description

The function creates a 'FAMTdata' object containing the expression, the covariates and the annotations dataset if provided. The function checks the consistency of dataframes between them. Then missing values of expression can be imputed.

Usage

```
as.FAMTdata(expression, covariates = NULL, annotations = NULL, idcovar = 1,
idannot = NULL, na.action=TRUE)
```

Arguments

expression	An expression data frame with genes in rows and arrays in columns. The arrays are identified by the column names.
covariates	An optional data frame with arrays in rows, and covariates in columns. One column must contain the array identification (NULL by default).
annotations	An optional data frame containing informations on the genes (NULL by default)
idcovar	The column number corresponding to the array identification in the covariates data frame (1 by default)
idannot	The column number corresponding to the gene identification in annotations data frame (NULL by default)
na.action	If TRUE (default value), missing expression data are imputed using nearest neighbor averaging ([impute.knn](#) function of 'impute' package).

Details

The `as.FAMTdata` function creates a single R object containing the data stored: - in one mandatory data-frame: the 'expression' dataset with m rows (if m tests) and n columns (n is the sample size) containing the observations of the responses. - and two optional data frames: the 'covariates' dataset with n rows and at least 2 columns, one giving the specification to match 'expression' and 'covariates' and the other one containing the observations of at least one covariate. The optional dataset,'annotations', can be provided to help interpreting the factors: with m rows and at least one column to identify the variables (ID).

Value

expression	The expression data frame
covariates	The optional covariates data frame
annotations	The optional data frame containing annotations. The genes annotations such as the functional categories should be in a character form, not in a factor form.
idcovar	The column number corresponding to the array identification in the covariate data frame (which should correspond to the column names in 'expression')
na.expr	Rows and columns of expression with missing values

Note

The class of the data produced with the `as.FAMTdata` function is called 'FAMTdata'. We advise to carry out a summary of FAMT data with the function `summaryFAMT`.

Author(s)

David Causeur

See Also

`summaryFAMT`

Examples

```
# The data are divided into one mandatory data-frame, the gene expressions,
#  and two optional datasets: the covariates, and the annotations.

# The expression dataset with 9893 rows (genes) and 43 columns (arrays)
#  containing the observations of the responses.
# The covariates dataset with 43 rows (arrays) and 6 columns:
#  the second column gives the specification to match 'expression'
#  and 'covariates' (array identification), the other ones contain
#  the observations of covariates.
# The annotations dataset contains 9893 rows (genes) and
#  6 columns to help interpreting the factors, the first one (ID)
#  identifies the variables (genes).

data(expression)
data(covariates)
```

```
data(annotations)

# Create the 'FAMTdata'
#############################################
chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)
# 'FAMTdata' summary
summaryFAMT(chicken)
```

---

covariates   *Covariates data frame*

Description

A data frame with 6 covariates in columns and 43 arrays in rows, describing the arrays of the expression dataset. See also expression, annotations

Usage

```
data(covariates)
```

Format

A data frame with 43 observations on the following 6 variables.

AfClass a factor with levels F (Fat) L (Lean) NC (Intermediate) giving the abdominal fatness class

ArrayName Identifying the arrays (character)

Mere a factor with 8 levels giving the dam of the offsprings

Lot a factor with 4 levels giving the hatch

Pds9s a numeric vector giving the body weight

Af a numeric vector giving the abdominal fatness, the experimental condition of main interest in this example

Source

UMR Genetique Animale - INRA/AGROCAMPUS OUEST - Rennes, France.

References

Blum Y., Le Mignon G., Lagarrigue S. and Causeur S. (2010) - A factor model to analyze heterogeneity in gene expression, BMC Bioinformatics, 11:368.

Le Mignon, G. and Desert, C. and Pitel, F. and Leroux, S. and Demeure, O. and Guernec, G. and Abasht, B. and Douaire, M. and Le Roy, P. and Lagarrigue S. (2009) - Using trancriptome profling to characterize QTL regions on chicken chromosome 5. BMC Genomics, 10:575.

Examples

```
data(covariates)
dim(covariates)
summary(covariates)
```

---

defacto	*FAMT factors description*

Description

This function helps the user to describe and interpret the factors using some available external information on either genes or arrays. Diagnostic plots are provided.

Usage

```
defacto(model, plot = TRUE, axes = c(1, 2), select.covar = NULL,
select.annot = NULL, lim.b = 0.01, lab = TRUE, cex = 1)
```

Arguments

model	'FAMTmodel' object
plot	Boolean (TRUE by default). If TRUE, diagnostic plots are provided (unless the 'FAMTmodel' has less than one factor).
axes	Vector of length 2, specifying the factors to plot.
select.covar	Selection of external covariates. If NULL (default value), the function takes all covariates except the array identifiers and those used in the model.
select.annot	Selection of external annotations. If NULL (default value), the function takes all the available factors in 'annotations'.
lim.b	Proportion of variables with the highest loadings for each factor to appear on plots or in tables (0.01 by default).
lab	Boolean. If TRUE (default value), array names are labeled on the figure
cex	A numerical value giving the amount by which plotting text and symbol should be enlarged relative to the default (1 is the default value)

Value

loadings	highest loadings (B matrix) for each factor. The proportion of loading is determined by "lim.b"
covariates	Matrix of p-values for the tests of linear relationships between scores on each factor (rows) and external covariates (columns).
annotations	Matrix of p-values for the tests of linear relationships between loadings of each factor (rows) and external annotations (columns).

Author(s)

David Causeur, Maela Kloareg

See Also

[as.FAMTdata](), [modelFAMT]()

Examples

```
## FAMT data
data(expression)
data(covariates)
data(annotations)

# Create the 'FAMTdata'
###########################################
chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)
# 'FAMTdata' summary
## Not run: summaryFAMT(chicken)

# FAMT complete multiple testing procedure
###########################################
model = modelFAMT(chicken,x=c(3,6),test=6,nbf=3)
# summary on the 'FAMT model'
## Not run: summaryFAMT(model)

# Factors description
###########################################
chicken.defacto = defacto(model,axes=1:2,select.covar=4:5,select.annot=3:6,
cex=0.6)
```

---

emfa   *Factor Analysis model adjustment with the EM algorithm*

Description

A function to fit a Factor Analysis model with the EM algorithm.

Usage

```
emfa(data, nbf, x = 1, test = x[1], pvalues = NULL, min.err = 0.001)
```

Arguments

data	'FAMTdata' object, see [as.FAMTdata]()
nbf	Number of factors of the FA model, see [nbfactors]()
x	Column number(s) corresponding to the experimental condition and the optional covariates (1 by default) in the covariates data frame.

test	Column number corresponding to the experimental condition (x[1] by default) on which the test is performed.
pvalues	p-values of the individual tests. If NULL, the classical procedure is applied (see [raw.pvalues](#))
min.err	Stopping criterion value for iterations in EM algorithm (default value: 0.001)

Details

In order to use this function, the number of factors is needed (otherwise, use [nbfactors](#)).

Value

B	Estimation of the loadings
Psi	Estimation of Psi
Factors	Scores of the individuals on the factors
commonvar	Proportion of genes common variance (modeled on the factors)
SelectHo	Vector of row numbers corresponding to the non-significant genes

Author(s)

David Causeur

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, p.1406-1415

See Also

[as.FAMTdata](#), [nbfactors](#)

Examples

```
## Reading 'FAMTdata'
data(expression)
data(covariates)
data(annotations)
chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)

# EM fitting of the Factor Analysis model
chicken.emfa = emfa(chicken,nbf=3,x=c(3,6),test=6)
chicken.emfa$commonvar
```

---

expression   *Gene expressions data frame*

Description

This dataset concerns hepatic transciptome profiles of 43 half sib male chickens selected for their variability on abdominal fatness (AF). Genes are in rows (9893 genes) and arrays in columns (43 arrays).

Usage

```
data(expression)
```

Format

A data frame with 9893 genes on 43 arrays.

Source

UMR Genetique Animale - INRA/AGROCAMPUS OUEST - Rennes, France.

References

Blum Y., Le Mignon G., Lagarrigue S. and Causeur S. (2010) - A factor model to analyze heterogeneity in gene expression, BMC Bioinformatics, 11:368.

Le Mignon, G. and Desert, C. and Pitel, F. and Leroux, S. and Demeure, O. and Guernec, G. and Abasht, B. and Douaire, M. and Le Roy, P. and Lagarrigue S. (2009) - Using trancriptome profling to characterize QTL regions on chicken chromosome 5. BMC Genomics, 10:575.

Examples

```
data(expression)
dim(expression)
summary(expression)
```

---

modelFAMT   *The FAMT complete multiple testing procedure*

Description

This function implements the whole FAMT procedure (including nbfactors and emfa). The number of factors considered in the model is chosen to reduce the variance of the number of the false discoveries. The model parameters are estimated using an EM algorithm. Factor-adjusted tests statistics are derived, as well as the corresponding p-values.

Usage

```
modelFAMT(data, x = 1, test = x[1], nbf = NULL, maxnbfactors = 8,
min.err = 0.001)
```

Arguments

data	'FAMTdata' object, see `as.FAMTdata`
x	Column number(s) corresponding to the experimental condition and the optional covariates (1 by default) in the covariates data frame.
test	Column number corresponding to the experimental condition (x[1] by default) one which the test is performed.
nbf	The number of factors of the FA model (NULL by default). If NULL, the function estimates the optimal nbf (see `nbfactors`)
maxnbfactors	The maximum number of factors (8 by default)
min.err	Stopping criterion value for iterations (default value:0.001)

Value

adjpval	Vector of FAMT factor-adjusted p-values
adjtest	Vector of FAMT factor-adjusted F statistics
adjdata	Factor-adjusted FAMT data
FA	Estimation of the FA model parameters
pval	Vector of classical p-values
x	Column number(s) corresponding to the experimental condition and the optional covariates in the covariates data frame
test	Column number corresponding to the experimental condition on which the test is performed
nbf	The number of factors used to fit the FA model
idcovar	The column number used for the array identification in the 'covariates' data frame

Note

The user can perform individual test statistics putting the number of factors (nbf) equal to zero. The result of this function is a 'FAMTmodel'. It is used as argument in other functions of the package : `summaryFAMT`, `pi0FAMT` or `defacto`. We advise to carry out a summary of FAMT model with the function `summaryFAMT`.

Author(s)

David Causeur

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, p.1406-1415

See Also

as.FAMTdata, raw.pvalues, nbfactors, emfa, summaryFAMT

Examples

```
## Reading 'FAMTdata'
data(expression)
data(covariates)
data(annotations)

chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)

# Classical method with modelFAMT
## Not run: modelpval=modelFAMT(chicken,x=c(3,6),test=6,nbf=0)
## Not run: summaryFAMT(modelpval)

# FAMT complete multiple testing procedure
# when the optimal number of factors is unknown
## Not run: model = modelFAMT(chicken,x=c(3,6),test=6)

# when the optimal number of factors has already been estimated
 model = modelFAMT(chicken,x=c(3,6),test=6,nbf=3)

summaryFAMT(model)
hist(model$adjpval)
## End(Not run)
```

---

nbfactors   *Estimation of the optimal number of factors of the FA model*

Description

The optimal number of factors of the FA model is estimated to minimize the variance of the number of false positives (see Friguet et al., 2009).

Usage

```
nbfactors(data, x = 1, test = x[1], pvalues = NULL, maxnbfactors = 8,
diagnostic.plot = FALSE, min.err = 0.001)
```

Arguments

data	'FAMTdata' object, see as.FAMTdata
x	Column number(s) corresponding to the experimental condition and the optional covariates (1 by default) in the covariates data frame
test	Column number corresponding to the experimental condition (x[1] by default) on which the test is performed

pvalues	Vector of p-values for the individual tests. If NULL, the classical procedure is applied (see `raw.pvalues`)
maxnbfactors	The maximum number of factors for the FA model (8 by default)
diagnostic.plot	
	boolean (FALSE by default). If TRUE, the values of the variance inflation criteria for each number of factors are plotted
min.err	Stopping criterion value for iterations (default value : 0.001)

Value

optimalnbfactors	
	Optimal number of factors of the FA model (an elbow criterion is used)
criterion	Variance criterion for each number of factors

Author(s)

David Causeur

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, p.1406-1415

See Also

`as.FAMTdata`, `emfa`

Examples

```
## Reading 'FAMTdata'
data(expression)
data(covariates)
data(annotations)
chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)

# Estimation of the number of factors
## Not run: nbfactors(chicken,x=c(3,6),test=6)

# Estimation of the number of factors with a graph of variance inflation
# criterion
## Not run: nbfactors(chicken,x=c(3,6),test=6, diagnostic.plot=TRUE)
```

---

pi0FAMT   *Estimation of the Proportion of True Null Hypotheses*

Description

A function to estimate the proportion pi0 of true null hypotheses from a 'FAMTmodel' (see also function "pval.estimate.eta0" in package "fdrtool").

Usage

```
pi0FAMT(model, method = c("smoother", "density"),
diagnostic.plot = FALSE)
```

Arguments

model   'FAMTmodel' object (see [modelFAMT](#))

method   algorithm used to estimate the proportion of null p-values. Available options are
                 "density" and "smoother" (as described in Friguet and Causeur, 2010)

diagnostic.plot

         if TRUE the histogram of the p-values with the estimate of pi0 horizontal line is
plotted. With the "smoother" method, an additional graph is displayed showing
the spline curve used to estimate pi0. With the "density" method, the estimated
convex density of the p-values is plotted onto the histogram

Details

The quantity pi0, i.e. the proportion of null hypotheses, is an important parameter when controlling the false discovery rate (FDR). A conservative choice is pi0 = 1 but a choice closer to the true value will increase efficiency and power - see Benjamini and Hochberg (1995, 2000), Black(2004) and Storey (2002) for details. The function pi0FAMT provides 2 algorithms to estimate this proportion. The "density" method is based on Langaas et al. (2005)'s approach where the density of p-values f(p) is first estimated considering f as a convex function, and the estimation of pi0 is got for p=1. The "smoother" method uses the smoothing spline approach proposed by Storey and Tibshirani(2003).

Value

pi0The estimated proportion pi0 of null hypotheses.

Author(s)

Chloe Friguet & David Causeur

References

Friguet C. and Causeur D. (2010) Estimation of the proportion of true null hypohteses in high-dimensional data under dependence. Submitted.

"density" procedure: Langaas et al (2005) Estimating the proportion of true null hypotheses, with application to DNA microarray data. JRSS. B, 67, 555-572.

"smoother" procedure: Storey, J. D., and R. Tibshirani (2003) Statistical significance for genome-wide experiments. Proc. Nat. Acad. Sci. USA, 100, 9440-9445.

See Also

[modelFAMT](#)

Examples

```
# Reading 'FAMTdata'
data(expression)
data(covariates)
data(annotations)
chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)

# FAMT complete multiple testing procedure
model = modelFAMT(chicken,x=c(3,6),test=6,nbf=3)

# Estimation of the Proportion of True Null Hypotheses
# "density" method
## Not run:  pi0FAMT(model,method="density",diagnostic.plot=TRUE)

# "smoother" method
pi0FAMT(model,method="smoother",diagnostic.plot=TRUE)
```

---

raw.pvalues	*Calculation of classical multiple testing statistics and p-values*

Description

Calculates for each gene expression, the Fisher test statistics and the corresponding p-value for H0: the gene expression does not depend on the experimental condition in a model with possible covariates.

Usage

```
raw.pvalues(data, x = 1, test = x[1])
```

Arguments

data	'FAMTdata' object, see `as.FAMTdata`
x	Column number(s) corresponding to the experimental condition and the optional covariates (1 by default) in the 'covariates' data frame.
test	Column number corresponding to the experimental condition (x[1] by default) of interest in the multiple testing procedure.

Value

pval	Vector containing the p-values
test	Vector containing the F statistics
resdf	Residual degrees of freedom

Author(s)

David Causeur

See Also

`as.FAMTdata`

Examples

```
data(expression)
data(covariates)
data(annotations)

# Create the 'FAMTdata'
##############################################
chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)
# 'FAMTdata' summary
summaryFAMT(chicken)

# Calculation of classical p-values
##############################################
# test on the 6th covariate:
rawpval = raw.pvalues(chicken,x=6)
hist(rawpval$pval)

# with a supplementary covariate (third column of the covariates data frame)
## Not run: rawpval = raw.pvalues(chicken,x=c(3,6),test=6)
## Not run: hist(rawpval$pval)
```

---

residualsFAMT	*Calculation of residual under null hypothesis*

Description

internal function

Description

The function produces summaries of 'FAMTdata' or 'FAMTmodel'. The function involves a specific method depending on the class of the main argument.

If the main argument is a 'FAMTdata' object, the function provides, for the 'expression file', the number of tests (which corresponds to the number of genes or rows), the sample size (which is the number of arrays or columns). The function provides classical summaries for 'covariates' and 'annotations' data (see summary in `FAMT-package`).

If the argument is a 'FAMTmodel', the function provides the numbers of rejected genes using classical and FAMT analyses, the annotation characteristics of significant genes, and the estimated proportion of true null hypotheses.

Usage

```
summaryFAMT(obj, pi0 = NULL, alpha = 0.15, info = c("ID", "Name"))
```

Arguments

obj	'FAMTdata' or 'FAMTmodel', see also `as.FAMTdata`, `modelFAMT`
pi0	Proportion of tests under H0. NULL, by default, it is estimated.
alpha	Type I levels for the control of the false discovery rate (0.15 by default) if the first argument is 'FAMTmodel' (it can be a single value or a vector).
info	Names of the columns containing the genes identification and array names in the original data frames, necessary if the first argument is 'FAMTmodel'

Value

If the argument is a 'FAMTdata': a list with components expression:

expression$'Number of tests'
> Number of genes

expression$'Sample size'
> Number of arrays

covariates	Classical summary of covariates
annotations	Classical summary of annotations

If the argument is a 'FAMTmodel':

nbreject	Matrix giving the numbers of rejected genes with the classical analysis and with the FAMT analysis for the given Type I levels alpha.
DE	Identification of the significant genes by their annotations.
pi0	Estimation of the proportion of true null hypotheses, estimated with the "smoother" method, see `pi0FAMT`.

Author(s)

David Causeur

See Also

as.FAMTdata, modelFAMT

Examples

```
## Reading 'FAMTdata'
data(expression)
data(covariates)
data(annotations)
chicken = as.FAMTdata(expression,covariates,annotations,idcovar=2)

## Summary of a 'FAMTdata'
############################################
summaryFAMT(chicken)

## Summary of a 'FAMTmodel'
############################################
# FAMT complete multiple testing procedure
model = modelFAMT(chicken,x=c(3,6),test=6,nbf=3)
summaryFAMT(model)
```

Index