Package 'permuco'

August 25, 2024

Type Package

Title Permutation Tests for Regression, (Repeated Measures) ANOVA/ANCOVA and Comparison of Signals

Version 1.1.3

Date 2024-08-25

Maintainer Jaromil Frossard <jaromil.frossard@gmail.com>

URL https://github.com/jaromilfrossard/permuco

BugReports https://github.com/jaromilfrossard/permuco/issues

Description Functions to compute p-values based on permutation tests. Regression, ANOVA and ANCOVA, omnibus F-tests, marginal unilateral and bilateral t-tests are available. Several methods to handle nuisance variables are implemented (Kherad-Pajouh, S., & Renaud, O. (2010) <doi:10.1016/j.csda.2010.02.015> ; Kherad-Pajouh, S., & Renaud, O. (2014) <doi:10.1007/s00362-014-0617-3> ; Winkler, A. M., Ridgway, G. R., Webster, M. A., Smith, S. M., & Nichols, T. E. (2014) <doi:10.1016/j.neuroimage.2014.01.060>). An extension for the comparison of signals issued from experimental conditions (e.g. EEG/ERP signals) is provided. Several corrections for multiple testing are possible, including the cluster-mass statistic (Maris, E., & Oostenveld, R. (2007) <doi:10.1016/j.jneumeth.2007.03.024>) and the threshold-free cluster enhancement (Smith, S. M., & Nichols, T. E. (2009) <doi:10.1016/j.neuroimage.2008.03.061>).

License GPL (>= 2)

Imports permute, Matrix, stats, graphics, Rcpp

LazyData TRUE

Encoding UTF-8

Suggests R.rsp, testthat, covr, knitr, rmarkdown

VignetteBuilder R.rsp, knitr

LinkingTo Rcpp

RoxygenNote 7.2.3

NeedsCompilation yes

Author Jaromil Frossard [aut, cre],
Olivier Renaud [aut]

Contents

---

aovperm	*P-values based on permutation tests for ANOVA and repeated measures ANOVA designs.*

Description

Provides p-values for omnibus tests based on permutations for factorial and repeated measures ANOVA. This function produces the F statistics, parametric p-values (based, on Gaussian and sphericity assumptions) and p-values based on the permutation methods that handle nuisance variables.

Usage

```
aovperm(
  formula,
  data = NULL,
  np = 5000,
  method = NULL,
  type = "permutation",
  ...
)
```

Arguments

formula	A formula object. The formula for repeated measures ANOVA should be written using the same notation as aov by adding +Error(id/within), where id is the factor that identify the subjects and within is the within factors.
data	A data frame or matrix.
np	The number of permutations. Default value is 5000.
method	A character string indicating the method used to handle nuisance variables. Default is NULL and will change if to "freedman_lane" for the fixed effects model and "Rd_kheradPajouh_renaud" for the random effect models. See Details for other methods.
type	A character string to specify the type of transformations: "permutation" and "signflip" are available. Is overridden if P is given. See help from Pmat.
...	Futher arguments, see details.

Details

The following methods are available for the fixed effects model defined as $y = D\eta + X\beta + \epsilon$. If we want to test $\beta = 0$ and take into account the effects of the nuisance variables D, we transform the data :

method argument	$y*$	$D*$	$X*$
"draper_stoneman"	y	D	PX
"freedman_lane"	$(H_D + PR_D)y$	D	X
"manly"	Py	D	X
"terBraak"	$(H_{X,D} + PR_{X,D})y$	D	X
"kennedy"	$PR_D y$		$R_D X$
"huh_jhun"	$PV'R_D y$		$V'R_D X$
"dekker"	y	D	$PR_D X$

The following methods are available for the random effects model $y = D\eta + X\beta + E\kappa + Z\gamma + \epsilon$. If we want to test $\beta = 0$ and take into account the effect of the nuisance variable D we can transform the data by permutation:

method argument	$y*$	$D*$	$X*$	$E*$	$Z*$

| "Rd_kheradPajouh_renaud" | $PR_D y$ | $R_D X$ | $R_D E$ | $R_D Z$ |
| "Rde_kheradPajouh_renaud" | $PR_{D,E} y$ | $R_{D,E} X$ | | $R_{D,E} Z$ |

Other arguments could be pass in . . . :

P : a matrix, of class matrix or Pmat, containing the permutations (for the reproductibility of the results). The first column must be the identity permutation (not checked). P overwrites np argument. rnd_rotation : a random matrix of size $n \times n$ to compute the rotation used for the "huh_jhun" method. coding_sum : a logical set to TRUE defining the coding of the design matrix to contr.sum to test the main effects. If it is set to FALSE the design matrix is computed with the coding defined in the dataframe. The tests of simple effets are possible with a coding of the factors of the dataframe set to contr.treatment.

Value

A lmperm object containing most of the objects given in an [lm](#) object, an ANOVA table with parametric and permutation p-values, the test statistics and the permutation distributions.

Author(s)

jaromil.frossard@unige.ch

See Also

[lmperm](#) [plot.lmperm](#)

Other main function: [clusterlm](#)(), [lmperm](#)()

Examples

```
## data
data("emergencycost")

## centrering the covariate to the mean
emergencycost$LOSc <- scale(emergencycost$LOS, scale = FALSE)

## ANCOVA
## Warning : np argument must be greater (recommendation: np>=5000)
mod_cost_0 <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost, np = 2000)
mod_cost_0

## same analysis but with signflip
## Warning : np argument must be greater (recommendation: np>=5000)
mod_cost_0s <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost, type="signflip", np = 2000)
mod_cost_0s

## Testing at 14 days
emergencycost$LOS14 <- emergencycost$LOS - 14
```

```
mod_cost_14 <- aovperm(cost ~ LOS14*sex*insurance, data = emergencycost, np = 2000)
mod_cost_14

## Effect of sex within the public insured
contrasts(emergencycost$insurance) <- contr.treatment
contrasts(emergencycost$sex) <- contr.sum
emergencycost$insurance <- relevel(emergencycost$insurance, ref = "public")

mod_cost_se <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost,
                        np = 2000, coding_sum = FALSE)
mod_cost_se


## Repeated measures ANCOVA
## data
data(jpah2016)

## centrering the covariate
jpah2016$bmic <- scale(jpah2016$bmi, scale = FALSE)

## Warning : np argument must be greater (recommendation: np>=5000)
mod_jpah2016 <- aovperm(iapa ~ bmic*condition*time+ Error(id/(time)),
                    data = jpah2016, method = "Rd_kheradPajouh_renaud")
mod_jpah2016
```

---

as.Pmat   *Method to convert into* Pmat *object.*

Description

Convert a matrix into a Pmat object.

Usage

```
as.Pmat(x)
```

Arguments

x   a matrix.

See Also

Other pmat: Pmat_product(), Pmat()

---

attentionshifting_design

Datatset of the design for the data attentionshifting_signal

Description

Design of an experiment measuring the EEG brain activity of 15 participants who have been shown images of neutral and angry faces. Those faces were shown at a different visibility 16ms and 166ms and were displayed either to the left or to the right of a screen. The laterality, sex, age, and 2 measures of anxiety for each subjects are also available. The amplitude of the EEG recording are located in the dataset attentionshifting_signal.

- id : identifier of the subject.

- visibility : time of exposure to the image (16ms: subliminal or 166ms:supraliminal).

- emotion : type of emotion of the image (angry or neutral).

- direction : position of image one the screen (left or right).

- laterality_id : measure of laterality of the subject.

- age : age of the subject.

- sex : sex of the subject.

- STAIS_state : measure of the state of anxiety of the subject.

- STAIS_trait : measure of the personality trait of anxiety of the subject.

Usage

```
data(attentionshifting_design)
```

Format

A data frame with 120 rows and 10 variables.

See Also

Other dataset: `attentionshifting_signal`, `emergencycost`, `jpah2016`

---

attentionshifting_signal

Dataset containing the event-related potential of the electrod O1 of a control experiment.

Description

The ERP of the electrod O1 of an experiment in attention shifting. This dataset contains the amplitude of the signals sampled at 1024 Hz. The design of the experiment is given in the dataset attentionshifting_design.

Usage

```
data(attentionshifting_signal)
```

Format

A data frame with 120 rows and 819 variables.

Details

- ERP (in muV) of the electrod O1 measured from -200 to 600 timeframes before and after the onset of the stimulus.

See Also

Other dataset: attentionshifting_design, emergencycost, jpah2016

---

clusterlm *Cluster-mass test for longitudinal data*

Description

Compute the cluster-mass test for longitudinal linear model.

Usage

```
clusterlm(
  formula,
  data = NULL,
  np = 5000,
  method = NULL,
  type = "permutation",
  test = "fisher",
  threshold = NULL,
  aggr_FUN = NULL,
```

```
    multcomp = "clustermass",
    ...
)
```

Arguments

formula   A formula object where the left part is a matrix defined in the global environ-
                 ment.

data   A data frame for the independent variables.

np   The number of permutations. Default value is 5000.

method   A character string indicating the method used to handle nuisance variables. De-
                 fault is NULL and will switch to "freedman_lane" for the fixed effects model
                 and to "Rd_kheradPajouh_renaud" for the repeated measures ANOVA. See
                 lmperm or aovperm for details on the permutation methods.

type   A character string to specify the type of transformations: "permutation" and
                 "signflip" are available. Is overridden if P is given. See help from Pmat.

test   A character string to specify the name of the test. Default is "fisher". "t" is
                 available for the fixed effects model.

threshold   A numerical value that specify the threshold for the "clustermass" multiple
                 comparisons procedure. If it is a vector each value will be associated to an
                 effect. If it is scalar the same threshold will be used for each test. Default value
                 is NULL and will compute a threshold based on the 0.95 quantile of the choosen
                 test statistic.

aggr_FUN   A function used as mass function. It should aggregate the statistics of a cluster
                 into one scalar. Default is the sum of squares fot t statistic and sum for F statistic.

multcomp   A vector of character defining the methods of multiple comparisons to com-
                 pute. Default is "clustermass", and the additional options are available :
                 "tfce","bonferroni", "holm", "troendle", "minP" and "benjamini_hochberg".

...   Futher arguments, see details.

Details

The random effects model is only available with a F statistic.

Other arguments could be pass in ... :

P : A matrix containing the permutation of class matrix or Pmat; which is used for the repro-
ducibility of the results. The first column must be the identity. P overwrites np argument.

rnd_rotation : A matrix of random value to compute a rotation of size $n \times n$ that will be used for
the "huh_jhun" method.

p_scale = FALSE : if set to TRUE, the several multiple comparisons procedures are compute on the
$1 - p$ scale, where p is the p-value. The threshold have to be set between 0 and 1 (eg: threshold =
0.95). The function aggr_FUN should be big when there is evidence against the null (eg: aggr_FUN

= function(p)sum(abs(log(1-p))). Moreover under the probability scale the cluster mass statistics is sensitive to the number permutations.

H, E, ndh : the parameters used for the "tfce" method. Default values are set to H = 2 for the height parameter, to E = 0.5 for the extend parameter and to ndh = 500 for the number terms to approximate the integral.

alpha = 0.05 : the type I error rate. Used for the troendle multiple comparisons procedure.

return_distribution = FALSE : return the permutation distribution of the statistics. Warnings : return one high dimensional matrices (number of test times number of permutation) for each test. coding_sum : a logical defining the coding of the design matrix to contr.sum: set by default to TRUE for ANOVA (when the argument test is "fisher") to tests main effects and is set to FALSE when test is "t". If coding_sum is set to FALSE the design matrix is computed with the coding defined in the dataframe and the tests of simple effects are possible with a coding of the dataframe set to contr.treatment.

Value

A clusterlm object. Use the plot.clusterlm or summary.clusterlm method to see results of the tests.

Author(s)

jaromil.frossard@unige.ch

References

Maris, E., & Oostenveld, R. (2007). Nonparametric statistical testing of EEG-and MEG-data. Journal of neuroscience methods, 164(1), 177-190.

Smith, S. M., & Nichols, T. E. (2009). Threshold-free cluster enhancement: addressing problems of smoothing, threshold dependence and localisation in cluster inference. Neuroimage, 44(1), 83-98.

See Also

plot.clusterlm summary.clusterlm

Other main function: aovperm(), lmperm()

Examples

```
## Cluster-mass for repeated measures ANOVA
## Warning : np argument must be greater (recommendation: np >= 5000)
electrod_O1 <- clusterlm(attentionshifting_signal ~ visibility*emotion*direction
        + Error(id/(visibility*emotion*direction)), data = attentionshifting_design,
        np = 50)

## Results
plot(electrod_O1)

## Results with labels on the x axis that represent seconds from time-locked event:
```

```
plot(electrod_O1, nbbaselinepts = 200, nbptsperunit = 1024)

## Tables of clusters
electrod_O1

## Not run:
## Change the function of the aggregation

## Sum of squares of F statistics
electrod_O1_sum <- clusterlm(attentionshifting_signal ~ visibility*emotion*direction
        + Error(id/(visibility*emotion*direction)), data = attentionshifting_design,
        aggr_FUN = function(x)sum(x^2))

## Length of the cluster
electrod_O1_length <- clusterlm(attentionshifting_signal ~ visibility*emotion*direction
        + Error(id/(visibility*emotion*direction)), data = attentionshifting_design,
        aggr_FUN = function(x)length(x))


## All multiple comparisons procedures for repeated measures ANOVA
## Permutation method "Rde_kheradPajouh_renaud"
full_electrod_O1 <- clusterlm(attentionshifting_signal ~ visibility*emotion*direction
        + Error(id/(visibility*emotion*direction)), data = attentionshifting_design,
        method = "Rde_kheradPajouh_renaud", multcomp = c("troendle", "tfce",
        "clustermass", "bonferroni", "holm", "benjamini_hochberg"))

## End(Not run)
```

---

compute_clusterdepth   *Cluster-depth correction*

Description

Compute the clusterdepth test correction given a matrix a permuted statistical signals.

Usage

```
compute_clusterdepth(distribution, threshold, alternative = "two.sided")
```

Arguments

distribution	A matrix of permuted statistical signal. The first row indicating the observed statistics.
threshold	A scalar that represents the threshold to create the clusters.
alternative	A character string indicating the alternative hypothesis. Default is "greater". Choose between "greater", "less" or "two.sided".

See Also

Other multcomp: [compute_clusterdepth_head](), [compute_clustermass](), [compute_maxT](),
[compute_minP](), [compute_stepdownmaxT](), [compute_tfce](), [compute_troendle]()

---

compute_clusterdepth_head

Cluster-depth correction (from the head only)

Description

Compute the cluster-depth test correction (from the head) given a matrix a permuted statistical
signals.

Usage

```
compute_clusterdepth_head(distribution, threshold, alternative = "two.sided")
```

Arguments

distribution	A matrix of permuted statistical signal. The first row indicating the observed statistics.
threshold	A scalar that represents the threshold to create the clusters.
alternative	A character string indicating the alternative hypothesis. Default is "greater". Choose between "greater", "less" or "two.sided".

See Also

Other multcomp: [compute_clusterdepth](), [compute_clustermass](), [compute_maxT](), [compute_minP](),
[compute_stepdownmaxT](), [compute_tfce](), [compute_troendle]()

---

compute_clustermass   *Clustermass test correction*

Description

Compute the clustermass test correction given a matrix a permuted statistical signals.

Usage

```
compute_clustermass(distribution, threshold, aggr_FUN, alternative = "greater")
```

Arguments

`distribution`	A matrix of permuted statistical signal. The first row indicating the observed statistics.
`threshold`	A scalar that respresents the threshold to create the clusters.
`aggr_FUN`	A function to compute the clustermasses. See details for examples.
`alternative`	A character string indicating the alternative hypothesis. Default is `"greater"`. Choose between `"greater"`, `"less"` or `"two.sided"`.

Details

The `aggr_FUN` argument may take predefined function as the sum: `aggr_FUN = sum` and also user-defined function as the sum of squares: `aggr_FUN = function(x){sum(x^2)}`

See Also

Other multcomp: `compute_clusterdepth_head()`, `compute_clusterdepth()`, `compute_maxT()`, `compute_minP()`, `compute_stepdownmaxT()`, `compute_tfce()`, `compute_troendle()`

`compute_maxT`	*The max-T correction*

Description

Compute the max-T correction given a matrix a permuted statistics.

Usage

```
compute_maxT(distribution, alternative)
```

Arguments

`distribution`	A matrix of permuted statistical signal. The first row indicating the observed statistics.
`alternative`	A character string indicating the alternative hypothesis. Default is `"greater"`. Choose between `"greater"`, `"less"` or `"two.sided"`.

See Also

Other multcomp: `compute_clusterdepth_head()`, `compute_clusterdepth()`, `compute_clustermass()`, `compute_minP()`, `compute_stepdownmaxT()`, `compute_tfce()`, `compute_troendle()`

---

compute_minP *The min-P correction*

Description

Compute the min-P correction given a matrix a permuted statistics.

Usage

```
compute_minP(distribution, alternative)
```

Arguments

distribution	A matrix of permuted statistical signal. The first row indicating the observed statistics.
alternative	A character string indicating the alternative hypothesis. Default is "greater". Choose between "greater", "less" or "two.sided".

See Also

Other multcomp: compute_clusterdepth_head(), compute_clusterdepth(), compute_clustermass(), compute_maxT(), compute_stepdownmaxT(), compute_tfce(), compute_troendle()

---

compute_stepdownmaxT *Step-down version of the max-T correction*

Description

Compute the Step-down version of the max-T given a matrix a permuted statistics.

Usage

```
compute_stepdownmaxT(distribution, alternative)
```

Arguments

distribution	A matrix of permuted statistical signal. The first row indicating the observed statistics.
alternative	A character string indicating the alternative hypothesis. Default is "greater". Choose between "greater", "less" or "two.sided".

See Also

Other multcomp: compute_clusterdepth_head(), compute_clusterdepth(), compute_clustermass(), compute_maxT(), compute_minP(), compute_tfce(), compute_troendle()

---

compute_tfce   *Threshold-Free Cluster-Enhancement correction*

Description

Compute the TFCE correction given a matrix a permuted statistical signals.

Usage

```
compute_tfce(distribution, alternative = "greater", E = 0.5, H = 1, ndh)
```

Arguments

distribution	A matrix of permuted statistical signal. The first row indicating the observed statistics.
alternative	A character string indicating the alternative hypothesis. Default is "greater". Choose between "greater", "less" or "two.sided".
E	A scalar that represent the extend parameter of the TFCE transformation. Default is E = 0.5.
H	A scalar that represent the height parameter of the TFCE transformation. Default is H = 1.
ndh	The number of terms in the approximation of the integral.

Details

The approximation of the TFCE integral starts from the minimal statistic (among the permuted statistic), not 0. Indeed, the value of the integral (from 0, to this minimal value) is the same for all time points, all permutations and does not need to be computed to get p-values.

See Also

Other multcomp: [compute_clusterdepth_head](), [compute_clusterdepth](), [compute_clustermass](), [compute_maxT](), [compute_minP](), [compute_stepdownmaxT](), [compute_troendle]()

---

compute_troendle   *The Troendle's correction*

Description

Compute the Troendle's correction given a matrix a permuted statistics.

Usage

```
compute_troendle(distribution, alternative)
```

Arguments

distribution   A matrix of permuted statistical signal. The first row indicating the observed statistics.

alternative   A character string indicating the alternative hypothesis. Default is ʺgreaterʺ. Choose between ʺgreaterʺ, ʺlessʺ or ʺtwo.sidedʺ.

See Also

Other multcomp: `compute_clusterdepth_head()`, `compute_clusterdepth()`, `compute_clustermass()`, `compute_maxT()`, `compute_minP()`, `compute_stepdownmaxT()`, `compute_tfce()`

---

emergencycost   *Dataset of cost of emergency patients.*

Description

Observational data from 176 emergency patients with variables :

Usage

```
data(emergencycost)
```

Format

A data frame with 176 rows and 5 variables.

Details

- sex.
- age.
- insurance : the type of insurance, private or semi private (`semi_private`) or public (`public`).
- LOS : the length of the stay in days.
- cost : the cost in CHF.

References

Heritier, S., Cantoni, E., Copt, S., & Victoria-Feser, M. P. (2009). Robust methods in Biostatistics (Vol. 825). John Wiley & Sons.

See Also

Other dataset: `attentionshifting_design`, `attentionshifting_signal`, `jpah2016`

---

jpah2016   *Dataset of a control study in psychology.*

Description

A subset of a control experiment measuring the impulsive approach tendencies toward physical activity or sedentary behaviors.

Usage

```
data(jpah2016)
```

Format

A data frame with 38 rows and 8 variables.

Details

- id identifier of the subject.

- bmi body mass index.

- age.

- sex.

- condition the experimental condition where the task was to approach physical activity and avoid sedentary behavior (ApSB_AvPA), approach sedentarity behavior and avoid physical activity (ApPA_AvSB), and a control condition (control).

- time pre, post.

- iapa measure of impulsive approach tendencies toward physical activity (dependant variable).

- iasb measure of impulsive approach tendencies toward sedentary behavior (dependant variable).

References

Cheval, B., Sarrazin, P., Pelletier, L., & Friese, M. (2016). Effect of retraining approach-avoidance tendencies on an exercise task: A randomized controlled trial. Journal of Physical Activity and Health, 13(12), 1396-1403.

See Also

Other dataset: attentionshifting_design, attentionshifting_signal, emergencycost

---

lmperm	*Permutation tests for regression parameters*

Description

Compute permutation marginal tests for linear models. This function produces t statistics with univariate and bivariate p-values. It gives the choice between multiple methods to handle nuisance variables.

Usage

```
lmperm(
  formula,
  data = NULL,
  np = 5000,
  method = NULL,
  type = "permutation",
  ...
)
```

Arguments

formula	A formula object.
data	A data frame or matrix.
np	The number of permutations. Default value is 5000.
method	A character string indicating the method use to handle nuisance variables. Default is `"freedman_lane"`. For the other methods, see details.
type	A character string to specify the type of transformations: "permutation" and "signflip" are available. Is overridden if P is given. See help from Pmat.
...	Futher arguments, see details.

Details

The following methods are available for the fixed effects model defined as $y = D\eta + X\beta + \epsilon$. If we want to test $\beta = 0$ and take into account the effects of the nuisance variables D, we transform the data :

method argument	$y*$	$D*$	$X*$
`"draper_stoneman"`	y	D	PX
`"freedman_lane"`	$(H_D + PR_D)y$	D	X
`"manly"`	Py	D	X
`"terBraak"`	$(H_{X,D} + PR_{X,D})y$	D	X
`"kennedy"`	$PR_D y$		$R_D X$
`"huh_jhun"`	$PV'R_D y$		$V'R_D X$
`"dekker"`	y	D	$PR_D X$

Other arguments could be pass in ... :

P : a matrix containing the permutations of class matrix or Pmat for the reproductibility of the results. The first column must be the identity. P overwrites np argument.

rnd_rotation : a random matrix of size $n \times n$ to compute the rotation used for the "huh_jhun" method.

Value

A lmperm object. See aovperm.

Author(s)

jaromil.frossard@unige.ch

References

Kherad-Pajouh, S., & Renaud, O. (2010). An exact permutation method for testing any effect in balanced and unbalanced fixed effect ANOVA. Computational Statistics & Data Analysis, 54(7), 1881-1893.

Kherad-Pajouh, S., & Renaud, O. (2015). A general permutation approach for analyzing repeated measures ANOVA and mixed-model designs. Statistical Papers, 56(4), 947-967.

Winkler, A. M., Ridgway, G. R., Webster, M. A., Smith, S. M., & Nichols, T. E. (2014). Permutation inference for the general linear model. Neuroimage, 92, 381-397.

See Also

aovperm plot.lmperm

Other main function: aovperm(), clusterlm()

Examples

```
## data
data("emergencycost")

## Testing at 14 days
emergencycost$LOS14 <- emergencycost$LOS - 14

## Univariate t test
contrasts(emergencycost$insurance) <- contr.sum
contrasts(emergencycost$sex) <- contr.sum

## Warning : np argument must be greater (recommendation: np>=5000)
modlm_cost_14 <- lmperm(cost ~ LOS14*sex*insurance, data = emergencycost, np = 2000)
modlm_cost_14
```

plot.clusterlm	*Plot cluster or parameters.*

Description

Plot method for class `clusterlm`.

Usage

```
## S3 method for class 'clusterlm'
plot(
  x,
  effect = "all",
  type = "statistic",
  multcomp = x$multcomp[1],
  alternative = "two.sided",
  enhanced_stat = FALSE,
  nbbaselinepts = 0,
  nbptsperunit = 1,
  distinctDVs = NULL,
  ...
)
```

Arguments

x	A `clusterlm` object.
effect	A vector of character naming the effects to display. Default is `"all"`.
type	A character string that specified the values to highlight. `"statistic"` or `"coef"` are available. Default is `"statistic"`.
multcomp	A character string specifying the method use to correct the p-value. It should match the one computed in the object. Default is the (first) method in the call to clusterlm. See clusterlm.
alternative	A character string specifying the alternative hypothesis for the t-test. The available options are `"greater"`, `"less"` and `"two.sided"`. Default is `"two.sided"`.
enhanced_stat	A logical. Default is FALSE. If TRUE, the enhanced statistic will be plotted overwise it will plot the observed statistic. Change for the `"tfce"` or the `"clustermass"` multiple comparisons procedures.
nbbaselinepts	An integer. Default is 0. If the origin of the x axis should be shifted to show the start of the time lock, provide the number of baseline time points.
nbptsperunit	An integer. Default is 1. Modify this value to change the scale of the label from the number of points to the desired unit. If points are e.g. sampled at 1024Hz, set to 1024 to scale into seconds and to 1.024 to scale into milliseconds.
distinctDVs	Boolean. Should the DVs be plotted distictively, i.e. should the points be unlinked and should the name of the DVs be printed on the x axis ? Default is FALSE if the number of DV is large thant 15 or if the method is "clustermass" or "tfce".

... further argument pass to plot.

See Also

Other plot: `plot.lmperm()`

`plot.lmperm` *Plot method for class* `"lmperm"`.

Description

Show the density of statistics and the test statistic.

Usage

```
## S3 method for class 'lmperm'
plot(x, FUN = density, ...)
```

Arguments

x	A "lmperm" object.
FUN	A function to compute the density. Default is density.
...	futher arguments pass to plot.

Details

Other argument can be pass to the function :

`effect` : a vector of character string indicating the name of the effect to plot.

See Also

Other plot: `plot.clusterlm()`

Pmat   *Create a set of permutations/signflips.*

Description

Compute a permutation matrix used as argument in aovperm, lmperm, clusterlm functions. The first column represents the identity permutation. Generally not suited for the "huh_jhun" method, as the dimension of this matrix does not correspond to the number of observations and may vary for different factors.

Usage

```
Pmat(np = 5000, n, type = "permutation", counting = "random")
```

Arguments

np	A numeric value for the number of permutations. Default is 5000.
n	A numeric value for the number of observations.
type	A character string to specify the type of transformations: "permutation" and "signflip" are available. See details.
counting	A character string to specify the selection of the transformations. "all" and "random" are available. See details.

Details

`couting` can set to :

`"random"` : np random with replacement permutations/signflips among the $n!/2^n$ permutations/signflips.

`"all"` : all $n!/2^n$ possible permutations/signflips.

Value

A matrix n x np containing the permutations/signflips. First permutation is the identity.

See Also

Other pmat: `Pmat_product()`, `as.Pmat()`

Examples

```
## data
data("emergencycost")

## Create a set of 2000 permutations
set.seed(42)
pmat = Pmat(np = 2000, n = nrow(emergencycost))
sfmat = Pmat(np = 2000, n = nrow(emergencycost), type = "signflip")
```

```
## centrering the covariate to the mean
emergencycost$LOSc <- scale(emergencycost$LOS, scale = FALSE)

## ANCOVA
mod_cost_0 <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost, np = 2000)
mod_cost_1 <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost, P = pmat)
mod_cost_2 <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost, P = pmat)
mod_cost_3 <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost, P = sfmat)
mod_cost_4 <- aovperm(cost ~ LOSc*sex*insurance, data = emergencycost,type="signflip")

## Same p-values for both models 1 and 2 but different of model 0
mod_cost_0
mod_cost_1
mod_cost_2
mod_cost_3
mod_cost_4
```

Pmat_product   *Multiplies a vector with a Pmat object*

Description

Multiplies a vector or matrix with a Pmat object

Usage

```
Pmat_product(x, P, type = NULL)
```

Arguments

x	A a vector or matrix
P	A Pmat object
type	A character string indicating the type of transformation. This argument need to be specified if P is not of class Pmat.

Value

A matrix n x np containing the permutated/signflipped vectors.

See Also

Other pmat: Pmat(), as.Pmat()

---

print.clusterlm *Print* clusterlm *object.*

Description

Display the corrected p-values for each effects. Results of the ″clustermass″ procedure.

Usage

```
## S3 method for class 'clusterlm'
print(x, multcomp = NULL, alternative = "two.sided", ...)
```

Arguments

x	A clusterlm object.
multcomp	A character string indicating the multiple comparison procedure to print. Default is NULL a print the first multiple comparisons procedure of the clusterlm object.
alternative	A character string indicating the alternative hypothesis. Choose between ″two.sided″, ″greater″, ″less″. Default is ″two.sided″.
...	Further arguments pass to print.

See Also

Other summary: summary.clusterlm()

---

summary.clusterlm *Summarize of a* clusterlm *object.*

Description

Display the corrected p-values for each effects.

Usage

```
## S3 method for class 'clusterlm'
summary(
  object,
  alternative = "two.sided",
  multcomp = NULL,
  table_type = NULL,
  ...
)
```

Arguments

`object`	A `clusterlm` object.
`alternative`	A character string indicating the alternative hypothesis. Choose between `"two.sided"`, `"greater"`, `"less"`. Default is `"two.sided"`.
`multcomp`	A character string indicating the multiple comparison procedure to display.
`table_type`	A character string indicating the type of table to display. Choose between `"cluster"`, which aggregates test into pseudo-clusters (see details for the interpretations) or `"full"` which displays the p-values for all tests. See details for default values.
`...`	Further arguments see details.

Details

It creates the full table when the number of tests is <=15 and creates a table of pseudo-clusters overwise. Note that for the `"troendle"` method is not based on clustering of the data and the table of pseudo-clusters should only be used to facilitate the reading of the results.

Value

A table for each effect indicating the statistics and p-values of the clusters.

See Also

Other summary: print.clusterlm()

Index