Package 'npExact'

October 21, 2013

Type Package

Title Nonparametric (or distribution-free) hypothesis tests

Version 0.2

Date 2013-09-26

Description This package contains several new hypothesis tests, which do not make assumptions on the underlying distributions.

Depends R (>= 2.10.0), stats

License GPL-2

R topics documented:

`npExact-package` *Nonparametric hypothesis tests*

Description

`npExact` provides distribution-free hypothesis tests.

Details

Package:	npExact
Type:	Package
Version:	0.2
Date:	2013-09-26
License:	GPL-2

This package contains several new hypothesis tests, which do not require that the user makes assumptions on the underlying distributions.

However, all tests except "npStochin" can only be applied if there are exogenously given bounds known to the user before gathering the data such that it is known by definition of the underlying process that all observations lie within these bounds.

So for instance, if the data involves percentages then the lower bound is 0 and the upper bound is 100, by definition of the data and not something (like normality) that cannot be deduced from the properties of the data.

Author(s)

Karl Schlag, Oliver Reiter, Peter Saffert, Christian Pechhacker, Simona Jokubauskaite, Tautvilas Janusauskas

References

Karl Schlag, A New Method for Constructing Exact Tests without Making any Assumptions (August, 2008) Department of Economics and Business Working Paper 1109, Universitat Pompeu Fabra

See Also

http://homepage.univie.ac.at/karl.schlag/research/statistics/exacthypothesistesting8.pdf

http://homepage.univie.ac.at/karl.schlag/research/statistics.html

---

bargaining *Amount sent in the Ultimatum Game*

Description

The Ultimatum game was played separately in four different countries. This data contains the offers of 30 students in Israel and 27 in the United States on a scale from 0 to 10. This dataset is taken from Roth et al. (1991).

Usage

```
bargaining
```

Format

A data frame containing 30 observations for Israel and 27 for the US.

References

Roth et al. (1991), American Economic Review, Vol. 81, No. 5 (Dec., 1991), pp. 1068-1095.

`french` *Indices of minority shareholder protection of countries with civil law with and without french origin.*

Description

This data contains the indices of minority shareholder protection on a scale from 0 to 1 in 51 countries with civil law, differentiating between those with (32 observations) and those without (19 observations) french origin. A higher value of the index means that country is more protected. The data set is taken from Djankov et al. (2008).

Usage

```
french
```

Format

A dataframe containing 32 observations of countries with french origin and 19 countries without french origin.

References

Djankov et al. (2008), J. Financial Economics 88 (2008) 430-465.

`mshscores` *Indices of minority shareholder protection of countries with common and with civil law.*

Description

This data contains the indices of minority shareholder protection on a scale from 0 to 1 in 51 countries with civil law and 21 countries with common loaw. A higher value of the index means that country is more protected. The data set is taken from Djankov et al. (2008).

Usage

```
mshscores
```

Format

A dataframe containing 51 observations for civil law and 21 for common law.

References

Djankov et al. (2008), J. Financial Economics 88 (2008) 430-465.

`npMeanPaired` *A test for the mean difference between two bounded random variables given matched pairs.*

Description

This test requires that the user knows bounds before gathering the data such that the properties of the data generating process imply that all observations will be within these bounds. The data input consists of pairs of observations, each pair consisting of an observation of each random variable, different pairs being independently generated.

Under alternative = "greater", it is a test of the null hypothesis $H_0 : E(x_1) <= E(x_2)$ against the alternative hypothesis $H_1 : E(x_1) > E(x_2)$.

Usage

```
npMeanPaired(x1, x2, lower = 0, upper = 1, alpha = 0.05,
             alternative = "greater", epsilon = 1 * 10^(-6),
             iterations = 5000)
```

Arguments

`x1, x2`	the (non-empty) numerical data vectors which contain the variables to be tested. The first values of the vectors are assumed to be the first matched pair of observations, the second values the second matched pair and so on.
`lower, upper`	the theoretical lower and upper bounds on the data outcomes known ex-ante before gathering the data.
`alpha`	the type I error.
`alternative`	a character string describing the alternative hypothesis, can take values "greater", "less" or "two.sided".
`iterations`	Number of iterations.
`epsilon`	the tolerance in terms of probability of the monte carlo simulations.

Details

This test uses the known bounds of the variables to transform the data into [0, 1]. Then a random transformation is used to turn the data into binary-valued variables. On this variables the exact McNemar Test with level `pseudoalpha` is performed and the result recorded. The random transformation and the test are then repeated `iterations` times. If the proportion of rejected iterations is at least `theta`, then the null hypothesis is rejected.

`theta` (and a value `mu` of the difference between the two means in the set of the alternative hypothesis) is found in an optimization procedure. `theta` and `mu` are chosen as to maximize the set of data generating processes belonging to the alternative hypothesis that yield type II error probability below 0.5. Please see the cited paper below for further information.

Value

A list with class "nphtest" containing the following components:

`method`	a character string indicating the name and type of the test that was performed.

data.name	a character string giving the name(s) of the data.
alternative	a character string describing the alternative hypothesis.
estimate	the sample means of the given data.
probrej	the probability with which the null hypothesis can be rejected.
bounds	the lower and upper bounds of the variables.
null.value	the specified hypothesized value of the difference of the variable means.
alpha	the type I error.
theta	the parameter that minimizes the type II error.
pseudoalpha	theta * alpha, this is the alpha that is used during the iterations.
rejection	Logical. Given alpha, the type I error, and theta, can a rejection be achieved?
iterations	the number of iterations that were performed.

Author(s)

Karl Schlag and Christian Pechhacker

References

Schlag, Karl H. 2008, A New Method for Constructing Exact Tests without Making any Assumptions, Department of Economics and Business Working Paper 1109, Universitat Pompeu Fabra. Available at http://www.econ.upf.edu/en/research/onepaper.php?id=1109.

See Also

http://homepage.univie.ac.at/karl.schlag/research/statistics.html

Examples

```
## test whether pain after the surgery is less than before the surgery
data(pain)
npMeanPaired(pain$before, pain$after, lower = 0, upper = 100)

## when the computer was used in the surgery
before_pc <- pain[pain$pc == 1, "before"]
after_pc <- pain[pain$pc == 1, "after"]
npMeanPaired(before_pc, after_pc, lower = 0, upper = 100)

## when only hands were used
before_hand <- pain[pain$pc == 0, "before"]
after_hand <- pain[pain$pc == 0, "after"]
npMeanPaired(before_hand, after_hand, lower = 0, upper = 100)
```

npMeanSingle *A test on the mean of a single variable.*

Description

npMeanSingle provides an exact test for the mean of a single sample based on Schlag (2008).

Usage

```
npMeanSingle(x, mu, lower = 0, upper = 1, iterations = 5000,
             alpha = 0.05, alternative = "greater",
             epsilon = 1 * 10^(-6), ignoreNA = FALSE)
```

Arguments

x	a (non-empty) numeric vector of data values.
mu	test value for $E(x) <= \mu$.
lower, upper	the theoretical lower and upper bounds on the data outcomes known ex-ante before gathering the data.
iterations	number of iterations.
alpha	the type I error.
alternative	a character string describing the alternative hypothesis, can take values "greater", "less" or "two.sided".
epsilon	the tolerance in terms of probability of the monte carlo simulations.
ignoreNA	if TRUE, NA values will be omitted. Default: FALSE

Details

This test performs a test on the mean of a variable, of which only exogenous bounds must be known, but not the distribution of the variable.

With the known bounds, the data is transformed to lie in [0, 1]. Then, 0 and 1 are randomly assigned to each value, according to its probability (which is its value after the transformation into [0, 1]). The exact binomial test then calculates the rejection probability of this new data. These steps are repeated very often. If the fraction of rejection is greater than theta, one can reject the null hypothesis.

theta (and a mu in the set of the alternative hypothesis) is found in an optimization procedure. theta and the alternative mu are chosen as to bring the type II error to 0.5. Please see the cited paper below for further information.

Value

A list with class "nphtest" containing the following components:

method	a character string indicating the name and type of the test that was performed.
data.name	a character string giving the name(s) of the data.
alternative	a character string describing the alternative hypothesis.
estimate	the estimated mean or difference in means depending on whether it was a one-sample test or a two-sample test.

probrej	the probability with which the null hypothesis can be rejected.
bounds	the lower and upper bounds of the variables.
null.value	the specified hypothesized value of the correlation between the variables.
alpha	the type I error
theta	the parameter that minimizes the type II error.
pseudoalpha	theta * alpha, this is the alpha that is used during the iterations.
rejection	Logical. Given alpha, the type I error, and theta, can a rejection be achieved?
iterations	the number of iterations that were performed.

Author(s)

Karl Schlag, Peter Saffert and Oliver Reiter

References

Schlag, Karl H. 2008, A New Method for Constructing Exact Tests without Making any Assumptions, Department of Economics and Business Working Paper 1109, Universitat Pompeu Fabra. Available at http://www.econ.upf.edu/en/research/onepaper.php?id=1109.

See Also

http://homepage.univie.ac.at/karl.schlag/research/statistics.html

Examples

```
## test whether Americans gave more than 5 dollars in a round of
## the Ultimatum game
data(bargaining)
us_offers <- bargaining$US
npMeanSingle(us_offers, mu = 5, lower = 0, upper = 10, alternative =
"greater", ignoreNA = TRUE) ## no rejection

## test if the decrease in pain before and after the surgery is smaller
## than 50
data(pain)
pain$decrease <- with(pain, before - after)
without_pc <- pain[pain$pc == 0, "decrease"]
npMeanSingle(without_pc, mu = 50, lower = 0, upper = 100,
alternative = "less")
```

---

npMeanUnpaired *A test on the means of two unpaired variables.*

Description

A test on the mean of two variables, where the observations do not have to be paired.

Usage

```
npMeanUnpaired(x1, x2, lower = 0, upper = 1, iterations = 5000, alpha = 0.05,
               alternative = "greater", epsilon = 1 * 10^(-6), ignoreNA = FALSE)
```

Arguments

x1, x2	the (non-empty) numerical data vectors which contain the variables to be tested.
lower, upper	the theoretical lower and upper bounds on the data outcomes known ex-ante before gathering the data.
iterations	the number of iterations.
alpha	the type I error.
alternative	a character string describing the alternative hypothesis, can take values "greater", "less" or "two.sided".
epsilon	the tolerance in terms of probability of the monte carlo simulations.
ignoreNA	if TRUE, NA values will be omitted. Default: FALSE

Details

This function performs a test on the means of two variables, $H_0 : E(x_1) >= E(x_2)$, with the corresponding alternative hypothesis $H_1 : E(x_1) < E(x_2)$ or $E(x_2) - E(x_1) > 0$.

First, the data is transformed to lie in [0, 1]. This is done with the ex ante known bounds of the variables. Afterwards, the iteration phase takes place: In each iteration, each observation is randomly assigned a value 0, 1 (according to its probability, the value in [0, 1]). A randomized Fischer-Tocher-Test with level pseudoalpha is then applied to the data and its result recorded. If the fraction of the rejections is at least theta, then one can reject the null hypothesis.

theta (and a mu in the set of the alternative hypothesis) is found in an optimization procedure. theta and the alternative mu are chosen as to bring the type II error to 0.5. Please see the cited paper below for further information.

Value

A list with class "nphtest" containing the following components:

method	a character string indicating the name and type of the test that was performed.
data.name	a character string giving the name(s) of the data.
alternative	a character string describing the alternative hypothesis.
estimate	the sample means of the two variables.
probrej	the probability with which the null hypothesis can be rejected.
bounds	the lower and upper bounds of the variables.
null.value	the specified hypothesized value of the correlation between the variables.
alpha	the type I error.
theta	the parameter that minimizes the type II error.
pseudoalpha	theta * alpha, this is the alpha that is used during the iterations.
rejection	Logical. Given alpha, the type I error, and theta, can a rejection be achieved?
iterations	the number of iterations that were performed.

Author(s)

Karl Schlag, Christian Pechhacker, Peter Saffert and Oliver Reiter

References

Karl Schlag (2008), A New Method for Constructing Exact Tests without Making any Assumptions. Available at http://www.econ.upf.edu/en/research/onepaper.php?id=1109.

See Also

http://homepage.univie.ac.at/karl.schlag/research/statistics.html

Examples

```
## test whether countries with french origin score lower than
## countries with no french origin
data(french)
npMeanUnpaired(french[,1], french[,2], alternative = "less", ignoreNA =
TRUE)

## test whether American tend to be more generous than Isrealis
## in a round of the Ultimatum game
data(bargaining)
npMeanUnpaired(bargaining$US, bargaining$IS, lower = 0, upper = 10, ignoreNA = TRUE)
```

---

npStochin *A test for stochastic inequality.*

Description

This test performs the exact McNemar test on a bootstrapped sample of the data. The corresponding null hypothesis for this test is $H_0 : P(x_2 > x_1) <= P(x_2 < x_1)$, or equivalently: $P(x_2 > x_1) - P(x_2 < x_1) <= d$.

Usage

```
npStochin(x1, x2, d = 0, alternative = "greater", iterations = 5000,
          alpha = 0.05, epsilon = 1 * 10^(-6), ignoreNA = FALSE)
```

Arguments

x1, x2	the (non-empty) numerical data vectors which contain the variables to be tested.
d	the difference in probabilities of the null hypothesis: $P(x_2 > x_1) - P(x_2 < x_1) <= d$. Default is 0.
alternative	a character string describing the alternative hypothesis. Default is "greater". If "less" is given, x1 and x2 are switched for each other.
iterations	Number of iterations.
alpha	the type I error.
epsilon	the tolerance in terms of probability of the monte carlo simulations.
ignoreNA	if TRUE, NA values will be omitted. Default: FALSE

Details

The function draws two random samples (with the length of the shorter vector) of the data vectors and sees them as matched pairs. These matched pairs are then compared and the occurrences of $x_1 > x_2$ and $x_1 < x_2$ are counted. The exact binomial test then tests whether the success probability of $x_1 > x_2$ is greater than 0.5 (in the default case). These steps are then repeated `iterations` times and rejection at level `alpha` times `theta` is recorded (for `theta`, see below). If the rejection probability (the fraction of rejections in all iterations) is above the threshold `theta`, the null hypothesis is rejected.

`theta` (and a `d` in the set of the alternative hypothesis) is found in an optimization procedure. `theta` and the alternative `d` are chosen so as to bring the type II error to 0.5. Please see the cited paper below for further information.

Value

A list with class "nphtest" containing the following components:

`method`	a character string indicating the name and type of the test that was performed.
`data.name`	a character string giving the name(s) of the data.
`alternative`	a character string describing the alternative hypothesis.
`estimate`	an estimate of $P(x_2 > x_1) - P(x_2 < x_1)$.
`probrej`	the probability with which the null hypothesis can be rejected.
`bounds`	the lower and upper bounds of the variables.
`null.value`	the specified hypothesized value of the correlation between the variables.
`alpha`	the type I error.
`theta`	the parameter that minimizes the type II error.
`pseudoalpha`	`theta` * `alpha`, this is the alpha that is used during the iterations.
`rejection`	Logical. Given `alpha`, the type I error, and `theta`, can a rejection be achieved?
`iterations`	the number of iterations that were performed.

Author(s)

Karl Schlag and Peter Saffert

References

Schlag, Karl H. 2008, A New Method for Constructing Exact Tests without Making any Assumptions, Department of Economics and Business Working Paper 1109, Universitat Pompeu Fabra. Available at http://www.econ.upf.edu/en/research/onepaper.php?id=1109.

See Also

http://homepage.univie.ac.at/karl.schlag/research/statistics.html

Examples

```
data(french)
x <- french[, 1]
y <- french[, 2]
npStochin(x, y, ignoreNA = TRUE)
```

`npVarianceSingle` *A test on the variance of a variable.*

Description

This is a program that implements a nonrandomized one-sided test for the variance of a single sample of independent observations. The null hypothesis that is being tested is $H_0 : Var(x) >= w$ against $H_1 : Var(x) < w$

Usage

```
npVarianceSingle(x, lower = 0, upper = 1, variance,
      alternative = "greater", alpha = 0.05, iterations = 5000,
      epsilon = 1 * 10^(-6), ignoreNA = FALSE)
```

Arguments

`x`	a (non-empty) numeric vector of data values.
`lower, upper`	the theoretical lower and upper bounds on the data outcomes known ex-ante before gathering the data.
`variance`	the value of the variance to be tested as $H_0 : Var(x) >= w$.
`alternative`	a character string describing the alternative hypothesis, can take values "greater", "less" or "two.sided"
`alpha`	the type I error.
`iterations`	the number of iterations.
`epsilon`	the tolerance in terms of probability of the monte carlo simulations.
`ignoreNA`	if `TRUE`, NA values will be omitted. Default: `FALSE`

Details

This function does the following steps in each iteration: First, it permutes the data vector. Then it calculates pairwise differences and squares them: $(x_{1:(n/2)} - x_{(n/2+1):n})^2$. These values all lie in [0, 1] and are then randomly transformed into 0, 1. A binomial test with $p = 2 * \text{variance}/(\text{upper} - \text{lower})^2$ is then performed on the data. If the fraction of recorded rejections lies above the value `theta`, the null hypothesis can be rejected.

`theta` (and a `w` in the set of the alternative hypothesis) is found in an optimization procedure. `theta` and the alternative `w` are chosen as to bring the type II error to 0.5. Please see the cited paper below for further information.

Value

A list with class "nphtest" containing the following components:

`method`	a character string indicating the name and type of the test that was performed.
`data.name`	a character string giving the name(s) of the data.
`alternative`	a character string describing the alternative hypothesis.
`estimate`	the estimated mean or difference in means depending on whether it was a one-sample test or a two-sample test.

probrej	the probability with which the null hypothesis can be rejected.
bounds	the lower and upper bounds of the variables.
null.value	the specified hypothesized value of the correlation between the variables.
alpha	the type I error.
theta	the parameter that minimizes the type II error.
pseudoalpha	theta * alpha, this is the alpha that is used during the iterations.
rejection	Logical. Given alpha, the type I error, and theta, can a rejection be achieved?
iterations	the number of iterations that were performed.

Author(s)

Karl Schlag and Oliver Reiter

References

Karl Schlag (2008). Exact tests for correlation and for the slope in simple linear regressions without making assumptions. Available at http://www.econ.upf.edu/en/research/onepaper.php?id=1097.

See Also

http://homepage.univie.ac.at/karl.schlag/research/statistics.html

Examples

```
## see if the minority share holder shores have a variance greater
## than 0.05
data(mshscores)

scores <- as.vector(as.matrix(mshscores))
npVarianceSingle(scores, lower = 0, upper = 1, var = 0.05, ignoreNA = TRUE)
```

---

pain *Pain experienced before and after a knie operation*

Description

There are two ways to determine where to start an operation on a knee, either with a computer or manually. The data describes the pain experienced by the patients before and after the surgery.

Usage

```
pain
```

Format

A dataframe containing 50 observations. Column "pc" indicates if a computer was used (coded with "1") or not (coded with "0")

Source

Karl Schlag

References

Karl Schlag

`uncertainty` *Uncertainty in a game theoretical experiment.*

Description

In an experiment, subjects played a similar game twice. Choices could be between 110 and 170. Each time, before they made their own choice, they had to indicate an interval [L, U] that they believed would contain the choice of their opponent. They paid some additional money if the choice of their opponent was in the interval they specified, and were paid more the smaller this interval was. So the width W_i of this interval in round i gives an indication of how uncertain they are in round i. The data contains the interval width in round 1 and 2.

Usage

```
uncertainty
```

Format

A dataframe containing the 25 intervals in each round of the game.

Source

Karl Schlag

References

Karl Schlag

Index