Package ‘NPC’

August 29, 2016

Type Package
Title Nonparametric Combination of Hypothesis Tests
Version 1.1.0
Depends R (>= 3.2.0)
LazyData true
Description An implementation of nonparametric combination of hypothesis tests.
This package performs nonparametric combination (Pesarin and Salmaso 2010),
a permutation-based procedure for jointly testing multiple hypotheses. The
tests are conducted under the global "sharp" null hypothesis of no effects,
and the component tests are combined on the metric of their p-values. A
key feature of nonparametric combination is that it accounts for the
dependence among tests under the null hypothesis. In addition to the
"NPC" function, which performs nonparametric combination itself, the
package also contains a number of helper functions, many of which calculate
a test statistic given an input of data.
License GPL-3
Imports permute, dplyr, coin, matlab
Suggests car, mvtnorm, plyr, xtable
RoxygenNote 5.0.1
NeedsCompilation no
Author Devin Caughey [aut, cre]
Maintainer Devin Caughey <devin.caughey@gmail.com>
Repository CRAN
Date/Publication 2016-03-28 16:51:57

R topics documented:

DiffMeans ......................................................... 2
DiffSumObs ......................................................... 3
DiffSumWithNA ....................................................... 3
FWE ................................................................. 4
Description

Calculates the difference of means between two groups of observations. The result is intended to be used as a test statistic in a permutation test.

Usage

DiffMeans(y, tr, tl, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>Response variable (vector)</td>
</tr>
<tr>
<td>tr</td>
<td>Treatment variable (vector)</td>
</tr>
<tr>
<td>tl</td>
<td>The level of treatment variable (e.g., &quot;treated&quot; or 1) that indicates treated observations</td>
</tr>
<tr>
<td>...</td>
<td>Additional arguments (included for compatibility but ignored)</td>
</tr>
</tbody>
</table>

Value

A scalar test statistic

Note

Returns NA if there are any missing values in y or tr.

Author(s)

Devin Caughey <devin.caughey@gmail.com>
DiffSumObs

Difference in the Number of Non-Missing Responses

Description
Calculates the difference in the number of non-missing responses between two groups of observations. The result is intended to be used as a test statistic in a permutation test.

Usage
DiffSumObs(y, tr, tl, ...)

Arguments
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>Response variable (vector)</td>
</tr>
<tr>
<td>tr</td>
<td>Treatment variable (vector)</td>
</tr>
<tr>
<td>tl</td>
<td>The level of treatment variable (e.g., &quot;treated&quot; or 1) that indicates treated observations</td>
</tr>
<tr>
<td>...</td>
<td>Additional arguments (included for compatibility but ignored)</td>
</tr>
</tbody>
</table>

Value
A scalar test statistic

Note
May not behave well if the treatment variable contains missing values.

Author(s)
Devin Caughey <devin.caughey@gmail.com>

DiffSumWithNA

Sum Test Statistic for MCAR Data

Description
This function calculates a test statistic analogous to the difference of means, but adjusted to accommodate missing responses. The result is intended to be used as a test statistic in a permutation test. Under the assumption that the data are missing completely at random, the resulting permutation test is "nearly" exact under the null of distributional equality of the observed responses (Pesarin and Salmaso 2010, 234–44).

Usage
DiffSumWithNA(y, tr, tl, ...)

FWE Adjustment Using Closed Testing or Stepdown MinP

Description
This function adjusts a set of p-values to control the family-wise error rate (FWE). It does so using either closed testing (Marcus, Peritz, and Gabriel 1976) or the stepdown MinP algorithm (Westfall and Young 1993, 66–7). Because these methods take into account the dependence among the p-values, they are much less conservative than traditional FWE adjustments like the Bonferroni correction. Closed testing becomes very computationally intensive as the number of p-values increases.

Usage
FWE(PV, stepdown = TRUE, cfun = NULL)

Arguments
- **PV**: A matrix containing the permutation distribution of (pseudo) p-values. The first row should be the observed p-values and the remaining rows the permutation distribution.
- **stepdown**: Use stepdown MinP to adjust p-values? (logical)
- **cfun**: Combining function to be used for closed-testing adjustment (function)
**Value**

A vector of adjusted p-values

**Note**

Derived from the `FWE.minP` function (http://www.wiley.com/legacy/wileychi/pesarin/supp/R_functions.zip) provided by Pesarin and Salmaso (2010) and the `closetest` function (http://static.gest.unipd.it/~salmaso/web/clostest.r) provided by Basso et al. (2009).

**Author(s)**

Devin Caughey <devin.caughey@gmail.com>

**References**


---

### Description

Calculates the weighted average of the block-specific differences of means, where the weights are proportional to the harmonic mean cluster size in the block (Donner and Klar, 1993; Bowers, Fredrickson and Hansen, 2010). The result is intended to be used as a test statistic in a permutation test.

### Usage

```
HarmonicWtdMean(y, tr, tl, block, ...)  
```

### Arguments

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>Response variable (vector)</td>
</tr>
<tr>
<td>tr</td>
<td>Treatment variable (vector)</td>
</tr>
<tr>
<td>tl</td>
<td>The level of treatment variable (e.g., &quot;treated&quot; or 1) that indicates treated observations</td>
</tr>
<tr>
<td>block</td>
<td>Blocking/stratification variable (vector)</td>
</tr>
<tr>
<td>...</td>
<td>Additional arguments (included for compatibility but ignored)</td>
</tr>
</tbody>
</table>
Author(s)

Devin Caughey <devin.caughey@gmail.com>

References


Description

Implements the recommendations of Hogg et al. (1975) for data-driven selection of rank-based permutation test statistics. It is designed to select a powerful test statistic given the tailweight and skew of the distribution of $y$. Best used for continuous distributions that differ in location only (i.e., under a constant treatment effect assumption).

Usage

HoggAdapt(y)

Arguments

y response variable (numeric vector)

Value

fun, a function designed to be used by the function NPC

Author(s)

Devin Caughey <devin.caughey@gmail.com>

References


See Also

NPC
KS

Kolmogorov-Smirnov Test Statistic

Description
Calculates the two-sample Kolmogorov-Smirnov statistic comparing the treated and control distributions y. The result is intended to be used as a test statistic in a permutation test.

Usage
KS(y, tr, tl, ...)

Arguments
- y: Response variable (vector)
- tr: Treatment variable (vector)
- tl: The level of treatment variable (e.g., "treated" or 1) that indicates treated observations
- ...: Additional arguments (included for compatibility but ignored)

Value
A scalar test statistic

Note
Returns NA if there are any missing values in y or tr.

Author(s)
Devin Caughey <devin.caughey@gmail.com>

LogRank

Log-Rank Test Statistic

Description
Calculates the difference of means between two groups of observations. The result is intended to be used as a test statistic in a permutation test.

Usage
LogRank(y, tr, tl, event, block, ...)

Note
Returns NA if there are any missing values in y or tr.
Arguments

- \( y \) Response variable, typically a duration (vector)
- \( tr \) Treatment variable (vector)
- \( tl \) The level of treatment variable (e.g., "treated" or 1) that indicates treated observations
- \( event \) Status variable indicating for each observation whether \( y \) was censored (0 or FALSE) or observed (1 or TRUE).
- \( block \) Block or stratification variable (vector)
- \( \ldots \) Additional arguments (included for compatibility but ignored)

Value

A scalar test statistic

Author(s)

Devin Caughey <devin.caughey@gmail.com>

---

MinimumCF  

*Tippett’s Minimum Combining Function*

Description

Returns the minimum (multiplied by -1) of a vector of p-values, to be used as a global test statistic.

Usage

MinimumCF(p, B)

Arguments

- \( p \) Vector of p-values
- \( B \) Number of samples from permutation space (not used by this combining function)

Value

A scalar global test statistic

Author(s)

Devin Caughey <devin.caughey@gmail.com>

References

NormalCF

*Liptak’s Normal Combining Function*

**Description**

Returns the negative of the sum of the normal quantiles of a vector of p-values, to be used as a global test statistic.

**Usage**

`NormalCF(p, B)`

**Arguments**

- `p` Vector of p-values
- `B` Number of samples from permutation space, for transformation to the open (0, 1) interval.

**Value**

A scalar global test statistic

**Author(s)**

Devin Caughey <devin.caughey@gmail.com>

**References**


---

NPC

*Nonparametric Combination of Dependent Hypothesis Tests*

**Description**

This function conducts multiple hypothesis tests under the sharp null of no effects and combines the results into a single global p-value.
Usage

NPC(data, keep = TRUE, tr.var, tr.label, y.vars, comb.fun = "ProductCF", n.perms = 1000, block.var = NULL, clust.var = NULL, event.var = NULL, alternative = "greater", seed = 1, na.rm = TRUE, FWE.adj = TRUE, step.down = identical(comb.fun, "MinimumCF"), test.statistic = "StudentsT", return.matrix = FALSE, print.steps = TRUE, adapt.test = logical(length(y.vars)))

Arguments

data Data frame with treatment, response, and other variables
keep Subset of observations (default keeps all)
tr.var Name of treatment variable (character)
tr.label Level of 'tr.var' indicating treated units (character)
y.vars Names of response variables (character)
comb.fun Combining function (character or function)
n.perms Number of permutations, in addition to the one observed
block.var Variable defining blocks within which to restrict permutations (character)
clust.var Variable defining clusters of observations assigned to treatment en bloc (character)
event.var Logical variable indicating whether duration variables were observed rather than censored (character)
alternative Scalar or vector indicating the alternative hypotheses ("greater", "less", or "two.sided"). If "less", test statistics are multiplied by -1. If "two.sided", the absolute value is used.
seed Random seed (numeric)
na.rm Delete observations with missing values? (logical)
FWE.adj Calculate FWE-adjusted p-values? (logical)
step.down If performing FWE adjustment, use stepdown MinP? (logical)
test.statistic Vector or list of test statistic functions (possibly quoted) for marginal tests. If single value is given, then it is used for all marginal tests.
return.matrix Return the permutation distribution of test statistics and p-values? (logical)
print.steps Print progress of the function? (logical)
adapt.test Use the adaptive test of Hogg et al. (1975)? (logical vector)

Value

p.values vector of marginal p-values plus the joint NPC p-value
p.matrix (optional) matrix containing the permutation distribution of marginal p-values
T.matrix (optional) matrix containing the permutation distribution of test statistics
Author(s)
Devin Caughey <devin.caughey@gmail.com>

References

Examples

library(NPC)
## Required for this example
library(mvtnorm)
library(car)

## COVARIANCE = CORRELATION = 0.25
cov <- 0.25
N <- 8
Tr <- c(rep(0, 4), rep(1, 4))
d1 <- 1
d2 <- 1
d3 <- 1
sd <- 1
sigma <-
  matrix(c(1, cov, cov, cov, 1, cov, cov, cov, 1), ncol = 3)
## Create error matrix
set.seed(2)
ee <- rmvnorm(N, c(0, 0, 0), sigma)
mean(c(cor(ee)[1, 2], cor(ee)[3, 2], cor(ee)[1, 3]))

Y1 <- -.5 + Tr*d1 + ee[, 1]
Y2 <- -.5 + Tr*d2 + ee[, 2]
Y3 <- -.5 + Tr*d3 + ee[, 3]
## Create data
(ex.dta <- data.frame(Tr, Y1, Y2, Y3))
(diffs <- round(colMeans(subset(ex.dta, Tr == 1, -Tr)) -
  colMeans(subset(ex.dta, Tr == 0, -Tr)), 2))
mean(diffs)
## NPC
npc.out <- NPC(data=ex.dta, tr.var="Tr", tr.label=1,
y.vars=c("Y1", "Y2", "Y3"), n.perms=1000,
alternative = "greater", seed=1, comb.fun="NormalCF",
test.statistic="DiffMeans", FWE.adj=FALSE)
round(npc.out$p.value, 2) ## one-sided
##> Y1  Y2  Y3  NPC
##> 0.25 0.05 0.07 0.08

## Compare with T-tests and MANOVA
t.test(Y1 ~ Tr, var.equal = TRUE, alternative = "less")
t.test(Y2 ~ Tr, var.equal = TRUE, alternative = "less")
t.test(Y3 ~ Tr, var.equal = TRUE, alternative = "less")
car::anova(lm(cbind(Y1, Y2, Y3) ~ Tr)) ## two-sided

---

Map P-Values to Open Unit Interval

Description

This function maps a vector of p-values to the open unit interval (that is, it bounds them away from 0 and 1). The interval approaches [0,1] as B (the number of permutations) increases. Intended as input for a p-value combining function.

Usage

open01(p, B)

Arguments

- **p**: Vector of p-values
- **B**: Number of samples from the permutation space

Value

Vector of transformed p-values

Author(s)

Devin Caughey <devin.caughey@gmail.com>

References

**ProductCF**

*Fisher’s Product Combining Function*

---

**Description**

Returns the Fisher combination of vector of p-values, to be used as a global test statistic.

**Usage**

```
ProductCF(p, B)
```

**Arguments**

- `p` Vector of p-values
- `B` Number of samples from permutation space, for transformation to the open (0, 1) interval.

**Value**

A scalar global test statistic

**Note**

The function is labeled the "product" function because the Fisher combination is permutationally equivalent to the product of the p-values.

**Author(s)**

Devin Caughey <devin.caughey@gmail.com>

**References**


**RankSum**

*Wilcoxon’s Rank-Sum Statistic*

**Description**
Calculates the sum of the ranks of the responses of treated units. The result is intended to be used as a test statistic in a permutation test.

**Usage**

```r
RankSum(y, tr, tl, ...)```

**Arguments**
- `y`: Response variable (vector)
- `tr`: Treatment variable (vector)
- `tl`: The level of treatment variable (e.g., "treated" or 1) that indicates treated observations
- `...`: Additional arguments (included for compatibility but ignored)

**Value**
A scalar test statistic

**Note**
Returns NA if there are any missing values in `y` or `tr`.

**Author(s)**
Devin Caughey <devin.caughey@gmail.com>

---

**RankSumWithNA**

*Rank-Sum Test Statistic for MCAR Data*

**Description**
This function calculates a test statistic analogous to the rank sum, but adjusted to accommodate missing responses. The result is intended to be used as a test statistic in a permutation test. Under the assumption that the data are missing completely at random, the resulting permutation test is an "nearly" exact test of distributional equality of the observed responses (Pesarin and Salmaso 2010, 234–44).

**Usage**

```r
RankSumWithNA(y, tr, tl, ...)```
Arguments

\begin{itemize}
\item \texttt{y}  \hspace{1cm} Response variable (vector)
\item \texttt{tr}  \hspace{1cm} Treatment variable (vector)
\item \texttt{tl}  \hspace{1cm} The level of treatment variable (e.g., "treated" or 1) that indicates treated observations
\item \ldots  \hspace{1cm} Additional arguments (included for compatibility but ignored)
\end{itemize}

Value

A scalar test statistic

Author(s)

Devin Caughey <devin.caughey@gmail.com>

References


---

\textbf{Students\textit{T}}  \hspace{1cm} \textit{Student's T Statistic}

Description

Calculates the T statistic of the difference between treated and control. The result is intended to be used as a test statistic in a permutation test.

Usage

\texttt{Students\textit{T}(y, tr, tl, \ldots)}

Arguments

\begin{itemize}
\item \texttt{y}  \hspace{1cm} Response variable (vector)
\item \texttt{tr}  \hspace{1cm} Treatment variable (vector)
\item \texttt{tl}  \hspace{1cm} The level of treatment variable (e.g., "treated" or 1) that indicates treated observations
\item \ldots  \hspace{1cm} Additional arguments (included for compatibility but ignored)
\end{itemize}

Details

As Chung and Romano (2013) show, using the studentized difference of means as a permutation test statistic yields an asymptotically valid test of the "weak null" of mean equality between groups.
**Value**

A scalar test statistic

**Note**

Returns NA if there are any missing values in y or tr.

**Author(s)**

Devin Caughey <devin.caughey@gmail.com>

**References**


---

**StudentWilcoxon**

**Studentized Wilcoxon Rank-Sum Statistic**

**Description**

This function returns a studentized version of Wilcoxon’s rank sum statistic.

**Usage**

`StudentWilcoxon(y, tr, tl, ...)`

**Arguments**

- `y` Response variable (vector)
- `tr` Treatment variable (vector)
- `tl` The level of treatment variable (e.g., "treated" or 1) that indicates treated observations
- `...` Additional arguments (included for compatibility but ignored)

**Details**

As Chung and Romano (2013) show, using the studentized rank sum as a permutation test statistic yields an asymptotically valid test of the "weak null" that responses are equally likely to be higher under treatment and control.

**Value**

A scalar test statistic

**Note**

Returns NA if there are any missing values in y or tr.
Author(s)
Devin Caughey <devin.caughey@gmail.com>

References

---

**t2p**

*Convert Test Statistics into P-Values*

**Description**
This function converts the permutation distribution of a test statistic into p-values, one for each permutation. The resulting array of p-values is suitable for nonparametric combination.

**Usage**
`t2p(T)`

**Arguments**
- `T` A vector or array containing the distribution of a test statistic across permutations. If it is an array the first dimension must index permutations.

**Details**
For each permutation, the value of the test statistic is compared to its distribution across permutations. The (estimated) p-value for that permutation is the proportion of permutations.

**Value**
An array of p-values

**Note**
Derived from Pesarin and Salmaso’s t2p function (http://www.wiley.com/legacy/wileychi/pesarin/supp/R_functions.zip).

Author(s)
Devin Caughey <devin.caughey@gmail.com>

References
Index

*Topic \textasciitilde kwd1
  DiffMeans, 2
  DiffSumObs, 3
  DiffSumWithNA, 3
  FWE, 4
  HarmonicWtdMean, 5
  HoggAdapt, 6
  KS, 7
  LogRank, 7
  MinimumCF, 8
  NormalCF, 9
  NPC, 9
  open01, 12
  ProductCF, 13
  RankSum, 14
  RankSumWithNA, 14
  StudentsT, 15
  StudentWilcoxon, 16
  t2p, 17

*Topic \textasciitilde kwd2
  DiffMeans, 2
  DiffSumObs, 3
  DiffSumWithNA, 3
  FWE, 4
  HarmonicWtdMean, 5
  HoggAdapt, 6
  KS, 7
  LogRank, 7
  MinimumCF, 8
  NormalCF, 9
  NPC, 9
  open01, 12
  ProductCF, 13
  RankSum, 14
  RankSumWithNA, 14
  StudentsT, 15
  StudentWilcoxon, 16
  t2p, 17

DiffMeans, 2
DiffSumObs, 3
DiffSumWithNA, 3
FWE, 4
HarmonicWtdMean, 5
HoggAdapt, 6
KS, 7
LogRank, 7
MinimumCF, 8
NormalCF, 9
NPC, 6, 9
open01, 12
ProductCF, 13
RankSum, 14
RankSumWithNA, 14
StudentsT, 15
StudentWilcoxon, 16
t2p, 17