Package ‘CarletonStats’

February 19, 2015

Type Package
Title Functions For Statistics Classes At Carleton College
Version 1.1
Date 2014-07-03
Author Laura Chihara
Maintainer Laura Chihara <lchihara@carleton.edu>
Description Includes commands for bootstrapping and permutation tests, a command for created grouped bar plots, and a demo of the quantile-normal plot for data drawn from different distributions.
License GPL-2
LazyData TRUE
Suggests MASS
NeedsCompilation no
Repository CRAN
Date/Publication 2014-07-03 15:51:20

R topics documented:

Carleton-package .................................................. 2
anovaSummarized ............................................... 2
boot ................................................................. 3
bootCor ............................................................ 5
bootPaired ......................................................... 6
confIntDemo ...................................................... 7
corDemo .......................................................... 8
groupedBar ......................................................... 9
Icecream ............................................................ 10
Milkshakes .......................................................... 11
missingLevel ....................................................... 12
permTest ........................................................... 12
permTestCor ....................................................... 14
permTestPaired ..................................................... 15
Description

Custom functions written for students taking statistics at Carleton College. Functions for the bootstrap and permutation tests as well as a grouped barplot. There are also some demonstrations to illustrate the concepts of correlation, the normal quantile-quantile plot and confidence intervals.

Details

Package: Carleton
Type: Package
Version: 1.1
Date: 2014-06-09
License: GPL-2

Author(s)

Laura Chihara
Maintainer: Laura Chihara <lchihara@carleton.edu>

Description

ANOVA F test when given summarized data (sample sizes, means and standard deviations).

Usage

anovaSummarized(N, mn, stdev)

Arguments

N  a vector with the sample sizes
mn a vector of means, one for each group in the sample
stdev a vector of standard deviations, one for each group in the sample
Details

Perform an ANOVA F test when presented with summarized data: sample sizes, sample means and sample standard deviations.

Value

Returns invisibly a list

- **Treatment SS**: The treatment sum of squares (also called the "between sum of squares").
- **Residual SS**: Residual sum of squares (also called the "within sum of squares").
- **Degrees of Freedom**: a vector with the numerator and denominator degrees of freedom.
- **Treatment Mean Square**: Treatment SS/numerator DF
- **Residual Mean Square**: Residual SS/denominator DF
- **Residual Standard Error**: Square root of Residual Mean Square
- **F**: the F statistic
- **P-value**: p-value

... 

Author(s)

Laura Chihara

Examples

#use the data set chickwts from base R
head(chickwts)

N <- table(chickwts$feed)
stdev <- tapply(chickwts$weight, chickwts$feed, sd)
mm <- tapply(chickwts$weight, chickwts$feed, mean)

anovaSummarized(N, mm, stdev)

Description

Bootstrap a single variable or a grouped variable
Usage

boot(x, ...)
## Default S3 method:
boot(x, group = NULL, fun = mean, conf.level = 0.95,
    B = 10000, plot.hist = TRUE, hist.title = NULL, legend.loc = "topright",
    plot.qq = FALSE,...)
## S3 method for class 'formula'
boot(formula, data, subset, ...)

Arguments

x            a numeric vector
group        an optional grouping variable (vector), usually a factor variable. If it is a binary
              numeric variable, it will be coerced to a factor.
fun           function that computes the statistic of interest. Default is the mean.
conf.level   confidence level for the bootstrap percentile interval. Default is 95%.
B             number of times to resample (positive integer greater than 2).
plot.hist    logical value. If TRUE, plot the histogram of the bootstrap distribution.
hist.title   a title for the histogram.
legend.loc   where to place the legend on the histogram. Default is "topright". Other
              options include "topleft", "bottomleft" and "bottomright".
plot.qq      Logical value. If TRUE, create a normal quantile-quantile plot of the bootstrap
              distribution.
formula      a formula \( x \sim g \) where \( x \) is a numeric vector and \( g \) a factor
              variable with two levels. If \( g \) is a binary numeric vector, it will be coerced to
              a factor variable.
data         a data frame that contains the variables given in the formula.
subset       an optional expression indicating what observations to use.
...           further arguments to be passed to or from methods.

Details

Perform a bootstrap of a statistic applied to a single variable, or to the difference of the statistic
computed on two samples (using the grouping variable). If \( x \) is a binary vector of 0’s and 1’s and
the function is the mean, then the statistic of interest is the proportion.

Observations with missing values are removed.

Value

A vector with the resampled statistics is returned invisibly.

Author(s)

Laura Chihara
References

Tim Hesterberg’s website http://www.timhesterberg.net/bootstrap

Examples

# ToothGrowth data (supplied by R)
# bootstrap mean of a single numeric variable
boot(ToothGrowth$len)

# bootstrap difference in mean of tooth length for two groups.
boot(ToothGrowth$len, ToothGrowth$supp)

# same as above using formula syntax
boot(len ~ supp, data = ToothGrowth)

---

bootCor  

**Bootstrap a correlation**

**Description**

Bootstrap the correlation of two numeric variables.

**Usage**

```r
bootCor(x, y, conf.level = 0.95, B = 10000, plot.hist = TRUE, hist.title = NULL,
        plot.qq = FALSE, legend.loc = "topleft")
```

**Arguments**

- `x`  
a numeric vector.
- `y`  
a numeric vector.
- `conf.level`  
confidence level for the bootstrap percentile interval.
- `B`  
number of times to resample (positive integer greater than 2).
- `plot.hist`  
a logical value. If TRUE, plot the bootstrap distribution of the resampled correlation.
- `hist.title`  
a user supplied title for the histogram.
- `plot.qq`  
a logical value. If TRUE a normal quantile-quantile plot of the bootstrapped values is created.
- `legend.loc`  
location to place the legend. Options include "topright", "topleft", "bottomright", "bottomleft".

**Details**

Bootstrap the correlation of two numeric variables. The bootstrap mean and standard error are printed as well as a bootstrap percentile confidence interval.

Observations with missing values are removed.
value

The command invisibly returns the correlations of the resampled observations.

Author(s)

Laura Chihara

References

Tim Hesterberg’s website http://www.timhesterberg.net/bootstrap

Examples

plot(states$ColGrad, states$InfMortality)
bootCor(states$ColGrad, states$InfMortality)

bootPaired  Bootstrap paired data

Description

Perform a bootstrap of two paired variables.

Usage

bootPaired(x, y, fun = mean, conf.level = 0.95, B = 10000, plot.hist = TRUE,
        hist.title = NULL, plot.qq = FALSE, legend.loc = "topright")

Arguments

x  a numeric vector.
y  a numeric vector.
fun function for the statistic you wish to compute.
conf.level confidence level for the bootstrap percentile interval.
B  number of resamples (positive integer greater than 2).
plot.hist logical. If TRUE, plot the histogram of the bootstrap distribution.
hist.title supply your own title for the histogram.
plot.qq logical. If TRUE, a normal quantile-quantile plot of the replicates will be created.
legend.loc location for the legend on the histogram. Options are "topright" "topleft", "bottomleft" and "bottomright".
Details
The command will compute the difference of \( x \) and \( y \) and bootstrap the difference. The mean and standard error of the bootstrap distribution will be printed as well as a bootstrap percentile interval. Observations with missing values are removed.

Value
The command invisibly returns a vector with the replicates of the statistic being bootstrapped.

Author(s)
Laura Chihara

References
Tim Hesterberg’s website http://www.timhesterberg.net/bootstrap

Examples

```r
# Bootstrap the mean difference of fat content in vanilla and chocolate ice cream. Data are paired because ice cream from the same manufacturer will have similar content.
Icecream
bootPaired(Icecream$VanillaFat, Icecream$ChocFat)
```

## Description
Draw many random samples and compute confidence interval. How many intervals capture the true mean?

## Usage
```
confIntDemo(distr = "normal", size = 20, conf.level = 0.95)
```

## Arguments
- **distr** distribution of the population to be sampled. Options include "normal", "exponential", "uniform" and "binary" (partial match allowed).
- **size** sample size
- **conf.level** confidence level.
Details

This simulation will draw 100 random samples from a given population distribution and compute the corresponding confidence intervals. The 100 intervals will be drawn with an indication of the ones that missed the true mean. A histogram of the population will also be created.

Value

The command invisibly returns the fraction of intervals that capture the true mean.

Author(s)

Laura Chihara

Examples

```r
confIntDemo()
confIntDemo(distr = "exponential", size = 40)
```

---

**corDemo**

*Correlation demonstration*

Description

For a given \( r \), create a scatterplot of two variables with that correlation.

Usage

```r
corDemo(r = 0)
```

Arguments

\( r \) a number between -1 and 1. Enter any number \( r \), \textit{latex}, to exit the interactive session.

Details

Demonstrate the concept of correlation by inputting a number between -1 and 1 and seeing a scatter plot of two variables with that correlation. Once you invoke this command, you can continue to enter values for \( r \). Type any number \textit{latex}) to exit.

Author(s)

Laura Chihara
Examples

```r
## Not run:
cordemo()
## End(Not run)
```

### groupedBar

**Grouped bar chart**

#### Description

Create a bar chart of a single categorical variable or a grouped bar chart of two categorical variables.

#### Usage

```r
groupedBar(x, ...)  
```

**Default S3 method:**

```r
groupedBar(x, condvar = NULL, percent = TRUE, ylim = NULL, color = NULL,  
  cex.axis = NULL, cex.names = NULL, legend = TRUE, legend.loc = "topright",  
  inset = NULL, ...)
```

**S3 method for class 'formula'**

```r
groupedBar(formula, data = parent.frame(), subset, ...)
```

#### Arguments

- **x**
  - a factor variable. If x is numeric, it will be coerced to a factor variable.

- **condvar**
  - a factor variable to condition on. If NULL, then a bar plot of just the x variable will be created. If condvar is numeric, it will be coerced to a factor variable.

- **percent**
  - a logical value. Should the y-axis give percent or counts?

- **ylim**
  - set the y-axis range.

- **color**
  - optional if you don’t like the default colors for the bars. Type `colors()` to see the options.

- **cex.axis**
  - magnification for the y-axis labels.

- **cex.names**
  - magnification for the bar labels.

- **legend**
  - a logical value. If TRUE, include a legend.

- **legend.loc**
  - location of the legend. Options include "topright", "topleft", "bottomright" and "bottomleft".

- **inset**
  - inset distances from the margins as a fraction of the plot region when legend is placed by keyword. See `legend`.

- **formula**
  - a formula of the form `x ~ condvar`. If x or condvar is (are) not a factor variable, then it (they) will be coerced into one.

- **data**
  - a data frame that contains the variables in the formula.

- **subset**
  - an optional vector specifying a subset of observations to be used.

- **...**
  - further arguments to be passed to or from methods.
Details
For a single factor variable, a bar plot. If two factor variables are given, then a bar plot of x conditioned by condvar. This command uses R’s table command so missing values are automatically removed.

Value
Returns invisibly a table of the variable(s).

Author(s)
Laura Chihara

Examples

```r
groupedBar(states$DeathPenalty)

groupedBar(states$DeathPenalty, states$Region, legend.loc = "topleft")
#same as above but using a formula syntax
groupedBar(DeathPenalty ~ Region, data = states, legend.loc = "topleft")
```

---

**Icecream**  
*Ice cream data*

Description
Nutritional information on vanilla and chocolate ice cream from a sample of companies.

Usage
Icecream

Format
A data frame with 39 observations on the following 7 variables.

- **Brand**  Brand name
- **VanillaCalories**  Calories per serving in vanilla
- **VanillaFat**  Fat per serving (g) in vanilla
- **VanillaSugar**  Sugar per serving (g) in vanilla
- **ChocCalories**  Calories per serving in chocolate
- **ChocFat**  Fat per serving (g) in chocolate
- **ChocSugar**  Sugar per serving (g) in chocolate
Milkshakes

Details
Nutritional information on vanilla and chocolate ice cream from a sample of companies.

Source
Data collected by Carleton student Ann Butkowski (2008).

Examples

```
head(Icecream)
t.test(Icecream$VanillaCalories, Icecream$ChocCalories, paired = TRUE)
```

<table>
<thead>
<tr>
<th>Milkshakes</th>
<th>Milkshakes (chocolate)</th>
</tr>
</thead>
</table>

Description
Nutritional information on chocolate milkshakes from a sample of restaurants.

Usage
Milkshakes

Format
A data frame with 29 observations on the following 11 variables.

- Restaurant Names of restaurants
- Type Type of restaurant, Dine In Fast Food
- Calories Calories per serving
- Fat Fat per serving (g)
- Sodium Sodium per serving (mg)
- Carbs Carbohydrates per serving (g)
- SizeOunces Size of milkshake (ounces)
- CalPerOunce Calories per ounce
- FatPerOunce Fat per ounce
- CarbsPerOunce Carbohydrates per ounce

Details
Nutritional information on chocolate milkshakes from a sample of restaurants.

Source
Data collected by Carleton students Yoni Blumberg (2013) and Lindsay Guthrie (2013).
**missingLevel**  
*Mis** *s** *s** *i** *n** *g** * o** *b** *s** *s** *e** *r** *v** *a** *t** *i** *o** *n** *s** *i** *n** * f** *a** *c** *t** *o** *r** *s**

**Description**

In data frames with factor variables, convert any observation with "" into <NA>.

**Usage**

`missingLevel(data)`

**Arguments**

data a data frame with factor variables.

**Details**

In a factor variable with the level """, this command will convert this to an <NA>.

**Value**

Returns the same data frame with """" replaced by <NA> in factor variables.

**Note**

When importing data from comma separated files (for example), missing values in a categorical variable are often denoted by """". We often do not want to treat this as a level of a factor variable in R.

**Author(s)**

Laura Chihara

**permTest**  
*Pe** *r** *m** *u** *t** *a** *t** *i** *o** *n** * t** *e** *s** *t*

**Description**

Permutation test to test a hypothesis involving two samples.

**Usage**

`permTest(x, ...)`

## Default S3 method:
`permTest(x, group, fun = mean, B = 9999, alternative = "two.sided", 
plot.hist = TRUE, legend.loc = "topright", plot.qq = FALSE, ...)`

## S3 method for class 'formula'
`permTest(formula, data = parent.frame(), subset, ...)"
Arguments

- **x**: a numeric vector. If the function is the mean (`fun = mean`) and `x` is a binary numeric vector of 0’s and 1’s, then the test is between proportions.
- **group**: a factor variable with two levels. If `group` is a binary numeric vector, it will be coerced into a factor variable.
- **fun**: the statistic of interest.
- **B**: the number of resamples (positive integer greater than 2).
- **alternative**: the alternative hypothesis. Options are "two.sided", "less" or "greater".
- **plot.hist**: a logical value. If TRUE, the permutation distribution of the statistic is plotted.
- **legend.loc**: location of the legend for the histogram. Options are "topleft", "topright", "bottomleft" or "bottomright".
- **plot.qq**: a logical value. If TRUE, then a normal quantile-quantile plot of the resampled test statistic is created.
- **formula**: a formula of the form `x ~ group` where `x` is numeric and `group` is a factor variable.
- **data**: a data frame with the variables in the formula.
- **subset**: an optional expression specifying which observations to keep.
- **...**: further arguments to be passed to or from methods.

Details

Permutation test to see if a population parameter is the same for two populations. For instance, test \( \mu \) where \( \mu \) denotes the population mean. The values of the numeric variable are randomly assigned to the two groups and the difference of the statistic for each group is calculated. The command will print the mean and standard error of the distribution of the test statistic as well as a P-value.

Observations with missing values are removed.

Value

Returns invisibly a vector of the replicates of the test statistic.

Author(s)

Laura Chihara

References

Tim Hesterberg’s website: [http://www.timhesterberg.net/bootstrap](http://www.timhesterberg.net/bootstrap)

Examples

```r
permTest(states03$ViolentCrime, states03$DeathPenalty)
# using formula syntax
permTest(ViolentCrime ~ DeathPenalty, data = states03, alt = "less")
```
**Description**

Hypothesis test for a correlation of two variables. The null hypothesis is that the population correlation is 0.

**Usage**

```r
permTestCor(x, y, B = 999, alternative = "two.sided", plot.hist = TRUE,
             legend.loc = "topright", plot.qq = FALSE)
```

**Arguments**

- `x` a numeric vector.
- `y` a numeric vector.
- `B` the number of resamples to draw (positive integer greater than 2).
- `alternative` alternative hypothesis. Options are "two.sided", "less" or "greater".
- `plot.hist` a logical value. If TRUE, plot the distribution of the correlations obtained from each resample.
- `legend.loc` location of the legend on the histogram. Options are "topright", "topleft", "bottomleft" and "bottomright".
- `plot.qq` a logical value. If TRUE, plot the normal quantile-quantile plot of the correlations obtained from each resample.

**Details**

Perform a permutation test to test \( \rho \), where \( \rho \) is the population correlation. The rows of the second variable are permuted and the correlation is re-computed.

The mean and standard error of the permutation distribution is printed as well as a P-value.

Observations with missing values are removed.

**Value**

Returns invisibly a vector of the correlations obtained by the randomization.

**Author(s)**

Laura Chihara

**References**

Tim Hesterberg’s website: [http://www.timhesterberg.net/bootstrap](http://www.timhesterberg.net/bootstrap)
permTestPaired

**Examples**

```r
plot(states03$HSGrad, states03$TeenBirths)
cor(states03$HSGrad, states03$TeenBirths)
permTestCor(states03$HSGrad, states03$TeenBirths)
```

---

**Description**

Permutation test for paired data.

**Usage**

```r
permTestPaired(x, y, fun = mean, B = 9999, alternative = "two.sided",
               plot.hist = TRUE, legend.loc = "topright", plot.qq = FALSE)
```

**Arguments**

- `x`: a numeric vector.
- `y`: a numeric vector.
- `fun`: statistic of interest.
- `B`: the number of resamples.
- `alternative`: the alternative hypothesis. Options are "two.sided", "less" and "greater".
- `plot.hist`: a logical value. If TRUE, create a histogram displaying the permutation distribution of the statistic.
- `legend.loc`: a logical value. If TRUE, include a legend on the histogram.
- `plot.qq`: a logical value. If TRUE, include a quantile-normal plot of the permutation distribution.

**Details**

For two paired numeric variables with n rows, randomly select k of the n rows (k also is random) and switch the entries *latex* and then compute the statistic on the difference of the two variables. Observations with missing values are removed.

**Value**

Returns invisibly a vector of the replicates of the test statistic (ex. mean of the difference of the resampled variables).

**Author(s)**

Laura Chihara
References

Tim Hesterberg’s website: http://www.timhestemberg.net/bootstrap

Examples

#Does vanilla ice cream have fewer calories than chocolate ice cream, on average?
#H0: mean amount of sugar is the same
#H1: mean amount of sugar is less in vanilla ice cream

permTestPaired(Icecream$VanillaSugar, Icecream$Choc Sugar, alternative = "less")

---

qqPlotDemo

Demonstration of the normal qq-plot.

Description

Demonstrate the normal quantile-quantile plot for samples drawn from different populations.

Usage

qqPlotDemo(n = 25, distribution = "normal", mu = 0, sigma = 1, df = 10, lambda = 10,
numdf = 10, dendf = 16, shape1 = 40, shape2 = 5)

Arguments

- `n`: sample size
- `distribution`: population distribution. Options are "normal", "t", "exponential", "chi.square", "F" or "beta" (partial matches are accepted).
- `mu`: mean for the normal distribution.
- `sigma`: (positive) standard deviation for the normal distribution.
- `df`: (positive) degrees of freedom for the t-distribution.
- `lambda`: positive rate for the exponential distribution.
- `numdf`: (positive) numerator degrees of freedom for the chi-square distribution.
- `dendf`: (positive) denominator degrees of freedom for the chi-square distribution.
- `shape1`: positive parameter for the beta distribution (shape1 = a).
- `shape2`: positive parameter for the beta distribution (shape2 = b).

Details

Draw a random sample from the chosen sample and display the normal qq-plot as well as the histogram of its distribution.

Value

Returns invisibly the random sample.
Author(s)
Laura Chihara

Examples

`qqPlotDemo(n = 30, distr = "exponential", lambda = 1/3)`

---

**Description**

Census data on the 50 states from 2003.

**Usage**

`states03`

**Format**

A data frame with 50 observations on the following 24 variables.

- **State**: the 50 states
- **Region**: a factor with levels Midwest, Northeast, South, West
- **Pop**: Population in 1000
- **Births**: Number of births
- **Deaths**: Number of deaths
- **Pop18**: Percent of population 18 years of age or younger
- **Pop65**: Percent of population 65 years of age or older
- **HSGrad**: Percent of population 25 years of age or older with a high school degree
- **ColGrad**: Percent of population 25 years of age or older with a college degree
- **TeacherPay**: Average teachers salary in dollars
- **InfMortality**: Infant mortality per 1000 live births
- **TeenBirths**: Live births per 1000 15-19 year old females
- **ViolentCrime**: Violent crime per 100000 population
- **PropertyCrime**: Property crime per 100000 population
- **DeathPenalty**: State has death penalty?
- **Executions**: Number of executions 1977-2003
- **Poverty**: Percent of population below the poverty level
- **Unemp**: Percent unemployed (of population 16 years or older)
- **Uninsured**: Percent uninsured (3 year average)
Income  Median household income in 1998 dollars
Earnings  Average hourly earnings of production workers in manufacturing
Heart  Deaths by heart disease per 100000 population
Vehicles  Deaths by motor vehicle accidents per 100000 population
Homeowners  Home ownership rate

Details
Data on the 50 states.

Source
United States Census Bureau http://www.census.gov/

---

stemplot

Stem and leaf plot

Description
Stem and leaf plot. Will accept a factor variable as a second argument to create stem plots for each of the levels.

Usage
stemPlot(x, ...)

## Default S3 method:
stemPlot(x, grpvar = NULL, varname = NULL, grpvarname = NULL, 
...)

## S3 method for class 'formula'
stemPlot(formula, data = parent.frame(), subset, ...)

Arguments

- **x**  a numeric variable.
- **grpvar**  a factor variable. A stem plot of x will be created for each level of the factor variable.
- **varname**  name of the numeric variable. This is for printing the output only. Change if you want to print out a name different from the actual variable name.
- **grpvarname**  name of the factor variable. This is for printing the output only. Change if you want to print out a name different from the actual variable name.
- **formula**  a formula of the form x ~ grpvar where x is numeric and grpvar is a factor variable.
- **data**  a data frame with the variables in the formula.
- **subset**  an optional expression specifying which observations to keep.
- **...**  further arguments to be passed to or from methods.
stemPlot

Details

This command is just an enhanced version of R’s `stem` command. It allows the user to create the stem plot for a numeric variable grouped by the levels of a factor variable.

Author(s)

Laura Chihara

Examples

```r
stemPlot(states03$Births, states03$Region)

stemPlot(Births ~ Region, data = states03)
```
Index

* Topic ANOVA
  anovaSummarized, 2

* Topic Correlation
  corDemo, 8

* Topic Summarized data
  anovaSummarized, 2

* Topic bar plot
  groupedBar, 9

* Topic bootstrap
  boot, 3
  bootPaired, 6

* Topic confidence interval
  confIntDemo, 7

* Topic correlation
  permTestCor, 14

* Topic datasets
  Icecream, 10
  Milkshakes, 11
  states03, 17

* Topic grouped bar plot
  groupedBar, 9

* Topic missing values
  missingLevel, 12

* Topic normal quantile-quantile plot
  qqPlotDemo, 16

* Topic paired data
  permTestPaired, 15

* Topic permutation test
  permTest, 12
  permTestCor, 14
  permTestPaired, 15

* Topic randomization
  boot, 3
  bootPaired, 6
  permTest, 12
  permTestCor, 14
  permTestPaired, 15

* Topic resampling
  boot, 3
  bootCor, 5
  bootPaired, 6
  permTest, 12
  permTestCor, 14
  permTestPaired, 15

* Topic stem plot
  stemPlot, 18

  anovaSummarized, 2
  boot, 3
  bootCor, 5
  bootPaired, 6

  Carleton (Carleton-package), 2
  Carleton-package, 2
  confIntDemo, 7
  corDemo, 8
  groupedBar, 9
  Icecream, 10
  legend, 9
  Milkshakes, 11
  missingLevel, 12
  permTest, 12
  permTestCor, 14
  permTestPaired, 15
  qqPlotDemo, 16
  states03, 17
  stemPlot, 18