Package ‘mcStats’

December 3, 2019

Title Visualize Results of Statistical Hypothesis Tests
Version 0.1.1
Maintainer Michael Czekanski <mczekanski@middlebury.edu>
Description Provides functionality to produce graphs of sampling distributions of test statistics from a variety of common statistical tests. With only a few keystrokes, the user can conduct a hypothesis test and visualize the test statistic and corresponding p-value through the shading of its sampling distribution. Initially created for statistics at Middlebury College.
Depends R (>= 3.4.0)
License GPL-3
Encoding UTF-8
LazyData true
RoxygenNote 6.1.1
Imports dplyr, ggplot2, ggthemes, gridExtra, magrittr, rlang, stats, tidy
Suggests testthat
NeedsCompilation no
Author Michael Czekanski [aut, cre],
    Alex Lyford [aut]
Repository CRAN
Date/Publication 2019-12-03 17:00:02 UTC

R topics documented:

  bootstrap .......................................................... 2
  hello ............................................................. 3
  labelBootResults ............................................... 3
  labelPDFDis ...................................................... 4
  mcDChiSq ......................................................... 4
  mcDF ............................................................ 5
  mcDNorm ........................................................ 5
  mcDT ............................................................. 6
Description

Bootstrap using given data and statistic

Usage

bootstrap(fun, data, h0, nreps, conf.level = 0.95, verbose = 1)

Arguments

fun  
function to calculate on each sample. This can be a user-defined function that takes in data as a vector and returns a statistic.
data  
data to use for bootstrapping. Should be a representative sample
h0    
null hypothesis value
nreps  
number of times to bootstrap
conf.level  
confidence value
verbose  
default is 1 which will create a graph. To turn this off use verbose = 0.

Value

results from bootstrapping. A vector of length @param nreps containing each statistic calculated

Examples

x <- rnorm(100)
bootstrap(mean, x, 0.5, 1000, verbose = 0)
bootstrap(mean, x, 0.5, 1000)
Print "hello world!"

Description
print "hello world!"

Usage
hello()

Examples
hello()

classBootResults Label Bootstrapped Results

Description
labels bootstrapped results. We use this to create colored histograms.

Usage
labelBootResults(results, lBound, uBound)

Arguments
results a vector, data from bootstrapping
lBound lower bound of confidence interval
uBound upper bound of confidence interval

Value
vector of labels corresponding to result values

Examples
x <- rnorm(100)
labelBootResults(x, -1, 1)
**labelPDFDis**  
*Label discrete PDF*

**Description**
labels a discrete pdf

**Usage**
`labelPDFDis(x, obsVal, expVal)`

**Arguments**
- `x`  
  x value
- `obsVal`  
  observed event
- `expVal`  
  expected value

**Value**
vector of labels for x value in relation to observed event

**Examples**
`labelPDFDis(0:10, 3, 5)`

---

**mcDChiSq**  
*Density of Chi-Square distribution*

**Description**
Density of Chi-Square distribution

**Usage**
`mcDChiSq(x, degFree, ...)`

**Arguments**
- `x`  
  x value
- `degFree`  
  degrees of freedom
- `...`  
  optional additional parameters which are ignored

**Value**
density of given Chi-Square dist. at x
**mcDF**

*Density of F-distribution*

**Description**
Density of F-distribution

**Usage**
mcDF(x, degFree1, degFree2, ...)

**Arguments**
- `x` : x value
- `degFree1` : degrees of freedom 1
- `degFree2` : degrees of freedom 2
- `...` : optional additional parameters which are ignored

**Value**
density of given F-dist. at x

---

**mcDNorm**

*dnorm but with more arguments*

**Description**
compute density of normal distribution while allowing for more arguments which are ignored

**Usage**
mcDNorm(x, mean = 0, sd = 1, log = FALSE, ...)

**Arguments**
- `x` : x value
- `mean` : mean of normal distribution
- `sd` : std. dev. of normal distribution
- `log` : logical; if TRUE probabilities are given as log(p). See stats::dnorm
- `...` : extra parameters which are ignored

**Value**
density of normal distribution
mcDT  

**Density of t-distribution**

**Description**
Density of t-distribution

**Usage**
mcDT(x, degFree, ...)

**Arguments**
- x  
  x value
- degFree  
  degrees of freedom
- ...  
  optional additional parameters which are ignored

**Value**

density of given t-dist. at x

---

shadePDFCts  

**Used to shade in a PDF**

**Description**
Returns density with extreme event region having NAs

**Usage**
shadePDFCts(x, fun, testStat, ...)

**Arguments**
- x  
  x value
- fun  
  density function to use
- testStat  
  test statistic value
- ...  
  optional parameters passed to density function

**Value**
density if outside of extreme event region
showANOVA

Show results of ANOVA

Description

Visualization of distributional results of ANOVA. Please see `aov` for more information on parameters.

Usage

```r
showANOVA(formula, data = NULL, verbose = 1, ...)
```

Arguments

- `formula`: formula specifying a model.
- `data`: data on which to perform ANOVA
- `verbose`: if verbose > 0 the resulting graph is printed
- `...`: Arguments passed to `lm`. See `aov` for more detail

Value

output of call to `aov`

Examples

```r
showANOVA(yield ~ N + P + K, npk)
```

showChiSq.Test

Show Chi-Square Test

Description

show results of a chi-square test visually using `chisq.test`

Usage

```r
showChiSq.Test(x, y = NULL, p = rep(1/length(x), length(x)), simulate.p.value = FALSE, nreps = 2000, verbose = 1)
```
showMcNemarTest

Arguments

- **x**: a numeric vector or matrix. x can also be factors.
- **y**: a numeric vector.
- **p**: a vector of probabilities the same length as x. Used for goodness-of-fit tests. Must be a valid distribution.
- **simulate.p.value**: boolean, if TRUE use simulation to estimate p-value.
- **nreps**: if simulate.p.value = TRUE number of simulations to complete.
- **verbose**: level of visual output, 0 = silent.

Value

results of chisq.test call

Examples

showChiSq.Test(x = c(1,2,1), y= c(1,2,2))

showMcNemarTest

Visualize results of McNemar’s Test

Description

relevant parameters are passed to mcnemar.test

Usage

showMcNemarTest(x, y = NULL, correct = TRUE, verbose = 1)

Arguments

- **x**: two dimensional contingency table as a matrix or a factor object.
- **y**: factor object, ignored if x is a matrix.
- **correct**: logical indicating whether or not to perform continuity correction.
- **verbose**: if verbose > 0 the resulting graph is printed.

Value

results of call to mcnemar.test
showMosaicPlot

Mosaic Plot

Description
Mosaic Plot

Usage
showMosaicPlot(x)

Arguments
x
must be a matrix with each row and column labelled

Value
mosaic plot showing observed proportions, colored by residuals from chi-sq. test

Examples
x <- matrix(runif(9,5,100), ncol = 3, dimnames = list(c("Yes1", "No1", "Maybe1"), c("Yes2", "No2", "Maybe2")))
showMosaicPlot(x)

showOLS

Show hypothesis tests from OLS

Description
Show hypothesis tests from OLS

Usage
showOLS(formula, data, verbose = 1)

Arguments
formula
forumula for regression. Passed to lm
data
data for regression. Passed to lm
verbose
if verbose > 0 the resulting graph is printed

Value
model object resulting from the regression
Examples

showOLS(mpg ~ cyl + disp, mtcars)

showProp.Test(x, n, p = 0.5)

Description

Show results of proportion test using binom.test

Usage

showProp.Test(x, n, p = 0.5)

Arguments

x: x value
n: number of repetitions
p: probability of success in one Bernoulli trial

Value

output of call to binom.test

Examples

showProp.Test(3, 10)

showT.Test(group1, group2 = NULL, mu = 0, paired = FALSE, verbose = 1)

Description

Runs z-test and outputs graph for interpretation using stats::t.test

Usage

showT.Test(group1, group2 = NULL, mu = 0, paired = FALSE, verbose = 1)

Arguments

group1: continuous data to test
group2: optional: second group to include for two sample t-test
mu: optional: mean to test against for one-sample t-test
paired: boolean, if TRUE perform matched pairs t-test
verbose: default is 1 which will create a graph. To turn this off use verbose = 0.
**showXtremeEventsCts**

**Value**

results of call to t.test

**Examples**

```r
x <- rnorm(100)
showT.Test(x, verbose = 0)
showT.Test(x)
```

---

**showXtremeEventsCts** *Highlight extreme events*

**Description**

Make graph highlighting events more extreme than observed sample

**Usage**

```r
showXtremeEventsCts(testID, testStat, densFun, degFree = NULL,
                      degFree1 = NULL, degFree2 = NULL, xlims, verbose = 1, ...)
```

**Arguments**

- `testID` name of hypothesis test
- `testStat` test statistic
- `densFun` function that computes appropriate density
- `degFree` degrees of freedom when only one is needed. This gets passed into densFun
- `degFree1` first degrees of freedom parameter when more than one is needed
- `degFree2` second degrees of freedom parameter when more than one is needed
- `xlims` x limits of the graph to be used. This is passed to ggplot
- `verbose` if verbose > 0 the resulting graph is printed
- `...` extra arguments passed to density function

**Value**

results of call testFun

**Examples**

```r
x <- rnorm(100)
showT.Test(x, verbose = 0)
showT.Test(x)
```
showXtremeEventsDis   Show Extreme Events from a Discrete Distribution

Description

Show Extreme Events from a Discrete Distribution

Usage

showXtremeEventsDis(testID, obsVal, expVal, xVals, probFun, ...)

Arguments

testID          name of test being performed. This is used to title the graph
obsVal          observed x value
expVal          expected x value
xVals           domain of x (possible values)
probFun         probability mass function for the given distribution
...             addition arguments passed to probFun

Value

graph coloring events by how extreme they are under the null hypothesis

Examples

showXtremeEventsDis("Prop. Test", 3, 5, 0:10, probFun = dbinom, size = 10, prob = 0.5)
Index

aov, 7
binom.test, 10
bootstrap, 2
chisq.test, 7, 8
hello, 3
labelBootResults, 3
labelPDFDis, 4
lm, 9
mcDChiSq, 4
mcDF, 5
mcDNorm, 5
mcDT, 6
mcnemar.test, 8
shadePDFCts, 6
showANOVA, 7
showChiSq.Test, 7
showMcNemarTest, 8
showMosaicPlot, 9
showOLS, 9
showProp.Test, 10
showT.Test, 10
showXtremeEventsCts, 11
showXtremeEventsDis, 12