Package ‘treeperm’

April 22, 2015

Type Package

Title Exact and Asymptotic K Sample Permutation Test

Version 1.6

Date 2015-04-22

Author Qiao Kang <qk@st-andrews.ac.uk>

Maintainer Qiao Kang <qk@st-andrews.ac.uk>

Depends R (>= 2.14.1)

Description
An implementation of permutation tests in R, supporting both exact and asymptotic K sample test of data locations. The p value of exact tests is found using tree algorithms. Tree algorithms treat permutations of input data as tree nodes and perform constraint depth-first searches for permutations that fall into the critical region of a test systematically. Pruning of tree search and optimisations at C level enable exact tests for certain large data sets.

License GPL-2

NeedsCompilation yes

Repository CRAN

Date/Publication 2015-04-22 22:49:55

R topics documented:

`treeperm-package` *Exact and Asymptotic K Sample Permutation Test*

Description

A package that performs exact K sample permutation test of data locations

Details

Package:	treeperm
Type:	Package
Version:	1.6
Date:	2015-04-22
License:	GPL-2

An implementation of permutation tests in R, supporting both exact and asymptotic K sample test of data locations. The p value of exact tests is found using tree algorithms. Tree algorithms treat permutations of input data as tree nodes and perform constraint depth-first searches for permutations that fall into the critical region of a test systematically. Pruning of tree search and optimisations at C level enable exact tests for certain large data sets.

Author(s)

Qiao Kang

Maintainer: Qiao <qk@st-andrews.ac.uk>

See Also

Permutation KPermutation plot.treeperm treeperm

Examples

```
value<-c(0,190,0,0,10,0,0,0,0,0,0,110,0,0,52,0,8,0,50,0,0,137,965,110)
label<-as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2))
data<-data.frame(label,value)
exact<-treeperm(value~label,frame=data,type="exact")
exact
summary(exact)
approximate<-treeperm(value~label,frame=data,type="approximate",size=999)
approximate
summary(approximate)
plot(exact,size=999)
```

KPermutation *K Sample Exact Permutation Test*

Description

This function performs a K sample permutation test of data locations by returning a p value. It is a wrapper function for C level computations. Although it supports two sample permutation test, users should really use the Permutation function because it is much faster in handling two sample case. Users are not expected to use this function directly because there is no error checking mechanisms. However, for purpose of performance testing and other demands, this function is visible in this package.

Usage

```
KPermutation(data, factor)
## S3 method for class 'KPermutation'
print(x,...)
```

Arguments

`x`	The KPermutation class to be printed.
`data`	The set of all data to be permuted
`factor`	The set of indices, indicating group assignments of values in vector data. This parameter should be in factor type.
`...`	Reserved for future.

Details

Details about implementations are located in KPermutation.c.

Value

`Statistics`	The observed test statistics for input data. F statistics is used in this case.
`Permutations`	The total number of permutations can be performed on input groups.
`pvalue`	The p value of this two sample permutation test.

Note

Although the return value contains F statistics, the actual computation uses a reduced statistics to speed up computation.

Author(s)

Qiao Kang

References

Ernst M. D 2004, 'Permutation Methods: A Basis for Exact Inference', Statistical Science, Vol. 19, No. 4, p 676-685

Examples

```
value<-c(0,190,0,0,10,0,0,0,0,0,0,110,0,0,52,0,8,0,50,0,0,137,965,110)
label<-as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2))
print(KPermutation(value,label))
function (data, factor)
{
    factor <- as.factor(factor)
    p <- .Call("calculate_K_pvalue", data, as.integer(factor))
    result <- list(Fstatistics = p[3], Permutations = p[2], pvalue = p[1])
    class(result) <- "KPermutation"
    result
  }
```

`Permutation` *Two Sample Exact Permutation Test*

Description

This function performs a two sample permutation test of data locations by returning a p value. It is a wrapper function for C level computations. Users are not expected to use this function directly because there is no error checking mechanisms. However, for purpose of performance testing and other demands, this function is visible in this package.

Usage

```
Permutation(data, factor)
## S3 method for class 'Permutation'
print(x,...)
```

Arguments

`x`	The Permutation class to be printed.
`data`	The set of all data to be permuted
`factor`	The set of indices, indicating group assignments of values in vector data. This parameter should be in factor type..
`...`	Reserved for future.

Details

Details about implementations are located in Permutation.c.

Value

`Statistics` The observed test statistics for input data. It is the sum of all elements in first group.

`Permutations` The total number of permutations can be performed on input groups.

`pvalue` The p value of this two sample permutation test.

Note

Although the return value uses a group mean value as test statistics, the actual computation uses the group sum to speed up computation.

Author(s)

Qiao Kang

References

Fisher, R. A. 1925, 'Statistical Methods for Research Workers', Oliver and Boyd, Edinburgh

Examples

```
value<-c(0,190,0,0,10,0,0,0,0,0,0,110,0,0,52,0,8,0,50,0,0,137,965,110)
label<-as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2))
print(Permutation(value,label))
```

`plot.treeperm` *Plot Permutations*

Description

This function use large sample approximation to visualize a permutation class after user performed a treeperm test.

Usage

```
## S3 method for class 'treeperm'
plot(x, size, ...)
```

Arguments

`x` The treeperm class to be visualized

`size` The size of simulated distribution

`...` Not necessarily required, this is reserved for future extensions.

Details

This function is independent of the type of permutation (either exact or approximate) contained in the treeperm object. It will use the stored data and factor in treeperm create an asymptotic permutation distribution and plot that distribution.

Value

A histogram for visualization.

Note

This function simulates a distribution for a given distribution. Please set the random seed in advance.

Author(s)

Qiao Kang

See Also

treeperm

Examples

```
value<-c(0,190,0,0,10,0,0,0,0,0,0,110,0,0,52,0,8,0,50,0,0,137,965,110)
label<-as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2))
data<-data.frame(label,value)
exact<-treeperm(value~label,frame=data,type="exact")
plot(exact,size=999)
function (x, size, ...)
{
    ran <- GetDistribution(x, size)
    r <- hist(ran, breaks = 50, main = "Permutations estimated by Monto carlo method",
        xlab = "F  statistics", ylab = "Frequency", col = "lightblue")
    top <- max(r$counts)
    points(x$Fstatistics, -top/150, type = "p", pch = 17, col = "red",
        lwd = 3)
    text(x$Fstatistics, -top/50, cex = 0.65, "Observed F statistics")
  }
```

treeperm *Permutation Tests Using Tree Algorithms*

Description

This function will perform a K samnple permutation test of data locations. Users may choose either asymptotic test or exact test.

Usage

```
## S3 method for class 'formula'
treeperm(formula, frame = list(), type, size, ...)
## Default S3 method:
treeperm(x,data,factor, type, size, ...)
## S3 method for class 'treeperm'
print(x,...)
## S3 method for class 'treeperm'
summary(object,...)
```

Arguments

`x`	Generic argument, used only in print.treeperm to represent the treeperm class.
`formula`	The formula to perform permutation test. The form should be value~label, where value is the set of data and label classifies value into groups.
`frame`	The dataframe storing vectors used in formula.
`data`	The data set to be permuted
`factor`	The index set that classifies parameter data into groups.
`type`	Which type of permutation the user want to use, either 'exact' or 'approximate'.
`size`	If the type is 'approximate', user must specify the size of permutations they want.
`object`	The treeperm class to be summarised.
`...`	Reserved for future.

Details

This function calls KPermutation or Permutation depending on the levels of factor.

Value

`result`	A permutation object returned by a permutation function, either class Permutation or class KPermutation depending the levels of argument factor.
`call`	User's calling script to this function at highest level.
`data`	The input data set.
`factor`	The index set that classifies data into groups.

Author(s)

Qiao

See Also

Permutation KPermutation

Examples

```
value<-c(0,190,0,0,10,0,0,0,0,0,0,110,0,0,52,0,8,0,50,0,0,137,965,110)
label<-as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2,1,1,1,2,2,2))
data<-data.frame(label,value)
exact<-treeperm(value~label,frame=data,type="exact")
summary(exact)
approximate<-treeperm(value~label,frame=data,type="approximate",size=999)
summary(approximate)
```

Index