Package 'factDesign'

November 12, 2026

Title Factorial designed microarray experiment analysis

Version 1.88.0

Author Denise Scholtens

Description This package provides a set of tools for analyzing data from a factorial designed microarray experiment, or any microarray experiment for which a linear model is appropriate. The functions can be used to evaluate tests of contrast of biological interest and perform single outlier detection.

Depends Biobase (>= 2.5.5)

Imports stats

Suggests affy, genefilter, multtest

Maintainer Denise Scholtens <dscholtens@northwestern.edu>

License LGPL

biocViews Microarray, DifferentialExpression

git_url https://git.bioconductor.org/packages/factDesign

git_branch RELEASE_3_23

git_last_commit 8221d18

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Contents

contrasts 2
estrogen 3
findFC 4
kRepsOverA 5
outliers 6

Index **8**

contrasts *Construct appropriate lambda matrix and test linear contrasts of the parameter estimates from a linear model.*

Description

'par2lambda' takes list of lm coefficient names and a corresponding list of numeric vectors corresponding to hypothesis tests of linear contrasts and returns a lambda matrix suitable for an F-test of the linear contrasts. 'par2lambda' is intended to be used in conjunction with 'contrastTest' or 'findFC'.

'contrastTest' performs an F test for simultaneous tests of linear contrasts using an appropriately specified lambda matrix and an lm object.

Usage

```
par2lambda(betaNames, betas, coefs)
contrastTest(model, lambda, cVec = NA, p = 0.01)
```

Arguments

betaNames	A character vector of the names of the coefficients in a linear model.
betas	A list of vectors of the parameters to be used in the contrasts.
coefs	A list of vectors of numeric coefficients corresponding to betas.
model	An lm object.
lambda	A matrix of coefficients in the appropriate order to be multiplied by the estimated coefficients of the lm object, possibly returned from 'par2lambda'.
cVec	A vector of constants for testing that the linear contrasts equal something other than zero. If this is unspecified, it is assumed to be zero.
p	The significance level at which to perform the contrast test.

Value

For par2lambda: A lambda matrix constructed for testing linear contrasts using lm output.

For contrastTest:

test	Returns "REJECT" or "FAIL TO REJECT" based on the result of the test of hypothesis.
Fstat	The F statistic for the test of contrast.
pvalue	The corresponding pvalue for the F test.
cEst	The contrast estimate.

Author(s)

Denise Scholtens

See Also

lm

Examples

```
data(estrogen)
ES <- pData(estrogen)[["ES"]]
TIME <- pData(estrogen)[["TIME"]]
fit <- lm(exprs(estrogen)["40079_at",] ~ ES + TIME + ES*TIME)
betaNames <- names(coef(fit))
betas <- list(c("ESP"),c("ESP","ESP:TIME48h"))
coefs <- list(c(1),c(1,1))
lambda <- par2lambda(betaNames,betas,coefs)
contrastTest(fit,lambda)
```

---

estrogen *Microarray Data from an Experiment on Breast Cancer Cells*

Description

Gene expression levels for 500 genes from a 2x2 factorial experiment on MCF7 breast cancer cells using Affymetrix HGU95av2 arrays.

Usage

```
data(estrogen)
```

Format

An ExpressionSet object with 500 genes, 8 samples, and 2 variables.

Details

The factors in this experiment were estrogen (ES: P or A) and length of exposure (TIME: 10 or 48 hours). Gene expression values were estimated using rma after quantile normalization (see the 'affy' package). Expression estimates are reported log base 2, as suggested by the rma method.

Source

Scholtens et al. Analyzing Factorial Designed Microarray Experiments. Journal of Multivariate Analysis. 2004;90(1):19-43.

Examples

```
data(estrogen)
pData(estrogen)
exprs(estrogen)[1,]
```

`findFC` *A function to find the fold change between two experimental conditions in a factorial experiment based on the linear model parameter estimates.*

Description

'findFC' constructs a point estimate of fold change using the linear model coefficients in an lm object.

Usage

```
findFC(model, lambdaNum, lambdaDenom, logbase=NULL)
```

Arguments

`model`	An lm object.
`lambdaNum`	A numeric vector of coefficients for the parameters to be used in the numerator of the fold change estimate.
`lambdaDenom`	A numeric vector of coefficients for the parameters to be used in the denominator of the fold change estimate.
`logbase`	By default, set to NULL. For log-transformed data, the base of the logarithm. Specify "exp" for natural log-transformed data.

Details

`logbase`=NULL if the data have not been log-transformed. The fold change estimate is calculated as the ratio for the parameter estimates corresponding to the experimental conditions of interest.

`logbase`="exp" if the data have been natural log-transformed. The fold change is calculated as the difference in the parameter estimates for the two conditions of interest, then exponentiated using exp().

`logbase` can be set to any number, for example 2, for other log transforms. The fold change is calculated as the difference in the parameter estimates for the two conditions of interest, then exponentiated with `logbase` as the base.

Value

A point estimate of the fold change between the experimental conditions specified in the lambdaNum and lambdaDenom vectors.

Author(s)

Denise Scholtens

See Also

`par2lambda`

Examples

```
data(estrogen)
ES <- pData(estrogen)[["ES"]]
TIME <- pData(estrogen)[["TIME"]]
fit <- lm(exprs(estrogen)["33744_at",] ~ ES + TIME + ES*TIME)
betaNames <- names(coef(fit))
betas <- list(c("(Intercept)","ESP","TIME48h","ESP:TIME48h"),
     c("(Intercept)","ESP"))
coefs <- list(c(1,1,1,1),c(1,1))
lambda <- par2lambda(betaNames,betas,coefs)
findFC(fit,lambda[1,],lambda[2,],logbase=2)
```

---

kRepsOverA — *A filter function for at least k sets of replicates in a factorial experiment to have mean larger than A.*

Description

'kRepsOverA' returns a filter function with bindings for 'k' and 'A'. This function evalutes 'TRUE' is at least 'k' of the means of the replicates are larger than 'A'.

Usage

```
kRepsOverA(k, A = 100, INDEX)
```

Arguments

k	The number of sets of replicates with mean greater than A.
A	The value to exceed.
INDEX	List of factors, each of the same length as the input vector.

Value

A function with bindings for 'A', 'k', and 'INDEX'.

Author(s)

Denise Scholtens

See Also

kOverA,pOverA

Examples

```
library(affy)
library(genefilter)
data(estrogen)
#select the replicates with values larger than 5
f1 <- kRepsOverA(1,5,INDEX=pData(estrogen))
genefilter(estrogen[1:30],f1)
```

`outliers` *Detect single outliers in experimental designs with only two replicates per treatment condition.*

Description

These function detect pairs of observations with unexpectedly large differences compared to the rest of the data and determine if one of the pair is a single outlier using median absolute deviation criteria.

Usage

```
outlierPair(x, INDEX, p = 0.05, na.rm = TRUE)
madOutPair(x, whichPair, c = 4)
```

Arguments

`x`	A vector of observations.
`INDEX`	A list of factors, each the same length as x, used to indicate the replicate observations.
`p`	The significance level at which to perform the test.
`na.rm`	If TRUE, will remove missing values.
`whichPair`	A result of outlierPair, recording which pair has largest difference between replicate observations.
`c`	The number of median absolute deviations to be used as a cutoff for determining single outliers.

Details

This outlier detection method is useful for small factorial designs in which the usual residuals from a linear model would have a large number of linear dependencies compared to the actual number of residuals. The function first calculates n difference between 2n replicates (call these pure residuals), and then constructs an F-statistic: f=(large squared p.r.)/((sum of remaining squared p.r.'s)/(n-1)). An p-value (adjusted for taking the largest of the p.r.'s) is calculated by n*Pr(F(1,n-1)>f). If f>=n-1, this p-value is exact, otherwise it is an upper bound.

Once pairs with significantly large differences are identified using outlierPair, madOutPair is applied. If only one of the tagged replicates falls outside the range of (med(x)-c*mad(x),med(x)+c*mad(x)), the observation is designated the single outlier.

Value

For `outlierPair`:

`test`	Returns TRUE if an outlier pair is detected at the specified level of significance p.
`pval`	The actual value of n*Pr(F(1,n-1)>f).
`whichPair`	The index of the pair of observations with the largest difference.

For `madOutPair`:

The index of the single outlier observation, or "NA" if no single outliers are detected.

Author(s)

Denise Scholtens

References

Scholtens et al. Analyzing Factorial Designed Microarray Experiments. Journal of Multivariate Analysis. 2004;90(1):19-43.

See Also

`madOutPair`

Examples

```
data(estrogen)

op1 <- outlierPair(exprs(estrogen)["728_at",],INDEX=pData(estrogen),p=.05)
print(op1)
madOutPair(exprs(estrogen)["728_at",],op1[[3]])

op2 <- outlierPair(exprs(estrogen)["33379_at",],INDEX=pData(estrogen),p=.05)
print(op2)
madOutPair(exprs(estrogen)["33379_at",],op2[[3]])
```

Index

∗ **datasets**
    estrogen, 3
∗ **design**
    contrasts, 2
    findFC, 4
    kRepsOverA, 5
    outliers, 6
∗ **htest**
    contrasts, 2
∗ **models**
    contrasts, 2
    findFC, 4
    outliers, 6
∗ **univar**
    kRepsOverA, 5

contrasts, 2
contrastTest *(*contrasts*)*, 2

estrogen, 3

findFC, 4

kOverA, *5*
kRepsOverA, 5

lm, *3*

madOutPair, *7*
madOutPair *(*outliers*)*, 6

outlierPair *(*outliers*)*, 6
outliers, 6

par2lambda, *4*
par2lambda *(*contrasts*)*, 2
pOverA, *5*