# Package ‘nparcomp’

October 13, 2022

**Type** Package  
**Title** Multiple Comparisons and Simultaneous Confidence Intervals  
**Version** 3.0  
**Date** 2019-06-24  
**Author** Frank Konietschke [aut, cre], Kimihiro Noguchi [ctr], Kerstin Rubarth  
**Maintainer** Kerstin Rubarth <kerstin.rubarth@charite.de>  
**Description** With this package, it is possible to compute nonparametric simultaneous confidence intervals for relative contrast effects in the unbalanced one way layout. Moreover, it computes simultaneous p-values. The simultaneous confidence intervals can be computed using multivariate normal distribution, multivariate t-distribution with a Satterthwaite Approximation of the degree of freedom or using multivariate range preserving transformations with Logit or Probit as transformation function. 2 sample comparisons can be performed with the same methods described above. There is no assumption on the underlying distribution function, only that the data have to be at least ordinal numbers. See Konietschke et al. (2015) <doi:10.18637/jss.v064.i09> for details.  
**License** GPL  
**LazyLoad** yes  
**Depends** multcomp, mvtnorm  
**NeedsCompilation** no  
**Repository** CRAN  
**Date/Publication** 2019-06-25 12:00:03 UTC  

### R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>nparcomp-package</td>
<td>2</td>
</tr>
<tr>
<td>appetite</td>
<td>4</td>
</tr>
<tr>
<td>colu</td>
<td>5</td>
</tr>
<tr>
<td>gao</td>
<td>6</td>
</tr>
<tr>
<td>gao_cs</td>
<td>7</td>
</tr>
<tr>
<td>impla</td>
<td>8</td>
</tr>
<tr>
<td>liver</td>
<td>9</td>
</tr>
<tr>
<td>mctp</td>
<td>10</td>
</tr>
</tbody>
</table>
nparcomp-package

Description

With this package, it is possible to compute nonparametric simultaneous confidence intervals for relative contrast effects in the unbalanced one way layout. Moreover, it computes simultaneous p-values. The simultaneous confidence intervals can be computed using multivariate normal distribution, multivariate t-distribution with a Satterthwaite Approximation of the degree of freedom or using multivariate range preserving transformations with Logit or Probit as transformation function. 2 sample comparisons can be performed with the same methods described above. There is no assumption on the underlying distribution function, only that the data have to be at least ordinal numbers.

Details

Package: nparcomp
Type: Package
Version: 1.0-0
Date: 2012-06-22
License: GPL
Author(s)

Frank Konietschke
Maintainer: Frank Konietschke <fkoniet@gwdg.de>

References


Examples

```r
## Not run:
# two sample comparisons: Nonparametric Behrens-Fisher Problem

data(impla)

a<-npar.t.test(impla~group, data = impla,
               method = "t.app",
               alternative = "two.sided")
summary(a)
plot(a)

#--Analysis of relative contrast effects in different contrast settings

data(liver)

# Williams Contrast

a<-nparcomp(weight ~dosage, data=liver, asy.method = "probit",
            type = "Williams", alternative = "two.sided",
            plot.simci = TRUE, info = FALSE)
summary(a)

# Dunnett dose 3 is baseline

c<-nparcomp(weight ~dosage, data=liver, asy.method = "probit",
            type = "Dunnett", control = "3",alternative = "two.sided",
            plot.simci = TRUE, info = FALSE)
summary(c)

# data(colu)
```
# Tukey comparison - one sided(lower)

```r
a <- nparcomp(corpora ~ dose, data = colu, asy.method = "mult.t",
type = "Tukey", alternative = "less")
summary(a)
plot(a)
```

# Tukey comparison - one sided(greater)

```r
b <- nparcomp(corpora ~ dose, data = colu, asy.method = "mult.t",
type = "Tukey", alternative = "greater")
summary(b)
plot(b)
```

## End(Not run)

---

**appetite**

*Appetite scores of colorectal cancer patients*

---

**Description**

Data from one of the quality of life measurements collected from colorectal cancer patients enrolled in the North Central Cancer Treatment Group phase III trials N9741. The patient received three treatment regimens: IFL (irinotecan, bolus fluorouracil, and leucovorin), FOLFOX (infused fluorouracil, leucovorin, and oxaliplatin), and IROX (irinotecan and oxaliplatin).

**Usage**

data(appetite)

**Format**

A data frame with 174 observations on the following 2 variables.

- **Group** A factor with levels FOLFOX IFL IROX.
- **Score** A numeric vector containing the appetite scores.

**Details**

The objective is to test whether there are differences between the treatment regimens in terms of different appetite scores.

**Source**

Examples

## Not run:
library(nparcomp)
data(appetite)

## End(Not run)

colu

### Description

Data from a fertility trial with 92 female Wistar rats: numbers of the corpora lutea in a placebo group and in 4 dose groups with an increasing dose of an active treatment.

### Usage

data(colu)

### Format

A data frame with 92 observations on the following 2 variables.

dose: A factor with levels dose1, dose2, dose3, dose4, Placebo, where Placebo is the placebo group and dose1-dose4 are the 4 dose groups with an increasing dose.

corpora: A numeric vector containing the numbers of the corpora lutea.

### Details

The objective is to test if the active treatment influences the fertility of the rats.

### Source


### Examples

## Not run:
library(nparcomp)
data(colu)
boxplot(corpora~dose,data=colu)

## End(Not run)
**gao**  
*Nonparametric multiple test procedure for many-to-one comparisons*

**Description**
This function can be used to perform the nonparametric multiple tests for many-to-one comparisons by Gao et al. (2008). The multiple level is strongly controlled by the Hochberg-adjustment.

**Usage**
gao(formula, data, alpha = 0.05, control = NULL, silent = FALSE)

**Arguments**
- **formula**: A two-sided 'formula' specifying a numeric response variable and a factor with more than two levels. If the factor contains less than 3 levels, an error message will be returned.
- **data**: A dataframe containing the variables specified in formula.
- **alpha**: The significance level (by default = 0.05).
- **control**: Character string defining the control group in Dunnett comparisons. By default it is the first group by lexicographical ordering.
- **silent**: A logical indicating more informations should be print on screen.

**Value**
- **Info**: Samples and sizes with estimated relative effects and variance estimators.

**Note**
The procedure can only be used to test hypotheses in terms of the distribution functions.

**Author(s)**
Frank Konietschke

**References**
See Also
For nonparametric all-pairs comparison see `gao_cs`.

Examples

```r
## Not run:
data(liver)
gao(weight ~ dosage, data=liver, alpha=0.05)
  # Control= 3
gao(weight ~ dosage, data=liver, alpha=0.05, control="3")
## End(Not run)
```

```r
gao_cs(formula, data, alpha = 0.05, silent = FALSE)
```

Arguments

- `formula`: A two-sided 'formula' specifying a numeric response variable and a factor with more than two levels. If the factor contains less than 3 levels, an error message will be returned.
- `data`: A dataframe containing the variables specified in formula.
- `alpha`: The significance level (by default = 0.05).
- `silent`: A logical indicating more informations should be print on screen.

Value

- `Info`: Samples and sizes with estimated relative effects and variance estimators.

Description

This function can be used to perform the nonparametric multiple tests for all-pairs comparisons by Gao et al. (2008). This procedure is a nonparametric equivalent of Campbell and Skillings (1981) sequential test procedure.
impla

Data from a fertility trial with 29 female Wistar rats: numbers of the implantations in a placebo group and in an active treatment group.

Usage

data(impla)
liver

Format

A data frame with 29 observations on the following 2 variables.

group A factor with levels Placebo, Verum, where Verum denotes the active treatment group.
impla A numeric vector.

Details

The objective is to test if the active treatment influences the fertility of the rats.

Source


Examples

## Not run:
library(nparcomp)
data(impla)
boxplot(impla~group,data=impla)
## End(Not run)

liver

Relative liver weights

Description

Data from a toxicity trial with male Wistar rats: Relative liver weights in a negative control group and in 4 dose groups with an increasing dose of an active treatment. After treatment the relative liver weights of the rats were computed.

Usage

data(liver)

Format

A data frame with 38 observations on the following 2 variables.
dosage A numeric vector indicating the dose/control group.
weight A numeric vector containing the relative liver weights.

Details

The objective is to test if the active treatment influences the liver weight of the rats.
mctp

Nonparametric multiple contrast tests and simultaneous confidence intervals (independent samples)

Description

The function mctp computes the estimator of nonparametric relative effects based on global rankings, simultaneous confidence intervals for the effects, and adjusted p-values based on contrasts in the setting of independent samples. Contrasts include "Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "GrandMean", and "UserDefined". The statistics are computed using multivariate normal distribution, multivariate Satterthwaite t-Approximation, and multivariate transformations (adjusted log odds or Fisher function). The function 'mctp' computes both the one-sided and two-sided simultaneous confidence intervals and adjusted p-values. The simultaneous confidence intervals can be plotted.

Usage

mctp(formula, data, type = c("Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "GrandMean", "UserDefined"), conf.level = 0.95, alternative = c("two.sided", "less", "greater"), asy.method = c("fisher", "mult.t", "normal", "log.odds"), plot.simci = FALSE, control = NULL, info = TRUE, rounds = 3, contrast.matrix = NULL, correlation = FALSE, effect=c("unweighted","weighted"), const=1/1.702)

Arguments

formula A two-sided 'formula' specifying a numeric response variable and a factor with more than two levels. If the factor contains less than 3 levels, an error message will be returned.

data A dataframe containing the variables specified in formula.

type Character string defining the type of contrast. It should be one of "Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "GrandMean", "UserDefined".

Examples

## Not run:
data(liver)
boxplot(weight~dosage, data=liver)

## End(Not run)
The confidence level for \texttt{conf.level}-confidence intervals (default is 0.95).

Character string defining the alternative hypothesis, one of "two.sided", "less", or "greater".

Character string defining the asymptotic approximation method, one of "fisher" (for using the Fisher transformation function), "log.odds" (for using the adjusted log odds effect sizes), "mult.t" (for using a multivariate t-distribution with a Satterthwaite Approximation), or "normal" (for using the multivariate normal distribution), "log.odds" (for using the adjusted log odds effect sizes).

A logical indicating whether you want a plot of the confidence intervals.

Character string defining the control group in Dunnett comparisons. By default, it is the first group by definition of the factor variable.

A logical whether you want a brief overview with informations about the output.

Number of rounds for the numeric values of the output (default is 3).

User-defined contrast matrix.

A logical whether the estimated correlation matrix and covariance matrix should be printed.

Character string defining the type of effect, one of "unweighted" and "weighted".

Number used for the adjustment of log odds when the "log.odds" option is chosen.

List of samples and sample sizes and estimated effect per group.

Contrast matrix.


The same as \texttt{Analysis} except that it assumes \texttt{rounds} = Inf.

The critical value and adjusted p-value for the overall hypothesis.

List of input arguments by user.

Character string specifying the alternative hypotheses.

Character string specifying the weight pattern for the reference distribution.

Character string specifying the contrast names.

Character string specifying the approximation method.

If the samples are completely separated the variance estimators are Zero by construction. In these cases the Null-estimators are replaced by 0.001. Estimated relative effects with 0 or 1 are replaced with 0.001, 0.999 respectively.

A summary and a graph can be created separately by using the functions \texttt{summary.mctp} and \texttt{plot.mctp}.

For the analysis, the R packages 'multcomp' and 'mvtnorm' are required.
Author(s)
Frank Konietschke, Kimihiro Noguchi

References

See Also
For simultaneous confidence intervals for relative contrast effects, see nparcomp.

Examples
```r
## Not run:
data(liver)

# Williams Contrast
a<-mctp(weight ~dosage, data=liver, asy.method = "fisher",
      type = "Williams", alternative = "two.sided",
      plot.simci = TRUE, info = FALSE)
summary(a)

# Dunnett Contrast
b<-mctp(weight ~dosage, data=liver, asy.method = "fisher",
      type = "Dunnett", alternative = "two.sided",
      plot.simci = TRUE, info = FALSE)
summary(b)

# Dunnett dose 3 is baseline
c<-mctp(weight ~dosage, data=liver, asy.method = "log.odds",
      type = "Dunnett", control = "3", alternative = "two.sided",
      plot.simci = TRUE, info = FALSE)
summary(c)

data(colu)

# Tukey comparison- one sided (less)
a<-mctp(corpora~ dose, data=colu, asy.method = "log.odds",
      type = "Tukey", alternative = "less",
      plot.simci = TRUE, info = FALSE)
summary(a)
```
mctp.rm
Nonparametric multiple contrast tests and simultaneous confidence intervals (repeated measures)

Description
The function mctp.rm computes the estimator of nonparametric relative effects based on global rankings, simultaneous confidence intervals for the effects, and adjusted p-values based on contrasts in the setting of a repeated measures design with n independent individuals and d repeated measures. Contrasts include "Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "GrandMean", and "UserDefined". The statistics are computed using multivariate normal distribution, multivariate Satterthwaite t-Approximation, and multivariate transformations (adjusted log odds or Fisher function). The function 'mctp.rm' computes both the one-sided and two-sided simultaneous confidence intervals and adjusted p-values. The confidence intervals can be plotted.

Usage
mctp.rm(formula, data, type = c("Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "GrandMean", "UserDefined"), conf.level = 0.95, alternative = c("two.sided", "less", "greater"), asy.method = c("log.odds", "fisher", "mult.t", "normal"), plot.simci = FALSE, control = NULL, info = TRUE, rounds = 3, contrast.matrix = NULL, correlation = FALSE, const=1/1.702)
Arguments

**formula**
A two-sided 'formula' specifying a numeric response variable and a repeated measures factor with more than two levels. If the factor contains less than 3 levels, an error message will be returned.

**data**
A dataframe containing the variables specified in formula.

**type**
Character string defining the type of contrast. It should be one of "Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "GrandMean", "UserDefined".

**conf.level**
The confidence level for confidence intervals (default is 0.95).

**alternative**
Character string defining the alternative hypothesis, one of "two.sided", "less", or "greater".

**asy.method**
Character string defining the asymptotic approximation method, one of "log.odds" (for using the adjusted log odds effect sizes), "mult.t" (for using a multivariate t-distribution with a Satterthwaite Approximation), "fisher" (for using the Fisher transformation function), or "normal" (for using the multivariate normal distribution).

**plot.simci**
A logical indicating whether you want a plot of the confidence intervals.

**control**
Character string defining the control group in Dunnett comparisons. By default, it is the first group by definition of the factor variable.

**info**
A logical whether you want a brief overview with informations about the output.

**rounds**
Number of rounds for the numeric values of the output (default is 3).

**contrast.matrix**
User-defined contrast matrix.

**correlation**
A logical whether the estimated correlation matrix and covariance matrix should be printed.

**const**
Number used for the adjustment of log odds when the "log.odds" option is chosen.

Value

**Data.Info**
List of samples and sample sizes and estimated effect per repeated measures level.

**Contrast**
Contrast matrix.

**Analysis**

**Analysis.Inf**
The same as Analysis except that it assumes rounds = Inf.

**Overall**
The critical value and adjusted p-value for the overall hypothesis.

**input**
List of input arguments by user.

**text.Output**
Character string specifying the alternative hypotheses.

**connames**
Character string specifying the contrast names.

**AsyMethod**
Character string specifying the approximation method.
Note

Estimated relative effects with 0 or 1 are replaced with 0.001 and 0.999.
A summary and a graph can be created separately by using the functions `summary.mctp.rm` and `plot.mctp.rm`.
For the analysis, the R packages `multcomp` and `mvtnorm` are required.

Author(s)

Marius Placzek, Kimihiro Noguchi

References


See Also

To analyse simple one-way layouts with independent samples use `mctp`.

Examples

```r
## Not run:
data(panic)
a<-mctp.rm(CGI~week, data=panic, type = "Dunnett",
            alternative = "two.sided",
            asy.method = "log.odds", plot.simci = FALSE,
            info = FALSE, contrast.matrix = NULL)
summary(a)
plot(a)

b<-mctp.rm(CGI~week, data=panic, type = "Dunnett",
            alternative = "two.sided",
            asy.method = "mult.t", plot.simci = FALSE,
            info = FALSE, contrast.matrix = NULL)
summary(b)
plot(b)

c<-mctp.rm(CGI~week, data=panic, type = "Dunnett",
            alternative = "two.sided",
            asy.method = "fisher", plot.simci = FALSE,
            info = FALSE, contrast.matrix = NULL)
summary(c)
plot(c)

d<-mctp.rm(CGI~week, data=panic, type = "Tukey",
            alternative = "two.sided",
            asy.method = "mult.t", plot.simci = TRUE)
summary(d)

## End(Not run)
```
The nonparametric Behrens-Fisher problem

Description

The function npar.t.test performs two sample tests for the nonparametric Behrens-Fisher problem, that is testing the hypothesis

\[ H_0 : p = 1/2 \]

where \( p \) denotes the relative effect of 2 independent samples and computes confidence intervals for the relative effect \( p \). The statistics are computed using standard normal distribution, Satterthwaite \( t \)-Approximation and variance stabilising transformations (Probit and Logit transformation function). For small samples there is also a studentized permutation test implemented. npar.t.test also computes one-sided and two-sided confidence intervals and \( p \)-values. The confidence interval can be plotted.

Usage

```r
npar.t.test(formula, data, conf.level = 0.95, alternative = c("two.sided", "less", "greater"), rounds = 3, method = c("logit", "probit", "normal", "t.app", "permu"), plot.simci = FALSE, info = TRUE, nperm=10000)
```

Arguments

- **formula**: A two-sided 'formula' specifying a numeric response variable and a factor with two levels. If the factor contains more than two levels, an error message will be returned.
- **data**: A dataframe containing the variables specified in formula.
- **conf.level**: The confidence level (default is 0.95).
- **alternative**: Character string defining the alternative hypothesis, one of "two.sided", "less" or "greater".
- **rounds**: Number of rounds for the numeric values of the output (default is 3).
- **method**: Character string defining the (asymptotic approximation) method, one of "logit", for using the logit transformation function, "probit", for using the probit transformation function, "normal", for using the standard normal distribution or "t.app" for using a \( t \)-Distribution with a Satterthwaite Approximation. The studentized permutation test can be obtained by choosing "permu".
- **plot.simci**: A logical indicating whether you want a plot of the confidence interval.
- **info**: A logical whether you want a brief overview with informations about the output.
- **nperm**: The number of permutations for the studentized permutation test. By default it is nperm=10,000.
Value

Info
List of samples and sample sizes.

Analysis
Effect: relative effect p(a,b) of the two samples ’a’ and ’b’, Estimator: estimated relative effect, Lower: Lower limit of the confidence interval, Upper: Upper limit of the confidence interval, T: teststatistic p.Value: p-value for the hypothesis by the choosen approximation method.

input
List of input by user.

Note
If the samples are completely separated the variance estimators are Zero by construction. In these cases the Null-estimators are replaced by a replacing method as proposed in the paper from Neubert and Brunner (2006). Estimated relative effects with 0 or 1 are replaced with 0.001, 0.999 respectively.

A summary and a graph can be created separately by using the functions `summary.nparttest` and `plot.nparttest`.

Author(s)
Frank Konietschke

References


See Also
For multiple comparison procedures based on relative effects, see `nparcomp`.

Examples

```r
## Not run:
data(impla)
a<-npar.t.test(impla~group, data = impla, method = "t.app",
               alternative = "two.sided", info=FALSE)
summary(a)
plot(a)
b<-npar.t.test(impla~group, data = impla, method = "permu",
               alternative = "two.sided", info=FALSE)
summary(b)
plot(b)
```
npar.t.test.paired  A 2-sample nonparametric studentized permutation test for paired data

Description
The function npar.t.test.paired performs a two sample studentized permutation test for paired data, that is testing the hypothesis

$$H_0 : p = 1/2$$

where p denotes the relative effect of 2 dependent samples, and computes a confidence interval for the relative effect p. In addition the Brunner-Munzel-Test accompanied by a confidence interval for the relative effect is implemented. npar.t.test.paired also computes one-sided and two-sided confidence intervals and p-values. The confidence interval can be plotted.

Usage

```r
npar.t.test.paired(formula, data, conf.level = 0.95, alternative = c("two.sided", "less", "greater"), nperm=10000, rounds = 3, info = TRUE, plot.simci = TRUE)
```

Arguments

- `formula` A two-sided 'formula' specifying a numeric response variable and a factor with two levels. If the factor contains more than two levels, an error message will be returned.
- `data` A dataframe containing the variables specified in formula.
- `conf.level` The confidence level (default is 0.95).
- `alternative` Character string defining the alternative hypothesis, one of "two.sided", "less" or "greater".
- `nperm` The number of permutations for the studentized permutation test. By default it is nperm=10,000.
- `rounds` Number of rounds for the numeric values of the output (default is 3).
- `info` A logical whether you want a brief overview with informations about the output.
- `plot.simci` A logical indicating whether you want a plot of the confidence interval.

Value

- `Info` List of samples and sample sizes.
- `Analysis` Effect: relative effect p(a,b) of the two samples 'a' and 'b', p.hat: estimated relative effect, Lower: Lower limit of the confidence interval, Upper: Upper limit of the confidence interval, T: studentized teststatistic p.value: p-value for the hypothesis.
- `input` List of input by user.
Note

A summary and a graph can be created separately by using the functions `summary.nparttestpaired` and `plot.nparttestpaired`.

Make sure that your dataset is ordered by subjects before applying npar.t.test.paired.

Author(s)

Frank Konietschke

References


See Also

For multiple comparison procedures based on relative effects, see `nparcomp`.

Examples

```r
## Not run:

data(PGI)
a<-npar.t.test.paired(PGIscore~timepoint, data = PGI,
                      alternative = "two.sided", info=FALSE, plot.simci=FALSE)
summary(a)
plot(a)

## End(Not run)
```

nparcomp

Nonparametric relative contrast effects

Description

The function nparcomp computes the estimator of nonparametric relative contrast effects, simultaneous confidence intervals for the effects and simultaneous p-values based on special contrasts like "Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "UserDefined". The statistics are computed using multivariate normal distribution, multivariate Satterthwaite t-Approximation and multivariate transformations (Probit and Logit transformation function). The function `nparcomp` also computes one-sided and two-sided confidence intervals and p-values. The confidence intervals can be plotted.
Usage

nparcomp(formula, data, type = c("Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "UserDefined"), control = NULL, conf.level = 0.95, alternative = c("two.sided", "less", "greater"), rounds = 3, correlation = FALSE, asy.method = c("logit", "probit", "normal", "mult.t"), plot.simci = FALSE, info = TRUE, contrast.matrix=NULL, weight.matrix=FALSE)

Arguments

formula A two-sided 'formula' specifying a numeric response variable and a factor with more than two levels. If the factor contains less than 3 levels, an error message will be returned.

data A dataframe containing the variables specified in formula.

type Character string defining the type of contrast. It should be one of "Tukey", "Dunnett", "Sequen", "Williams", "Changepoint", "AVE", "McDermott", "Marcus", "UmbrellaWilliams", "UserDefined".

control Character string defining the control group in Dunnett comparisons. By default it is the first group by definition of the dataset.

conf.level The confidence level for the confidence intervals (default is 0.95).

alternative Character string defining the alternative hypothesis, one of "two.sided", "less" or "greater".

rounds Number of rounds for the numeric values of the output. By default it is rounds=3.

correlation A logical whether the estimated correlation matrix and covariance matrix should be printed.

asy.method Character string defining the asymptotic approximation method, one of "logit", for using the logit transformation function, "probit", for using the probit transformation function, "normal", for using the multivariate normal distribution or "mult.t" for using a multivariate t-distribution with a Satterthwaite Approximation.

plot.simci A logical indicating whether you want a plot of the confidence intervals.

info A logical whether you want a brief overview with informations about the output.

contrast.matrix User defined contrast matrix.

weight.matrix A logical indicating whether the weight matrix should be printed.

Value

Data.Info List of samples and sample sizes.

Contrast Contrast matrix.
Analysis


input

List of input by user.

Note

If the samples are completely separated the variance estimators are Zero by construction. In these cases the Null-estimators are replaced by 0.001. Estimated relative effects with 0 or 1 are replaced with 0.001, 0.999 respectively.

A summary and a graph can be created separately by using the functions summary.nparcomp and plot.nparcomp.

For the analysis, the R packages 'multcomp' and 'mvtnorm' are required.

Author(s)

Frank Konietschke

References


See Also

For two-sample comparisons based on relative effects, see npar.t.test.

Examples

```r
## Not run:
data(liver)

# Williams Contrast

a<-nparcomp(weight ~dosage, data=liver, asy.method = "probit",
            type = "Williams", alternative = "two.sided",
            plot.simci = TRUE, info = FALSE,correlation=TRUE)
summary(a)

# Dunnett dose 3 is baseline

c<-nparcomp(weight ~dosage, data=liver, asy.method = "probit",
            type = "Dunnett", control = "3",
            plot.simci = TRUE, info = FALSE,correlation=TRUE)
summary(c)
```

```r
```
data(colu)

# Tukey comparison- one sided(lower)

a<-nparcomp(corpora~ dose, data=colu, asy.method = "mult.t",
            type = "Tukey",alternative = "less",
            plot.simci = TRUE, info = FALSE)
summary(a)

# Tukey comparison- one sided(greater)

b<-nparcomp(corpora~ dose, data=colu, asy.method = "mult.t",
            type = "Tukey",alternative = "greater",
            plot.simci = TRUE, info = FALSE)
summary(b)

## End(Not run)

---

**panic**

*Clinical Global Impression (CGI) Scores*

**Description**

Scores for the clinical global impression (CGI) measured on an ordinal scale (ranging from 2 to 8) during eight weeks for 16 patients with panic disorder attacks in a psychiatric clinical trial.

**Usage**

data(panic)

**Format**

A data frame with 80 observations on the following 2 variables.

- **CGI** A numeric vector containing the CGI score.
- **week** A numeric vector indicating the week (0,2,4,6,8) of measurement.

**Details**

Note that the first observation in each week corresponds to the first patient, the second one to the second patient, and so on. There are 5 repeated measures per patient.
Source

Examples
```r
## Not run:
data(panic)
boxplot(CGI~week,data=panic)
## End(Not run)
```

---

**PGI**  
*Patient Rated Global Impression (PGI) Scores*

Description
Scores for the patient rated global impression (PGI) measured on an ordinal scale (ranging from 1 to 6) being observed at baseline and after 4 weeks of treatment. The lower the score, the better the clinical impression.

Usage
data(PGI)

Format
A data frame with 30 observations on the following 3 variables.

- **patient**: A numeric vector indicating the patients.
- **timepoint**: A numeric vector indicating the week (0, 2, 4, 6, 8) of measurement.
- **PGIscore**: A numeric vector containing the PGI score.

Source

Examples
```r
## Not run:
data(PGI)
boxplot(PGIscore~timepoint,data=PGI)
## End(Not run)
```
plot.mctp

Visualizing the result of mctp

Description

This function takes an object of class "mctp" and creates a plot of the confidence intervals for the estimated effects.

Usage

## S3 method for class 'mctp'
plot(x,...)

Arguments

x An object of class "mctp", i.e. the result when applying mctp to a dataset. Otherwise an error will occur.
...
Arguments to be passed to methods.

Details

It is not possible to change any parameter set in the mctp-statement.

Since plot.mctp is a S3 method it suffices to use plot(x) as long as x is of class "mctp". It will be interpreted as plot.mctp(x).

Value

plot.mctp returns a graph that contains a confidence interval for the estimated effect of each contrast. It just visualizes the result of the mctp-statement.

Note

It is possible to create a graphical result of the multiple comparison test procedure directly by setting plot.simci=TRUE in the mctp-statement.

To get a complete result summary of mctp the function summary.mctp can be used.

Author(s)

Frank Konietschke, Kimihiro Noguchi

References


See Also

For further information on the usage of mctp, see mctp.
Examples

```r
data(liver)
a <- mctp(weight ~ dosage, data = liver, asy.method = "fisher",
        type = "Dunnett", alternative = "two.sided", plot.simci = FALSE,
        info = FALSE)
plot(a)
```

Description

This function takes an object of class "mctp.rm" and creates a plot of the confidence intervals for the estimated effects.

Usage

```r
## S3 method for class 'mctp.rm'
plot(x, ...)
```

Arguments

- `x` An object of class "mctp.rm", i.e. the result when applying `mctp.rm` to a dataset. Otherwise an error will occur.
- `...` Arguments to be passed to methods.

Details

It is not possible to change any parameter set in the `mctp.rm`-statement.

Since plot.mctp.rm is a S3 method it suffices to use `plot(x)` as long as `x` is of class "mctp.rm". It will be interpreted as `plot.mctp.rm(x)`.

Value

`plot.mctp.rm` returns a graph that contains a confidence interval for the estimated effect of each contrast. It just visualizes the result of the `mctp.rm`-statement.

Note

It is possible to create a graphical result of the multiple comparison test procedure directly by setting `plot.simci=TRUE` in the `mctp.rm`-statement.

To get a complete result summary of `mctp.rm` the function `summary.mctp.rm` can be used.

Author(s)

Marius Placzek, Kimihiro Noguchi
References


See Also

For further information on the usage of mctp.rm, see mctp.rm.

Examples

```r
## Not run:
data(panic)
a<-mctp.rm(CGI~week, data=panic, type = "Dunnett",
  alternative = "two.sided",
  asy.method = "fisher", contrast.matrix = NULL)
plot(a)
## End(Not run)
```

---

`plot.nparcomp`  
Visualizing the result of `nparcomp`

Description

This function takes an object of class "nparcomp" and creates a plot of the confidence intervals for the estimated nonparametric contrast effects.

Usage

```r
## S3 method for class 'nparcomp'
plot(x,...)
```

Arguments

- `x` An object of class "nparcomp", i.e. the result when applying `nparcomp` to a dataset. Otherwise an error will occur.
- `...` Arguments to be passed to methods.

Details

It is not possible to change any parameter set in the `nparcomp`-statement.

Since `plot.nparcomp` is an S3 method it suffices to use `plot(x)` as long as `x` is of class "nparcomp". It will be interpreted as `plot.nparcomp(x)`.  

plot.nparttest

Value

plot.nparcomp returns a graph that contains a confidence interval for the estimated nonparametric contrast effect of each contrast. It just visualizes the result of the nparcomp-statement.

Note

It is possible to create a graphical result directly by setting plot.simci=TRUE in the nparcomp-statement.

Author(s)

Frank Konietschke

References


See Also

For further information on the usage of nparcomp, see nparcomp.

Examples

```r
## Not run:
data(liver)
a<-nparcomp(weight ~dosage, data=liver, asy.method = "probit",
type = "Williams", alternative = "two.sided", plot.simci = FALSE, info = FALSE)
plot(a)
## End(Not run)
```

## Not run:
data(liver)
a<-nparcomp(weight ~dosage, data=liver, asy.method = "probit",
type = "Williams", alternative = "two.sided", plot.simci = FALSE, info = FALSE)
plot(a)
## End(Not run)
plot.nparttest

Arguments

x  x An object of class "nparttest", i.e. the result when applying \texttt{npar.t.test} to a dataset. Otherwise an error will occur.

\ldots  \ldots Arguments to be passed to methods.

Details

It is not possible to change any parameter set in the \texttt{npar.t.test}-statement.

Since plot.nparttest is a S3 method it suffices to use plot(x) as long as x is of class "nparttest". It will be interpreted as plot.nparttest(x).

Value

plot.npar.t.test returns a graph that contains a confidence interval for the estimated effect of the nonparametric t-test. It just visualizes the result of the \texttt{npar.t.test}-statement.

Note

It is possible to create a graphical result of the nonparametric t-test directly by setting plot.simci=TRUE in the \texttt{npar.t.test}-statement.

Author(s)

Frank Konietschke

References


See Also

For further information on the usage of npar.t.test, see \texttt{npar.t.test}.

Examples

\begin{verbatim}
## Not run:
data(impla)
a<-npar.t.test(impla~group, data = impla, method = "t.app",
alternative = "two.sided", plot.simci=FALSE)
plot(a)
## End(Not run)
\end{verbatim}
**plot.nparttestpaired**  
Visualizing the result of `npar.t.test.paired`

---

**Description**

This function takes an object of class "nparttestpaired" and creates a plot of the confidence intervals for the estimated effect resulting from the studentized permutation test and the Brunner-Munzel test.

**Usage**

```r
## S3 method for class 'nparttestpaired'
plot(x,...)
```

**Arguments**

- `x`  
  An object of class "nparttestpaired", i.e. the result when applying `npar.t.test.paired` to a dataset. Otherwise an error will occur.

- `...`  
  Arguments to be passed to methods.

**Details**

It is not possible to change any parameter set in the `npar.t.test.paired`-statement. Since `plot.nparttestpaired` is a S3 method it suffices to use `plot(x)` as long as `x` is of class "nparttestpaired". It will be interpreted as `plot.nparttestpaired(x)`.

**Value**

`plot.npar.t.test` returns a graph that contains a confidence interval for the estimated effect of the non-parametric studentized permutation test as well as. It just visualizes the result of the `npar.t.test.paired`-statement.

**Note**

It is possible to create a graphical result of the nonparametric studentized permutation test directly by setting `plot.simci=TRUE` in the `npar.t.test.paired`-statement.

**Author(s)**

Frank Konietschke

**References**


reaction

See Also
For further information on the usage of npar.t.test.paired, see npar.t.test.paired.

Examples
```r
## Not run:
data(PGI)
a <- npar.t.test.paired(PGI.score ~ timepoint, data = PGI,
                        alternative = "two.sided", info = TRUE, plot.simci = FALSE)
plot(a)
## End(Not run)
```

Description
Data from a toxicity trial with 40 mice.

Usage
data(reaction)

Format
A data frame with 40 observations on the following 2 variables.

- **Group**  A numeric vector indicating the group.
- **Time**   A numeric vector containing the reaction times.

Details
The objective is to test if the active treatment influences the reaction time of the mice.

Source

References
**Examples**

```r
## Not run:
library(nparcomp)
data(reaction)
boxplot(Time~Group,data=reaction)
## End(Not run)
```

**summary.mctp**

**Summary of mctp**

**Description**

The function summary.mctp produces a result summary of `mctp`. It can only be applied to objects of class "mctp".

**Usage**

```r
## S3 method for class 'mctp'
summary(object,...)
```

**Arguments**

- `object` An object of class "mctp", i.e. the result when applying `mctp` to a dataset. Otherwise an error will occur.
- `...` Arguments to be passed to methods.

**Details**

Since summary.mctp is a S3 method it suffices to use `summary(x)` as long as `x` is of class "mctp". It will be interpreted as `summary.mctp(x)`.  

**Value**

The function produces a summary of the result of `mctp` starting with some global information: alternative hypothesis, estimation method, type of contrast, confidence level. This is followed by:

- **Data.Info** List of samples and sample sizes and estimated effect per group.
- **Contrast** Contrast matrix.

**Note**

It is possible to create a graphical result of the multiple comparison test procedure by using the function `plot.mctp`. 
Author(s)
Frank Konietschke

References

See Also
For further information on the usage of mctp, see mctp.

Examples
```r
## Not run:
data(liver)
a <- mctp(weight ~ dosage, data = liver, asy.method = "fisher",
        type = "Dunnett", alternative = "two.sided", plot.simci = FALSE,
        info = FALSE)
summary(a)
## End(Not run)
```

Description
The function summary.mctp.rm produces a result summary of mctp.rm. It can only be applied to objects of class "mctp.rm".

Usage
```r
## S3 method for class 'mctp.rm'
summary(object, ...)
```

Arguments
- **object**: An object of class "mctp.rm", i.e. the result when applying mctp.rm to a dataset. Otherwise an error will occur.
- **...**: Arguments to be passed to methods.

Details
Since summary.mctp.rm is a S3 method it suffices to use summary(x) as long as x is of class "mctp.rm". It will be interpreted as summary.mctp.rm(x).
Value

The function produces a summary of the result of `mctp.rm` starting with some global information: alternative hypothesis, estimation method, type of contrast, confidence level. This is followed by:

Data.Info List of samples and sample sizes and estimated effect per group.
Contrast Contrast matrix.

Note

It is possible to create a graphical result of the multiple comparison test procedure by using the function `plot.mctp.rm`.

Author(s)

Marius Placzek

References


See Also

For further information on the usage of mctp.rm, see `mctp.rm`.

Examples

```r
## Not run:
data(panic)
a<-mctp.rm(CGI~week, data=panic, type = "Dunnett",
           alternative = "two.sided",
           asy.method = "fisher", contrast.matrix = NULL)
summary(a)
## End(Not run)
```
**summary.nparcomp**

**Summary of nparcomp**

**Description**

The function `summary.nparcomp` produces a result summary of `nparcomp`. It can only be applied to objects of class "nparcomp".

**Usage**

```r
## S3 method for class 'nparcomp'
summary(object,...)
```

**Arguments**

- `object` 
  An object of class "nparcomp", i.e. the result when applying `nparcomp` to a dataset. Otherwise an error will occur.

- `...` 
  Arguments to be passed to methods.

**Details**

Since `summary.nparcomp` is a S3 method it suffices to use `summary(x)` as long as `x` is of class "nparcomp". It will be interpreted as `summary.nparcomp(x)`.

**Value**

The function produces a summary of the result of `nparcomp` starting with some global information: alternative hypothesis, estimation method, type of contrast, confidence level, method, interpretation. This is followed by:

- `Data.Info` 
  List of samples and sample sizes.

- `Contrast` 
  Contrast matrix.

- `Analysis` 

- `Overall` 
  Overall p-value and critical value.

**Note**

It is possible to create a graphical result of the nonparametric test procedure `nparcomp` by using the function `plot.nparcomp`.

**Author(s)**

Frank Konietschke
summary.nparttest

References


See Also

For further information on the usage of nparcomp, see nparcomp.

Examples

## Not run:
data(liver)
a<-nparcomp(weight ~dosage, data=liver, asy.method = "probit", type = "Williams", alternative = "two.sided", plot.simci = FALSE, info = FALSE)
summary(a)
## End(Not run)

summary.nparttest Summary of npar.t.test

Description

The function summary.npar.t.test produces a result summary of npar.t.test. It can only be applied to objects of class "nparttest".

Usage

## S3 method for class 'nparttest'
summary(object,...)

Arguments

object An object of class "nparttest", i.e. the result when applying npar.t.test to a dataset. Otherwise an error will occur.

Arguments to be passed to methods.

Details

Since summary.npar.test is a S3 method it suffices to use summary(x) as long as x is of class "nparttest". It will be interpreted as summary.nparttest(x).
Summary

The function produces a summary of the result of \texttt{npar.t.test} starting with some global information: alternative hypothesis, confidence level, interpretation. This is followed by:

- **Info**: List of samples and sample sizes.
- **Analysis**: Effect: relative effect $p(a,b)$ of the two samples 'a' and 'b', Estimator: estimated relative effect, Lower: Lower limit of the confidence interval, Upper: Upper limit of the confidence interval, T: test statistic, p.Value: p-value for the hypothesis by the chosen approximation method.
- **Permutation Test**: Result of the studentized permutation test.

Note

You can create a graphical result of the nonparametric t-test by using the function \texttt{plot.nparttest}.

Author(s)

Frank Konietschke

References


See Also

For further information on the usage of \texttt{npar.t.test}, see \texttt{npar.t.test}.

Examples

```r
## Not run:
data(impla)
a <- npar.t.test(impla~group, data = impla, method = "t.app",
                alternative = "two.sided",
                plot.simci=FALSE, info=FALSE)
summary(a)

## End(Not run)
```
**Summary of `npar.t.test`**

---

**Description**

The function `summary.nparttestpaired` produces a result summary of `npar.t.test.paired`. It can only be applied to objects of class "nparttestpaired".

**Usage**

```r
define S3 method for class 'nparttestpaired'
summary(object,...)
```

**Arguments**

- `object`: An object of class "nparttestpaired", i.e. the result when applying `npar.t.test.paired` to a dataset. Otherwise an error will occur.
- `...`: Arguments to be passed to methods.

**Details**

Since `summary.nparttestpaired` is a S3 method it suffices to use `summary(x)` as long as `x` is of class "nparttestpaired". It will be interpreted as `summary.nparttestpaired(x)`.

**Value**

The function produces a summary of the result of `npar.t.test.paired` starting with some global information: alternative hypothesis, confidence level, interpretation. This is followed by:

- **Info**: List of samples and sample sizes.
- **Analysis**: Effect: relative effect \( p(a,b) \) of the two samples 'a' and 'b', \( \hat{p} \): estimated relative effect, Lower: Lower limit of the confidence interval, Upper: Upper limit of the confidence interval, \( T \): teststatistic, \( p.value \): \( p \)-value for the hypothesis by the choosen approximation method.

**Note**

You can create a graphical result of the nonparametric paired t-test by using the function `plot.nparttestpaired`.

**Author(s)**

Frank Konietschke

**References**


See Also

For further information on the usage of npar.t.test.paired, see `npar.t.test.paired`.

Examples

```r
## Not run:
data(PGI)
a<-npar.t.test.paired(PGIscore~timepoint, data = PGI, alternative = "two.sided", info=FALSE, plot.simci=FALSE)
summary(a)
## End(Not run)
```
Index

* Contrast
  gao, 6
  gao_cs, 7
  mctp, 10
  mctp.rm, 13
  nparcomp, 19

* Nonparametric Behrens-Fisher Problem in Paired Data
  npar.t.test.paired, 18

* Nonparametric Behrens-Fisher Problem
  npar.t.test, 16

* Nonparametric
  gao, 6
  gao_cs, 7
  mctp, 10
  mctp.rm, 13
  npar.t.test, 16
  npar.t.test.paired, 18
  nparcomp, 19

* Relative Effect
  gao, 6
  gao_cs, 7
  mctp, 10
  mctp.rm, 13
  npar.t.test, 16
  npar.t.test.paired, 18
  nparcomp, 19

* Repeated Measures
  mctp.rm, 13

* Simultaneous Confidene Intervals
  gao, 6
  gao_cs, 7
  mctp, 10
  mctp.rm, 13
  npar.t.test, 16
  npar.t.test.paired, 18
  nparcomp, 19

* aplot
  plot.mctp, 24
  plot.mctp.rm, 25
  plot.nparcomp, 26
  plot.nparttest, 27
  plot.nparttestpaired, 29

* datasets
  appetite, 4
  colu, 5
  impla, 8
  liver, 9
  panic, 22
  PGI, 23
  reaction, 30

* htest
  gao, 6
  gao_cs, 7
  mctp, 10
  mctp.rm, 13
  npar.t.test, 16
  npar.t.test.paired, 18
  nparcomp, 19
  nparcomp-package, 2

* package
  nparcomp-package, 2

* print
  summary.mctp, 31
  summary.mctp.rm, 32
  summary.nparcomp, 34
  summary.nparttest, 35
  summary.nparttestpaired, 37
  appetite, 4
  colu, 5
  gao, 6, 8
  gao_cs, 7, 7
  impla, 8
  liver, 9
  mctp, 10, 15, 24, 31, 32
  mctp.rm, 13, 25, 26, 32, 33

39
npar.t.test. 16, 21, 27, 28, 35–37
npar.t.test.paired, 18, 29, 30, 37, 38
nparcomp, 12, 17, 19, 19, 26, 27, 34, 35
nparcomp-package, 2

panic, 22
PGI, 23
plot.mctp, 11, 24, 31
plot.mctp.rm, 15, 25, 33
plot.nparcomp, 21, 26, 34
plot.nparttest, 17, 27, 36
plot.nparttestpaired, 19, 29, 37

reaction, 30

summary.mctp, 11, 24, 31
summary.mctp.rm, 15, 25, 32
summary.nparcomp, 21, 34
summary.nparttest, 17, 35
summary.nparttestpaired, 19, 37