Package ‘MultNonParam’

October 21, 2016

Type Package
Title Multivariate Nonparametric Methods
Version 1.2.5
Date 2016-10-20
Author John E. Kolassa & Stephane Jankowski
Maintainer John E. Kolassa <kolassa@stat.rutgers.edu>
Description A collection of multivariate nonparametric methods, selected in
part to support an MS level course in nonparametric statistical methods. Methods
include adjustments for multiple comparisons, implementation of multivariate
Mann-Whitney-Wilcoxon testing, inversion of these tests to produce a confidence
region, some permutation tests for linear models, and some algorithms for
calculating exact probabilities associated with one- and two- stage testing
involving Mann-Whitney-Wilcoxon statistics.
License GPL-2
RoxygenNote 5.0.1
NeedsCompilation yes
Repository CRAN
Date/Publication 2016-10-21 03:52:54

R topics documented:

  MultNonParam-package ........................................ 2
  aov.P ......................................................... 3
  betatest .................................................... 4
  dmannwhitney ............................................... 5
  higgins.fisher.kruskal.test .................................. 5
  page.test.unbalanced ....................................... 6
  probest .................................................... 7
  prostate.data ............................................... 8
  sensitivity.plot ............................................ 9
  sotiriou.data .............................................. 10
  symscorestat .............................................. 11
Description

A collection of multivariate nonparametric methods.

Details

Package: MultNonParam
Type: Package
Version: 1.0
Date: 2014-06-03
License: GPL-2

Author(s)

John E. Kolassa & Stephane Jankowski
Maintainer: John E. Kolassa <kolassa@stat.rutgers.edu>

References


understanding the molecular basis of histologic grade to improve prognosis, Journal of the National Cancer Institute 98 262-72.
G. E. Wilding, G. Shan and A. D. Hutson (2010), Exact Two-Stage Designs for Phase II Clinical Trials with Rank-Based Endpoints.

---

aov.P  
One-way ANOVA using permutation tests

Description

Aov.P uses permutation tests instead of classic theory tests to run a one-way or two-way ANOVA.

Usage

aov.P(dattab,permi,be)

Arguments

dattab : The table on which the ANOVA has to be done, or a vector of responses.
permi : If dattab is a table, ignored. If dattab is a vector, a vector of treatment labels.
be : If dattab is a table, ignored. If dattab is a vector, a vector of end points of blocks. In this case, blocks must form contiguous subvectors of dattab. If null, no blocking.

Details

The function calls a Fortran code to perform the permutation tests and the ANOVA. The function has to be applied directly on a cross-table of two variables.

Value

pv : the p-value obtained with the permutation tests.
tot

Author(s)

John E. Kolassa

References

Examples

```R
# Building an array for the example. The data is arbitrary.
dattab <- array(c(1.5, 1.8, 1.9, 2.1, 2.0, 2.5, 1.9, 2.0, 2.5,
                 2.8, 2.7, 2.6, 1.4, 1.6, 2.1, 1.8, 2.3, 2.4),
                 dim = c(3, 6))
aov.P(dattab)
```

Description

Calculate the p-value for the test of association between two variables using the permutation method.

Usage

```R
betatest(x, y)
```

Arguments

- `x`, `y` vectors for the two variables to test.

Details

Based on a Fortran routine.

Value

Returns a p-value.

Author(s)

John E. Kolassa

Examples

```R
# Example using data from plant Qn1 from the CO2 data set.
betatest(CO2$CO2$Plant == "Qn1", 4), CO2$CO2$Plant == "Qn1", 5))
```
**dmannwhitney**

*Probabilities associated with the Mann-Whitney test*

**Description**

The function estimates the probability associated with Mann-Whitney test.

**Usage**

```
dmannwhitney(u, m, n)
```

**Arguments**

- `u`  
  Mann-Whitney statistic
- `m`  
  number of observation in the first group
- `n`  
  number of observation in the second group

**Author(s)**

John E. Kolassa

**References**


---

**higgins.fisher.kruskal.test**

*Fisher’s Least Significant Difference (LSD) for Kruskal-Wallis test*

**Description**

Rank-based method for controlling experiment-wise error. Two hypothesis have to be respected: normality of the distribution and no ties in the data. The aim is to be able to detect, among k treatments, those who lead to significant differences in the values for a variable of interest.

**Usage**

```
higgins.fisher.kruskal.test(resp, grp, alpha=.05)
```

**Arguments**

- `resp`  
  vector containing the values for the variable of interest.
- `grp`  
  vector specifying in which group is each observation.
- `alpha`  
  level of the test.
Details
First, the Kruskal-Wallis test is used to test the equality of the distributions of each treatment. If the test is significant at the level \( \alpha \), the method can be applied.

Value
A matrix with two columns. Each row indicates a combination of two groups that have significantly different distributions.

Author(s)
John E. Kolassa

References

Examples
```
# Randomly generated data
resp <- c(rnorm(15,0,1), rnorm(15,50,1), rnorm(15,50,1))
grp <- c(rep(1,15), rep(2,15), rep(3,15))
higgins.fisher.kruskal.test(resp, grp)
# We expect to have 1-3 and 1-2 significantly different
# and not 2-3 since they are generated the same way (so same distribution).
```

---

**page.test.unbalanced**

Perform Page test for unbalanced two-way design

**Description**
Perform Page test for unbalanced two-way design

**Usage**
`page.test.unbalanced(x, trt, blk, sides = 2)`

**Arguments**
- `x` A vector of responses
- `trt` A vector of consecutive integers starting at 1 indicating treatment
- `blk` A vector of consecutive integers starting at 1 indicating block
- `sides` A single integer indicating sides. Defaults to 2.

**Value**
P-value for Page test.
Examples

```
pgtest.unbalanced(rnorm(15), rep(1:3,5), rep(1:5, rep(3,5)))
```

---

**Stratified Multivariate Kawaguchi Koch Wang Estimators**

**Description**

Function that return the estimators and their variance-covariance matrix calculated with the Kawaguchi - Koch - Wang method.

**Usage**

```
probest(ds, resp, grp, str = NULL, covs = NULL, delta = NA, correct = FALSE)
```

**Arguments**

- `ds`: The data.frame to be used.
- `resp`: The vector of the response manifest variable. There can be more than one variable. It has to be the name of the variable as a character string.
- `grp`: The vector of the variable that divides the population into groups. It has to be the name of the variable as a character string.
- `str`: The vector of the variable used for the strata. It has to be the name of the variable as a character string.
- `covs`: The covariates to be used in the model. It has to be the name of the variable as a character string. Default value is `NULL`
- `delta`: Offset for covariates. Default value is `NA`
- `correct`: Should the variance estimator be corrected as in Chen and Kolassa? Default value is `FALSE`.

**Details**

The function calls a Fortran code to calculate the estimators $b$ and their variance-covariance matrix $V_b$.

**Value**

- `b`: vector of adjusted estimates from the method
- `Vb`: the corresponding estimated covariance matrix

**Author(s)**

John E. Kolassa
References


Examples

```r
## We use here the data about breast cancer from the MultNonParam package.
data(sotiriou.data)
attach(sotiriou.data)

## First simple plot of the data
plot(AGE,TUMOR_SIZE,pch=(recur+1),
     main="Age and Tumor Size",
     sub="Breast Cancer Recurrence Data",
     xlab="Age (years)",ylab="Tumor Size",col=c("blue","darkolivegreen"))
legend(31,8,legend=c("Not Recurrent","Recurrent"),pch=1:2,col=c("blue","darkolivegreen"))

## AGE and TUMOR_SIZE are the response variables, recur is used for the groups,
## TAMOXIFEN_TREATMENT for the stratum and ELSTON.ELLIS GRADE is a covariable.
poc=probest(sotiriou.data,
    c("AGE","TUMOR_SIZE"),
    "recur",
    "TAMOXIFEN_TREATMENT",
    "ELSTON.ELLIS GRADE")
```

---

**prostate.data**  
Prostate cancer data set

**Description**

221 prostate cancer patients are collected in this data set.

**Usage**

```r
prostate.data
```

**Format**

- **hosp**: Hospital in which the patient is hospitalized.
- **stage**: stage of the cancer.
- **gleason score**: used to help evaluate the prognosis of the cancer.
- **psa**: prostate-specific antigen.
- **age**: age of the patient.
- **advanced**: boolean. TRUE if the cancer is advanced.
sensitivity.plot

Source

Examples

data(prostate.data)
attach(prostate.data)
plot(age,psa,
   main="Age and PSA",
   sub="Prostate Cancer Data",
   xlab="Age (years)",ylab="PSA")

Description
Compare the sensitivity of different statistics.

Usage
sensitivity.plot(y,sub,stats)

Arguments

  y         vector of the data.
  sub       subtitle for the plot.
  stats     vector of functions to be plotted.

Details
To compare the sensitivity, outliers are added to the original data. The shift of each statistics due to the new value is measured and plotted.

Author(s)
John E. Kolassa
**Examples**

```r
location <- function(y)(out <- c(mean(y), median(y), mean(y, trim = .1)))
names(out) <- c("mean", "median", "trim mean"); return(out)

# Draw a sensitivity plot comparing the performance of mean, median, 
# and trimmed mean for the DNase rat DNA density data. The data set 
# is in the R datasets package.

sensitivity.plot(DNase[DNase$Run == 1,3], "DNase Run 1", location)
```

---

**sotiriou.data**

 **Breast cancer data set**

---

**Description**

187 breast cancer patients are collected in this data set.

**Usage**

```r
sotiriou.data
```

**Format**

- **AGE**: Age of the patient
- **TUMOR_SIZE**: The size of the tumor, numeric variable
- **recur**: 1 if the patient has a recurrent breast cancer, 0 if it is not recurrent.
- **ELSTON.ELLIS GRADE**: Elston Ellis grading system in order to classify the breast cancers. It can be a low, intermediate or high grade (high being the worst prognosis)
- **TAMOXIFEN_TREATMENT**: boolean. `TRUE` if the patient is treated with the Tamoxifen treatment.

**Source**


**Examples**

```r
data(sotiriou.data)
attach(sotiriou.data)
plot(AGE, TUMOR_SIZE, pch = (recur + 1), 
    main = "Age and Tumor Size", 
    sub = "Breast Cancer Recurrence Data")
```
**Description**

`symscorestat` returns either exact or asymptotic p-values for score tests of the null hypothesis of univariate symmetry about 0.

**Usage**

```r
symscorestat(y, scores, exact, sides)
```

**Arguments**

- `y`: Vector of data on which test will be run.
- `scores`: Scores to be used for the test. Defaults to integers 1:length(y).
- `exact`: Logical variable indicating whether the exact p-value should be calculate. Default is false.
- `sides`: Integer; 1 for one sided test rejecting for large values of the statistic, and 2 for the two-sided test. Defaults to 1.

**Details**

The statistic considered here is the sum of scores corresponding to those entries in `y` that are positive. If `exact=T`, the function calls a Fortran code to cycle through all permutations. If `exact=F`, the expectation of the statistic is calculated as half the sum of the scores, the variance is calculated as one quarter the sum of squares of scores about their mean, and the statistic is compared to its approximating normal distribution.

**Value**

- `pv`: the p-value obtained with the permutation tests.
- `tot`: the total number of rearrangements of the data considered in calculating the p-value.

**Author(s)**

John E. Kolassa

**References**

Diagnosis for multivariate stratified Kawaguchi - Koch - Wang method

Description

Diagnostic tool that verifies the normality of the estimates of the probabilities $b$ with the Kawaguchi - Koch - Wang method. The diagnostic method is based on a Monte Carlo method.

Usage

testve(n, m, k, nsamp = 100, delta = 0, beta = 0, disc = 0)

Arguments

- **n**: number of observations in the first group.
- **m**: number of observations in the second group.
- **k**: number of stratum.
- **nsamp**: The number of estimates that will be calculated. Must be enough to be sure that the results are interpretable. Default value is 10000.
- **delta**: Offset that depends on group.
- **beta**: Correlation between $x$ and $y$.
- **disc**: The Mann Whitney test is designed to handle continuous data, but this method applies to discretized data; disc adjusts the discreteness.

Details

This function serves as a diagnosis to prove that the Kawaguchi - Koch - Wang method gives Gaussian estimates for $b$. It generates random data sets, to which the Mann Whitney test gets applied. $y$ is the generated response variable and $x$ the generated covariable related to $y$ through a regression model.

Value

The function returns a normal QQ plot.

Author(s)

John E. Kolassa
References

A. Kawaguchi, G. G. Koch and X. Wang (2012), *Stratified Multivariate Mann-Whitney Estimators for the Comparison of Two Treatments with Randomization Based Covariance Adjustment*.  

See Also

`probest` is the function used to calculate the estimates with the Kawaguchi - Koch - Wang method.

Examples

```r
#Dimensions of the data
n <- 10
m <- 15
k = 3
nsamp <- 100
testve(n,m,k,nsamp, beta=0.4)
```

**tukey.kruskal.test**  
*Tukey HSD procedure*

Description

Rank-based method for controlling experiment-wise error. We make the hypothesis of the normality of the distribution for the variable of interest.

Usage

```r
tukey.kruskal.test(resp, grp, alpha=.05)
```

Arguments

- `resp`: vector containing the values for the variable of interest.
- `grp`: vector specifying in which group is each observation.
- `alpha`: level of the test.

Details

The original Tuckey HSD procedure is supposed to be applied for equal sample sizes. However, the `tukey.kruskal.test` function performs the Tukey-Kramer procedure that works for unequal sample sizes.

Value

A logical vector for every combination of two groups. TRUE if the distribution in one group is significantly different from the distribution in the other group.
**Author(s)**

John E. Kolassa

**References**


**Examples**

```r
#Example using data from plant Qn1 from the CO2 data set.
tukey.kruskal.test(DNase[,3],DNase[,1])
#Here no group seem to have a different distribution from another one.
```

---

**util.jplot**  
*Smoothered plot*

**Description**

Graphical function that takes a noisy version of a piecewise constant function and plots a smoothered version.

**Usage**

```
util.jplot(x, y, ...)
```

**Arguments**

- `x, y`  
  Vectors of ordinate and abscissa for the points to be plotted.
- `...`  
  R graphical options to be added.

**Value**

A plot where steps are only represented by segments instead of noisy scattered points.

**Author(s)**

John E. Kolassa

**Examples**

```r
##Randomly generated data is used.
eps <- rnorm(300, 0,0.001)
#eps is added in order to have slightly different values of y for each point.
y <- c(rep(1,100),rep(2,110),rep(3,90))+eps
x <- seq(from=0,to=1, length.out=300)
```
### Graphs
par(mfrow=c(2,1))
# Normal plot of x by y
plot(x,y)
# util.jplot of x by y
util.jplot(x,y)

---

**wilding**  

*Two-stage Mann-Whitney test*

**Description**

Function that implements a two-stage Mann-Whitney test as described in the G. E. Wilding, G. Shan and A. D. Hutson paper (see references).

**Usage**

wilding(u1, u2, m1, n1, m2, n2)

**Arguments**

- `u1`, `u2`: The asymptotic null distribution of first and second stage test statistics.
- `n1`, `n2`: Number of Y observations for stage 1 and 2.
- `m1`, `m2`: Number of X observations for stage 1 and 2.

**Value**

The exact two-stage Mann-Whitney probability distribution.

**Author(s)**

John E. Kolassa

**References**

G. E. Wilding, G. Shan and A. D. Hutson (2010), *Exact Two-Stage Designs for Phase II Clinical Trials with Rank-Based Endpoints*.

Examples

```r
df.calcium <- data.frame(x1=c(1206,1202,1202,1201),
                        y1=c(1124, 1133, 1139, 1133),
                        x2=c(1017, 1017, 1012, 1020),
                        y2=c(965, 961, 960, 962))
attach(df.calcium)
u1 <- wilcox.test(x1,y1,exact=FALSE)$statistic
u2 <- wilcox.test(c(x1,x2),c(y1,y2),exact=FALSE)$statistic
wilding(u1[[1]],u2[[1]],length(x1),length(y1),length(x2), length(y2))
```
Index

*Topic package
   MultNonParam-package, 2

aov.P, 3
betatest, 4
dmannwhitney, 5
higgins.fisher.kruskal.test, 5
MultNonParam (MultNonParam-package), 2
MultNonParam-package, 2
page.test.unbalanced, 6
probest, 7, 13
prostate.data, 8
sensitivity.plot, 9
sotiriou.data, 10
symscorestat, 11
testve, 12
tukey.kruskal.test, 13
util.jplot, 14
wilding, 15