Package ‘lawstat’

January 17, 2017

Type Package
Title Tools for Biostatistics, Public Policy, and Law
Version 3.1
Date 2017-01-17
Depends R (>= 2.6.0), Hmisc, Kendall, mvtnorm, VGAM
Suggests fBasics
License GPL (>= 2)
Description Statistical tests widely utilized in biostatistics, public policy, and law. Along with the well-known tests for equality of means and variances, randomness, measures of relative variability etc, the package contains new robust tests of symmetry, omnibus and directional tests of normality, and their graphical counterparts such as Robust QQ plot; a robust trend tests for variances etc. All implemented tests and methods are illustrated by simulations and real-life examples from legal statistics, economics, and biostatistics.
NeedsCompilation no
Author Joseph L. Gastwirth [aut],
Yulia R. Gel [aut],
W. L. Wallace Hui [aut],
Vyacheslav Lyubchich [aut, cre],
Weihen Miao [aut],
Kimihiro Noguchi [aut]
Maintainer Vyacheslav Lyubchich <lyubchic@umces.edu>
Repository CRAN
Date/Publication 2017-01-17 00:28:15

R topics documented:

  bartels.test   .......................................................... 2
  bias           ............................................................ 3
  blackhire      ........................................................... 4
  brunner.munzel.test .................................................. 4
**bartels.test**

**Description**

This function performs the Bartels test for randomness which is based on the ranked version of von Neumann’s ratio (RVN). Users can choose whether to test against two-sided, negative or positive correlation. NAs from the data are omitted.

**Usage**

```r
bartels.test(y, alternative = c("two.sided", "positive.correlated", "negative.correlated"))
```

**Arguments**

- `y` a numeric vector of data values.
- `alternative` a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "negative.correlated", or "positive.correlated".
Value

A list with the following components.

- statistic: the value of the standardized Bartels statistic.
- parameter: RVN ratio.
- p.value: the p-value for the test.
- data.name: a character string giving the names of the data.
- alternative: a character string describing the alternative hypothesis.

Author(s)

Kimihiro Noguchi, Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


See Also

`runs.test`

Examples

```r
## Simulate 100 observations from an autoregressive model of
## the first order AR(1)
y = arima.sim(n = 100, list(ar = c(0.5)))

## Test y for randomness
bartels.test(y)

## Sample Output
##
## Bartels Test - Two sided
## data: y
## Standardized Bartels Statistic -4.4929, RVN Ratio =
## 1.101, p-value = 7.024e-06
```

Description

Prediction errors of 48-hour ahead MM5 forecasts of surface temperature measured at 96 different locations in the US Pacific Northwest on January 3, 2000. The prediction error, or "bias", is the difference between the forecasted and observed surface temperature. (MM5 is the fifth-generation Pennsylvania State University – National Center for Atmospheric Research Mesoscale Model.)
Usage

bias

Source

Data have been kindly provided by the research group of Professor Clifford Mass in the Department of Atmospheric Sciences at the University of Washington. Detailed information about the Pacific Northwest prediction effort and the associated data archive can be found online at www.atmos.washington.edu/mm5rt/info.html and www.atmos.washington.edu/marka/pnw.html, respectively.

blackhire

Hiring data for eight professions and two races

Description

Number of black and white candidates (hired or rejected) for eight professions.

Usage

blackhire

Format

An array with 2 rows by 2 columns by 8 levels.

References


brunner.munzel.test

The Brunner–Munzel Test for Stochastic Equality

Description

This function performs the Brunner–Munzel test for stochastic equality of two samples, which is also known as the Generalized Wilcoxon Test. NAs from the data are omitted.

Usage

brunner.munzel.test(x, y, alternative = c("two.sided", "greater", "less"), alpha = 0.05)
Arguments

- `x` the numeric vector of data values from the sample 1.
- `y` the numeric vector of data values from the sample 2.
- `alpha` significance level, default is 0.05 for 95% confidence interval.
- `alternative` a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less". User can specify just the initial letter.

Value

A list containing the following components:

- `statistic` the Brunner–Munzel test statistic.
- `parameter` the degrees of freedom.
- `conf.int` the confidence interval.
- `p.value` the p-value of the test.
- `data.name` a character string giving the name of the data.
- `estimate` an estimate of the effect size, i.e., \( P(X < Y) + 0.5 \times P(X = Y) \)

Author(s)

Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao This function was updated with the help of Dr. Ian Fellows

References


See Also

`wilcox.test`, `pwilcox`

Examples

```r
## Pain score on the third day after surgery for 14 patients under
## the treatment Y and 11 patients under the treatment N
## (see Brunner and Munzel, 2000).

Y <- c(1, 2, 1, 1, 1, 1, 1, 2, 4, 1, 1)
N <- c(3, 3, 4, 3, 1, 2, 3, 1, 1, 5, 4)

brunner.munzel.test(Y, N)
```
## Description

This function measures relative inequality (or relative variation) of the data. Coefficient of Dispersion (CD) is the ratio of the Average Absolute Deviation from the Median (MAAD) to the Median of the data. NAs from the data are omitted.

## Usage

```
cd(x)
```

## Arguments

- `x` a numeric vector of data values.

## Value

A list with the following numeric components.

- `statistic` the coefficient of dispersion.
- `data.name` a character string giving the name of the data.

## Author(s)

Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

## References


## See Also

- `gini.index`
- `j.maad`
Examples

```r
## The Baker v. Carr Case: one-person-one-vote decision.
## Measure of Relative Inequality of Population data in 33 districts
## of the Tennessee Legislature in 1900 and 1972. See
## popdata (see Gastwirth, 1988).

data(popdata)
cd(popdata[, "pop1900"])

## Measures of Relative Variability - Coefficient of Dispersion
##
## data: popdata[, "pop1900"]
## Coefficient of Dispersion = 0.1673

cd(popdata[, "pop1972"])

## Measures of Relative Variability - Coefficient of Dispersion
##
## data: popdata[, "pop1972"]
## Coefficient of Dispersion = 0.0081
```

---

**cmh.test**

*The Cochran-Mantel-Haenszel Chi-square Test*

**Description**

This function performs the Cochran-Mantel-Haenszel (CMH) procedure. The CMH procedure tests homogeneity of population proportions after taking into account other factors. This procedure is widely used in various law cases, in particular, on equal employment and discrimination, as well in biological and pharmaceutical studies.

**Usage**

```r
cmh.test(x)
```

**Arguments**

- `x`  
a numeric 2 x 2 x k array of data values.

**Details**

The test is based on the CMH procedure discussed by Gastwirth, 1984. The data should be input in a array of 2 rows x 2 columns x k levels. The output includes the Mantel-Haenszel Estimate, the pooled Odd Ratio, and the Odd Ratio between the rows and columns at each level. The Chi-square Test of Significance tests if there is an interaction or association between rows and columns.

The null hypothesis is that the pooled Odd Ratio is equal to 1, i.e., there is no interaction between rows and columns. For more details see Gastwirth (1984).
Notice that cmh.test can be viewed as a subset of mantelhaen.test, in the sense that cmh.test is for a 2 by 2 by k table without continuity correction whereas mantelhaen.test allows for a larger table, and for a 2 by 2 by k table, it has an option of performing continuity correction or not. However, in view of Gastwirth (1984), continuity correction is not recommended as it tends to overestimate the p-value.

Value

A list with class htest containing the following components:

- `OR`: Pooled Odd Ratio of the data.
- `ORK`: vector of Odd Ratio of each level
- `cmh`: the test statistic.
- `df`: degrees of freedom.
- `p.value`: the p-value of the test.
- `method`: type of test was performed.
- `data.name`: a character string giving the name of the data.

Author(s)

Min Qin, Wallace W. Hui, Yulia R. Gel, Joseph L. Gastwirth

References


See Also

mantelhaen.test

Examples

```R
## Sample Salary Data

data(blackhire)
cmh.test(blackhire)

## Sample Output
##
## Mantel-Haenszel Chi-square Test
##
## data:  blackhire
## Mantel-Haenszel Estimate = 0.477, Chi-squared = 145.840, df = 1.000, p-value = 0.000,
## Pooled Odd Ratio = 0.639, Odd Ratio of level 1 = 1.329, Odd Ratio of level 2 = 0.378, Odd
```
## data1963

**Population data of ratio of number of senators and representatives to population size in 1963**

### Description

The dataset of ratio of number of senators and representatives to population size in 13 districts in the United States in 1963 (Gastwirth, 1972).

### Usage

data1963

### Format

A data frame with 13 observations on the following 3 variables.

- **pop1963** population data in 1963
- **sen1963** Number of senators in each district in 1963
- **rep1963** Number of representatives in each district in 1963

### Source


### References

Description

This function measures relative inequality (or relative variation) of the data using the Gini Index. NAs from the data are omitted.

Usage

gini.index(x)

Arguments

x the input data.

Value

A list with the following numeric components.

statistic The Gini Index of the data.
parameter the mean difference of a set of numbers.
data.name a character string giving the name of the data.

Author(s)

Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


See Also

cd, j.maad, lorenz.curve
Examples

```r
## The Baker v. Carr Case: one-person-one-vote decision.
## Measure of Relative Inequality of Population data in 33 districts
## of the Tennessee Legislature in 1900 and 1972. See
## popdata (see Gastwirth (1988)).

data(popdata)
gini.index(popdata[, "pop1900"])

## Measures of Relative Variability - Gini Index
##
## data:  popdata[, "pop1900"]
## Gini Index = 0.1147, delta = 3389.765

gini.index(popdata[, "pop1972"])

## Measures of Relative Variability - Gini Index
##
## data:  popdata[, "pop1972"]
## Gini Index = 0.0055, delta = 1297.973
```

Description

This function computes the average absolute deviation from the sample median, which is a consistent robust estimate of the population standard deviation for normality distribution data. NAs from the data are omitted.

Usage

```r
j.maad(x)
```

Arguments

- `x`: a numeric vector of data values.

Value

A list with the following numeric components.

- `J`: Robust Standard Deviation J

Author(s)

Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao
References


See Also

`cd`, `gini.index`, `rqq`, `rjb.test`, `sj.test`

Examples

```r
## Simulate 100 observations: using rnorm()
## for normally distributed data, X=N(0,1)
x = rnorm(100)
j.maad(x)

## Sample Output
##
## MAAD estimated J = 0.9194124302405 for data x
```

`laplace.test`  
*Goodness-of-fit tests for the Laplace distribution*

Description

The function returns five goodness-of-fit test statistics for the Laplace distribution. The four statistics are: $A_2$ (Anderson-Darling), $W_2$ (Cramer-von Mises), $U_2$ (Watson), $D$ (Kolmogorov-Smirnov), and $V$ (Kuiper). By default, NAs are omitted. This function requires the VGAM package.

Usage

`laplace.test(y)`

Arguments

y  
a numeric vector of data values.

Value

A list with the following numeric components.

- $A_2$  
  the Anderson-Darling statistic.
- $W_2$  
  the Cramer-von Mises statistic.
- $U_2$  
  the Watson statistic.
- $D$  
  the Kolmogorov-Smirnov statistic.
- $V$  
  the Kuiper statistic.
levene.test

Author(s)
Kimihiro Noguchi, Yulia R. Gel

References


See Also
plaplace (in VGAM package)

Examples

```r
## Differences in flood levels example taken from Puig and Stephens (2000)
library(VGAM)
y<-c(1.96,1.97,3.60,3.80,4.79,5.66,5.76,5.78,6.27,6.30,6.76,7.65,7.84,7.99,8.51,9.18,10.13,10.24,10.25,10.43,11.45,11.48,11.75,11.81,12.33,12.78,13.06,13.29,13.98,14.18,14.40,16.22,17.06)
levene.test(y)
```

```r
## [1] 0.9177726
## The critical value at the 0.05 significance level is approximately 0.906.
## Thus, the null hypothesis would be rejected at the 0.05 level.
## For the tables of critical values, see Stephens (1986) or Puig and Stephens (2000).
```

levene.test  Levene’s Test of Equality of Variances

Description
The function performs the following tests for equality of the k population variances: classical Levene’s test, the robust Brown-Forsythe Levene-type test using the group medians and the robust Levene-type test using the group trimmed mean. More robust versions of the test using the correction factor or structural zero removal method are also available. Two options for calculating critical values, namely, approximated and bootstrapped, are available. Instead of the ANOVA statistic suggested by Levene, the Kruskal-Wallis ANOVA may also be applied using this function. By default, NAs from the data are omitted.

Usage
levene.test(y, group, location=c("median", "mean", "trim.mean"), trim.alpha=0.25, bootstrap = FALSE, num.bootstrap=1000, kruskal.test=FALSE, correction.method=c("none","correction.factor","zero.removal","zero.correction"))
Arguments

- `y` a numeric vector of data values.
- `group` factor of the data.
- `location` the default option is "median" corresponding to the robust Brown-Forsythe Levene-type procedure; "mean" corresponds to the classical Levene’s procedure, and "trim.mean" corresponds to the robust Levene-type procedure using the group trimmed means.
- `trim.alpha` the fraction (0 to 0.5) of observations to be trimmed from each end of `x` before the mean is computed.
- `bootstrap` the default option is FALSE, i.e., no bootstrap; if the option is set to TRUE, the function performs the bootstrap method described in Lim and Loh (1996) for Levene’s test.
- `num.bootstrap` number of bootstrap samples to be drawn when the bootstrap option is set to TRUE; the default value is 1000.
- `kruskal.test` use of Kruskal-Wallis statistic. The default option is FALSE, i.e., the usual ANOVA statistic is used in place of Kruskal-Wallis statistic.
- `correction.method` procedures to make the levene's test more robust; the default option is "none"; "correction.factor" applies the correction factor described by O'Brien (1978) and Keyes and Levy (1997); "zero.removal" performs the structural zero removal method by Hines and Hines (2000); "zero.correction" performs a combination of O’Brien’s correction factor and the Hines-Hines structural zero removal method (Noguchi and Gel, 2009); note that the options "zero.removal" and "zero.correction" are only applicable when the location is set to "median"; otherwise, "none" is applied.

Details

Levene (1960) proposed a test for homogeneity of variances in k groups which is based on the ANOVA statistic applied to absolute deviations of observations from the corresponding group mean. The robust Brown-Forsythe version of the Levene-type test substitutes the group mean by the group median in the classical Levene statistic. The third option is to consider ANOVA applied to the absolute deviations of observations from the group trimmed mean instead of the group means.

Value

A list with the following numeric components.

- `statistic` the value of the test statistic.
- `p.value` the p-value of the test.
- `method` type of test performed.
- `data.name` a character string giving the name of the data.
- `non.bootstrap.p.value` the p-value of the test without bootstrap method; i.e. the p-value using the approximated critical value.
levene.test

Note
modified from a response posted by Brian Ripley to the R-help e-mail list.

Author(s)
Kimihiro Noguchi, W. Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


See Also
neuhauser.hothorn.test, lnested.test, ltrend.test, mma.test, robust.mmm.test
Examples

```r
data(pot)
levene.test(pot[, "obs"], pot[, "type"], location="median", correction.method="zero.correction")

## modified robust Brown-Forsythe Levene-type test based on the absolute deviations
## from the median with modified structural zero removal method and correction factor
## data:  pot[, "obs"]
## Test Statistic = 6.5673, p-value = 0.001591

## Bootstrap version of the test. The calculation may take up a few minutes
## depending on the number of bootstrap sampling.
levene.test(pot[, "obs"], pot[, "type"], location="median", correction.method="zero.correction", bootstrap=TRUE,num.bootstrap=500)

## bootstrap modified robust Brown-Forsythe Levene-type test based on the absolute
## deviations from the median with structural zero removal method and correction factor
## data:  pot[, "obs"]
## Test Statistic = 6.9577, p-value = 0.001
```

Description

The function performs a test for a monotonic trend in variances. The test statistic is based on a combination of the finite intersection approach and the classical Levene procedure (using the group means), the modified Brown-Forsythe Levene-type procedure (using the group medians) or the modified Levene-type procedure (using the group trimmed means). More robust versions of the test using the correction factor or structural zero removal method are also available. Two options for calculating critical values, namely, approximated and bootstrapped, are available. By default, NAs from the data are omitted.

Usage

```r
lnested.test(y, group, location = c("median", "mean", "trim.mean"),
tail = c("right", "left", "both"), trim.alpha = 0.25,
bootstrap = FALSE, num.bootstrap = 1000,
correction.method = c("none", "correction.factor", "zero.removal", "zero.correction"),
correlation.method = c("pearson", "kendall", "spearman"))
```

Arguments

- `y` a numeric vector of data values.
group
factor of the data.

location
the default option is "median" corresponding to the robust Brown-Forsythe Levene-type procedure; "mean" corresponds to the classical Levene's procedure, and "trim.mean" corresponds to the robust Levene-type procedure using the group trimmed means.

tail
the default option is "right", corresponding to an increasing trend in variances as the one-sided alternatives; "left" corresponds to a decreasing trend in variances, and "both" corresponds to any (increasing or decreasing) monotonic trend in variances as the two-sided alternatives.

trim.alpha
the fraction (0 to 0.5) of observations to be trimmed from each end of 'x' before the mean is computed.

bootstrap
the default option is FALSE, i.e., no bootstrap; if the option is set to TRUE, the function performs the bootstrap method described in Lim and Loh (1996) for Levene's test.

num.bootstrap
number of bootstrap samples to be drawn when bootstrap is set to TRUE; the default value is 1000.

correction.method
procedures to make the Ltrend test more robust; the default option is "none"; "correction.factor" applies the correction factor described by O'Brien (1978) and Keyes and Levy (1997); "zero.removal" performs the structural zero removal method by Hines and Hines (2000); "zero.correction" performs a combination of O'Brien's correction factor and the Hines-Hines structural zero removal method (Noguchi and Gel, 2009); note that the options "zero.removal" and "zero.correction" are only applicable when the location is set to "median"; otherwise, "none" is applied.

correlation.method
measures of correlation; the default option is "pearson", the usual correlation coefficient which is equivalent to the t-test; nonparametric measures of correlation such as "kendall" (Kendall's tau) or "spearman" (Spearman's rho) may also be chosen, in which case, two libraries, Hmisc and Kendall, are required.

Value
A list with the following vector components.

T
the statistic and p-value of the test based on the Tippett p-value combination.

F
the statistic and p-value of the test based on the Fisher p-value combination.

N
the statistic and p-value of the test based on the Liptak p-value combination.

L
the statistic and p-value of the test based on the Mudholkar-George p-value combination.

Each of the vector components contains the following numeric components.

statistic
the value of the test statistic expressed in terms of correlation (Pearson, Kendall, or Spearman).

p.value
the p-value of the test.

method
type of test performed.
data.name a character string giving the name of the data.
non.bootstrap.statistic
the statistic of the test without bootstrap method.
non.bootstrap.p.value
the p-value of the test without bootstrap method.

Author(s)
Kimihiro Noguchi, W. Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


lorenz.curve

See Also

`neuhauser.hothorn.test, levene.test, ltrend.test, mma.test, robust.mmm.test`

Examples

```r
library(Hmisc)
library(Kendall)
data(pot)
l1nested.test(pot[, "obs"], pot[, "type"], location="median", tail="left",
correction.method="zero.correction")

## Intested test based on the modified Brown-Forsythe Levene-type procedure using the
## group medians with modified structural zero removal method and correction factor
## (left-tailed with Pearson correlation coefficient)
##
## data:  pot[, "obs"]
## Test Statistic (N) = 4.905, p-value = 0.0002618

l1nested.test(pot[, "obs"], pot[, "type"], location="median", tail="left",
correction.method="zero.correction", bootstrap=TRUE, num.bootstrap=500)

## bootstrap l1nested test based on the modified Brown-Forsythe Levene-type procedure
## using the group medians with modified structural zero removal method and correction
## factor (left-tailed with Pearson correlation coefficient)
##
## data:  pot[, "obs"]
## Test Statistic (N) = 4.9936, p-value = 0.000207
```

Description

This function plots the Lorenz curve that is a graphical representation of the cumulative distribution function. A user can choose for the Lorenz curve with single (default) or multiple weighting of data, for example, taking into account for single or multiple legislature representatives.

The input data should be a data frame with 2 columns. The first column will be treated as data vector, and the second column to be treated as a weight vector. Alternatively, data and weight can be entered as separate one-column vectors.

Usage

```r
lorenz.curve(data, weight=NULL, mul=FALSE, plot.it=TRUE,
             main=NULL, xlab=NULL, ylab=NULL, xlim=c(0,1), ylim=c(0,1), ...)```
Arguments

data
input data. If the argument is an array, a matrix, a data.frame, or a list with two or more columns, then the first column will be a treated as a data vector, and the second column to be treated as a weight vector. A separate weight vector is then ignored and not required. If the argument is a single column vector, then a user must enter a separate single-column weight vector. NA or character is not allowed.

weight
single column vector contains factors of single or multiple weights. Ignored if already included in the data argument. NA or character is not allowed.

mul
logical value indicates whether the Lorenz Curve with multiple weight is to be plotted. Default is single.

plot.it
logical value indicates whether the Lorenz Curve should be plotted on screen. Default is to plot.

main
Title of Lorenz Curve. Only required if user wants to override the default value.

xlab
label of x-axis. Only required if user wants to override the default value.

ylab
label of y-axis. Only required if user wants to override the default value.

xlim
plotting range of x-axis. Only required if user wants to override the default value.

ylim
plotting range of y-axis. Only required if user wants to override the default value.

... other graphical parameters to be passed to the plot function.

Value

x Culmulative fraction of the data argument.

y Culmulative fraction of the weight argument.

gini The Gini index of the input data.

relative.mean.dev Relative Mean Deviation of the input data.

L(1/2) Median of the culmulative fraction sum of the data.

Author(s)

Man Jin, Wallace W. Hui, Yulia R. Gel, Joseph L. Gastwirth

References


See Also

gini.index
Examples

```r
## Population Data of ratio of number of senators (second column) and
## representatives (third column) to population size (first column) in 1963
## First column is treated as data argument.

data(data1963)

## Single weight Lorenz Curve using number of senators as weight argument.
lorenz.curve(data1963)

## Multiple weight Lorenz Curve using number of senators as weight argument.
lorenz.curve(data1963, mul=TRUE)

## Multiple weight Lorenz Curve using number of representatives
## as weight argument.
lorenz.curve(data1963[,"pop1963"], data1963[,"rep1963"], mul=TRUE)
```

Description

The function performs a test for a linear trend in variances. The test statistic is based on the classical
Levene procedure (using the group means), the modified Brown-Forsythe Levene-type procedure
(using the group medians) or the modified Levene-type procedure (using the group trimmed means).
More robust versions of the test using the correction factor or structural zero removal method are
also available. Two options for calculating critical values, namely, approximated and bootstrapped,
are available. By default, NAs from the data are omitted.

Usage

```
ltrtest.test(y, group, score=NULL, location = c("median", "mean", "trim.mean"),
             tail = c("right", "left", "both"), trim.alpha = 0.25,
             bootstrap = FALSE, num.bootstrap = 1000,
             correction.method = c("none", "correction.factor", "zero.removal", "zero.correction"),
             correlation.method = c("pearson", "kendall", "spearman"))
```

Arguments

- `y`  a numeric vector of data values.
- `group`  factor of the data.
- `score` weights to be used in testing increasing/decreasing trend in group variances.
  "score" coincides by default with "group"; it can be chosen as a linear, quadratic
  or any other monotone function.
location
the default option is "median" corresponding to the robust Brown-Forsythe Levene-type procedure; "mean" corresponds to the classical Levene's procedure, and "trim.mean" corresponds to the robust Levene-type procedure using the group trimmed means.

tail
the default option is "right", corresponding to an increasing trend in variances as the one-sided alternatives; "left" corresponds to a decreasing trend in variances, and "both" corresponds to any (increasing or decreasing) monotonic trend in variances as the two-sided alternatives.

trim.alpha
the fraction (0 to 0.5) of observations to be trimmed from each end of 'x' before the mean is computed.

bootstrap
the default option is FALSE, i.e., no bootstrap; if the option is set to TRUE, the function performs the bootstrap method described in Lim and Loh (1996) for Levene's test.

num.bootstrap
number of bootstrap samples to be drawn when the bootstrap option is set to TRUE; the default value is 1000.

correction.method
procedures to make the ltrend.test more robust; the default option is "none"; "correction.factor" applies the correction factor described by O'Brien (1978) and Keyes and Levy (1997); "zero.remove" performs the structural zero removal method by Hines and Hines (2000); "zero.correction" performs a combination of O'Brien's correction factor and the Hines-Hines structural zero removal method (Noguchi and Gel, 2009); note that the options "zero.remove" and "zero.correction" are only applicable when the location is set to "median"; otherwise, "none" is applied.

correlation.method
measures of correlation; the default option is "pearson", the usual correlation coefficient which is equivalent to the t-test; nonparametric measures of correlation such as "kendall" (Kendall's tau) or "spearman" (Spearman's rho) may also be chosen, in which case, two libraries, Hmisc and Kendall, are required.

Value
A list with the following numeric components.

statistic
the value of the test statistic expressed in terms of correlation (Pearson, Kendall, or Spearman).

p.value
the p-value of the test.

method
type of test performed.

data.name
a character string giving the name of the data.

t.statistic
the value of the test statistic from Student's t-test.

non.bootstrap.p.value
the p-value of the test without bootstrap method.

log.p.value
the log of the p-value

log.q.value
the log of the (one minus the p-value).
Author(s)
Kimihiro Noguchi, W. Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


See Also
neuhauser.hothorn.test, levene.test, lnested.test, mma.test, robust.mmm.test

Examples
library(Hmisc)
library(Kendall)
data(pot)
ltrend.test(pot[, "obs"], pot[, "type"], location="median", tail="left",
correction.method="zero.correction")

## ltrend test based on the modified Brown-Forsythe Levene-type procedure using the
## group medians with modified structural zero removal method and correction factor
## (left-tailed with Pearson correlation coefficient)
##
## data:  pot[, "obs"]
## Test Statistic (Correlation) = -0.1929, p-value = 0.0001735

## Bootstrap version of the test. The calculation may take up a few minutes
## depending on the number of bootstrap sampling.

ltrend.test(pot[, "obs"], pot[, "type"], location="median", tail="left",
correction.method="zero.correction", bootstrap=TRUE, num.bootstrap=500)

## bootstrap ltrend test based on the modified Brown-Forsythe Levene-type procedure
## using the group medians with modified structural zero removal method and correction factor
## (left-tailed with Pearson correlation coefficient)
##
## data:  pot[, "obs"]
## Test Statistic (Correlation) = -0.1929, p-value = 0.0002

---

**michigan**

*Dioxin Levels for Counties in the Upper Peninsula of Michigan*

**Description**

Data contains 16 observations of the Dioxin Levels for counties in the Upper Peninsula of Michigan

**Usage**

michigan

**Format**

A univariate data set with 16 observations

**Source**

The Environmental Protection Agency (EPA) of the State of Michigan
Mudholkar-McDermott-Aumont test for ordered variances for normal samples

Description
The function performs a test for a monotonic trend in variances for normal samples. The test statistic is based on a combination of the finite intersection approach and the classical F (variance ratio) test. By default, NAs are omitted.

Usage
mma.test(y, group, tail=c("right","left","both"))

Arguments
- y: a numeric vector of data values.
- group: factor of the data.
- tail: the default option is "right", corresponding to an increasing trend in variances as the one-sided alternatives; "left" corresponds to a decreasing trend in variances, and "both" corresponds to any (increasing or decreasing) monotonic trend in variances as the two-sided alternatives.

Value
A list with the following vector components.
- T: the statistic and p-value of the test based on the Tippett p-value combination.
- F: the statistic and p-value of the test based on the Fisher p-value combination.
- N: the statistic and p-value of the test based on the Liptak p-value combination.
- L: the statistic and p-value of the test based on the Mudholkar-George p-value combination.

Each of the vector components contains the following numeric components.
- statistic: the value of the test statistic.
- p.value: the p-value of the test.
- method: type of test performed.
- data.name: a character string giving the name of the data.

Author(s)
Kimihiro Noguchi, Yulia R. Gel
References


See Also

neuhauser.hothorn.test, levene.test, lnested.test, ltrend.test, robust.mmm.test

Examples

data(pot)
mma.test(pot[,"obs"], pot[,"type"], tail="left")

## Mudholkar et al. (1993) test (left-tailed)
##
## data:  pot[, "obs"]
## Test Statistic (N) = 9.9429, p-value = 1.828e-12

---

neuhauser.hothorn.test

Neuhauser-Hothorn double contrast test for a monotonic trend in variances

Description

The function performs a test for a monotonic trend in variances. The test statistic suggested by Neuhauser and Hothorn (2000) is based on the classical Levene procedure (using the group means), the modified Brown-Forsythe Levene-type procedure (using the group medians) or the modified Levene-type procedure (using the group trimmed means). More robust versions of the test using the correction factor or structural zero removal method are also available. Two options for calculating critical values, namely, approximated and bootstrapped, are available. By default, NAs from the data are omitted. This function requires the mvtnorm package.

Usage

neuhauser.hothorn.test(y, group, location = c("median", "mean", "trim.mean"),
tail = c("right","left","both"), trim.alpha = 0.25,
bootstrap = FALSE, num.bootstrap = 1000,
correction.method = c("none","correction.factor","zero.removeal","zero.correction"))

Arguments

y       a numeric vector of data values.
group   factor of the data.
location

the default option is "median" corresponding to the robust Brown-Forsythe Levene-
type procedure; "mean" corresponds to the classical Levene's procedure, and
"trim.mean" corresponds to the robust Levene-type procedure using the group
trimmed means.

tail

the default option is "right", corresponding to an increasing trend in variances as
the one-sided alternatives; "left" corresponds to a decreasing trend in variances,
and "both" corresponds to any (increasing or decreasing) monotonic trend in
variances as the two-sided alternatives.

trim.alpha

the fraction (0 to 0.5) of observations to be trimmed from each end of 'x' before
the mean is computed.

bootstrap

the default option is FALSE, i.e., no bootstrap; if the option is set to TRUE, the
function performs the bootstrap method described in Lim and Loh (1996) for
Levene's test.

num.bootstrap

number of bootstrap samples to be drawn when the bootstrap option is set to
TRUE; the default value is 1000.

correction.method

procedures to make the Itrend test more robust; the default option is "none";
"correction.factor" applies the correction factor described by O'Brien (1978)
and Keyes and Levy (1997); "zero.removal" performs the structural zero re-
moval method by Hines and Hines (2000); "zero.correction" performs a com-
bination of O'Brien's correction factor and the Hines-Hines structural zero re-
moval method (Noguchi and Gel, 2009); note that the options "zero.removal"
and "zero.correction" are only applicable when the location is set to "median";
otherwise, "none" is applied.

Value

A list with the following numeric components.

statistic

the value of the test statistic.

p.value

the p-value of the test.

method

type of test performed.

data.name

a character string giving the name of the data.

non.bootstrap.p.value

the p-value of the test without bootstrap method.

Author(s)

Kimihiro Noguchi, Yulia R. Gel

References

Technometrics 31, 69-82.


**See Also**

levene.test, lnested.test, ltrend.test, mma.test, robust.mmm.test

**Examples**

```r
library(mvtnorm)
data(pot)
neuhauser.hothorn.test(pot[,"obs"], pot[,"type"], location="median", tail="left",
correction.method="zero.correction")
```

```r
## double contrast test based on the absolute deviations from the median with
## group medians with modified structural zero removal method and correction factor
## (left-tailed)
##
data:  pot[, "obs"]
## Test Statistic = -3.6051, p-value = 0.0003021
```

```r
## Bootstrap version of the test. The calculation may take up a few minutes
## depending on the number of bootstrap sampling.
neuhauser.hothorn.test(pot[,"obs"], pot[,"type"], location="median", tail="left",
correction.method="zero.correction", bootstrap=TRUE, num.bootstrap=500)
```

```r
## bootstrap double contrast test based on the absolute deviations from the median with
## modified structural zero removal method and correction factor
```
nig.parameter

## (left-tailed)
##
data:  pot[, "obs"]
## Test Statistic = -3.6051, p-value = 0.0001

### Description

The function produces four parameters, alpha (tail heaviness), beta (asymmetry), delta (scale), and mu (location) from the four variables, mean, variance, kurtosis, and skewness.

### Usage

```r
nig.parameter(mean=mean, variance=variance, kurtosis=kurtosis, skewness=skewness)
```

### Arguments

- **mean**: mean of the NIG distribution.
- **variance**: variance of the NIG distribution.
- **kurtosis**: excess kurtosis of the NIG distribution.
- **skewness**: skewness of the NIG distribution.

### Details

The parameters are generated on three conditions: 1. $3\cdot\text{kurtosis} > 5\cdot\text{skewness}^2$, 2. $\text{skewness} > 0$, and 3. $\text{variance} > 0$.

### Value

A list with the following numeric components.

- **alpha**: tail-heaviness parameter of the NIG distribution.
- **beta**: asymmetry parameter of the NIG distribution.
- **delta**: scale parameter of the NIG distribution.
- **mu**: location parameter of the NIG distribution.

### Author(s)

Kimihiro Noguchi, Yulia R. Gel
References


See Also

rnig (in fBasics package)

Examples

```
library(fBasics)
test<-rnig.parameter(0,2,5,1)
random<-rnig(1000000, alpha=test$alpha, beta=test$beta, mu=test$mu, delta=test$delta)
mean(random)
## [1] 0.0003896483
var(random)
## [1] 2.007351
kurtosis(random)
## [1] 5.085051
## attr(,"method")
## [1] "excess"
skewness(random)
## [1] 1.011352
## attr(,"method")
## [1] "moment"
```

popdata

*Population data in 33 districts of the Tennessee Legislature in 1900, 1960 and 1972*

Description


Usage

popdata
**Format**

A data frame with 33 observations on the following 3 variables.

- pop1900  population data in 1900
- pop1960  population data in 1960
- pop1972  population data in 1972

**Source**


**References**


---

**pot**

*Apertures of chupa-pots from three Philippine communities*

---

**Description**

The apertures of the chupa pots from the three Philippine locations: Dalupa (ApDl), Dangtalan (ApDg) and Paradijon (ApP).

**Usage**

pot

**Format**

A multivariate data set with 343 observations on 2 variables: apertures and locations.

**Details**

Archaeologists are concerned with the effect that increasing economic activity had on older civilizations. Economic growth and its related economic specialization led to the “standardization hypothesis”, i.e. increased production of an item would lead to its becoming more uniform. Kvamme, Stark and Longacre (1996) focused on earthenware, chupa-pots from three Philippine communities that differ in the way they organize ceramic production. In Dangtalan, pottery is primarily made for household use; in Dalupa there is a non-market barter economy where potters exchange their works. In the village of Paradijon, near the Provincial capital, full-time pottery specialists sell their output to shopkeepers for sale to the general public.

**Source**

The data are kindly provided by Professor Kvamme.
rjb.test

Test of Normality - Robust Jarque Bera Test

Description

This function performs the robust and classical Jarque-Bera tests of normality.

Usage

rjb.test(x, option = c("RJB", "JB"),
         crit.values = c("chisq.approximation", "empirical"), N = 0)

Arguments

x    a numeric vector of data values.
option The choice of the test must be "RJB" (default) or "JB".
crit.values a character string specifying how the critical values should be obtained, i.e. approximated by the chisq-distribution (default) or empirically.
N number of Monte Carlo simulations for the empirical critical values

Details

The test is based on a joint statistic using skewness and kurtosis coefficients. The Robust Jarque-Bera (RJB) is the robust version of the Jarque-Bera (JB) test of normality. In particular, RJB utilizes the robust standard deviation (namely the Average Absolute Deviation from the Median (MAAD)) to estimate sample kurtosis and skewness (default option). For more details see Gel and Gastwirth (2006). Users can also choose to perform the classical Jarque-Bera test (see Jarque, C. and Bera, A (1980)).

Value

A list with class htest containing the following components:

statistic the value of the test statistic.
parameter the degrees of freedom.
p.value the p-value of the test.
method type of test was performed.
data.name a character string giving the name of the data.

Note

Modified from 'jarque.bera.test' (in 'tseries' package).

References

Author(s)
W. Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


See Also
sj.test, rqq, jarque.bera.test (in tseries package).

Examples
```r
## Normally distributed data
x = rnorm(100)
rjb.test(x)

## Sample Output
##
## Robust Jarque Bera Test
##
## data:  x
## X-squared = 0.962, df = 2, p-value = 0.6182

## Using zuni data
data(zuni)
rjb.test(zuni[, "Revenue"])

## Robust Jarque Bera Test
##
## data:  zuni[, "Revenue"]
## X-squared = 54595.63, df = 2, p-value < 2.2e-16
```

rlm.test
Robust L1 Moment-Based (RLM) Goodness-of-Fit Test for the Laplace Distribution
Description

This function performs the robust test for the Laplace distribution. Two options for calculating critical values, namely, approximated with chisq distribution and empirical, are available.

Usage

```r
rlm.test(x, crit.values = c("chisq.approximation", "empirical"), N = 0)
```

Arguments

- `x` a numeric vector of data values.
- `crit.values` a character string specifying how the critical values should be obtained, i.e., approximated by the chisq-distribution (default) or empirical.
- `N` number of Monte Carlo simulations for the empirical critical values

Details

The test is based on a joint statistic using skewness and kurtosis coefficients. In particular, RLM uses the Average Absolute Deviation from the Median (MAAD), a robust estimate of standard deviation.

Value

A list with class htest containing the following components:

- `statistic` the value of the test statistic.
- `parameter` the degrees of freedom.
- `p.value` the p-value of the test.
- `method` type of test was performed.
- `data.name` a character string giving the name of the data.

Author(s)

Kimihiro Noguchi, W. Wallace Hui, Yulia R. Gel

References


See Also

- `sj.test`, `rjb.test`, `rqq`, `jarque.bera.test` (in `tseries` package).
**Examples**

```r
## Laplace distributed data
x = rexp(100)-rexp(100)
rlm.test(x)
```

## Sample Output

```r
## Robust L1 moment-based goodness-of-fit test using a Chi-squared approximated critical values
## data:  x
## Chi-squared statistic = 0.3945, df = 2, p-value = 0.821
```

---

**robust.mmm.test**  
*Robust Mudholkar-McDermott-Mudholkar test for ordered variances*

**Description**

The function performs a test for a monotonic trend in variances. The test statistic is based on a combination of the finite intersection approach and the two-sample t-test using Miller’s transformation. By default, NAs are omitted.

**Usage**

```r
robust.mmm.test(y, group, tail=c("right","left","both"))
```

**Arguments**

- `y`  
  a numeric vector of data values.

- `group`  
  factor of the data.

- `tail`  
  the default option is "right", corresponding to an increasing trend in variances as the one-sided alternatives; "left" corresponds to a decreasing trend in variances, and "both" corresponds to any (increasing or decreasing) monotonic trend in variances as the two-sided alternatives.

**Value**

A list with the following vector components.

- `T`  
  the statistic and p-value of the test based on the Tippett p-value combination.

- `F`  
  the statistic and p-value of the test based on the Fisher p-value combination.

- `N`  
  the statistic and p-value of the test based on the Liptak p-value combination.

- `L`  
  the statistic and p-value of the test based on the Mudholkar-George p-value combination.

Each of the vector components contains the following numeric components.
statistic the value of the test statistic.
p.value the p-value of the test.
method type of test performed.
data.name a character string giving the name of the data.

Author(s)
Kimihiro Noguchi, Yulia R. Gel

References

See Also
neuhauser.hothorn.test, levene.test, lnested.test, ltrend.test, mma.test

Examples

data(pot)
robust.mmm.test(pot[,"obs"], pot[,"type"], tail="left")

## Mudholkar et al. (1995) test (left-tailed)
## data:  pot[, "obs"]
## Test Statistic (N) = 7.4079, p-value = 8.109e-08

---

rqq Test of Normality using RQQ plots

Description
This function produces the robust quantile-quantile (RQQ) and classical quantile-quantile (QQ) plots for graphical assessment of normality and optionally adds a line, or a QQ line, to the produced plot. The QQ line may be chosen to be a 45 degree line or to pass through the first and third quartiles of the data. NAs from the data are omitted. Graphical parameters may be given as arguments to 'rqq'.

Usage
rqq(y, plot.it = TRUE, square.it=TRUE, scale = c("MAD", "J", "classical"),
location = c("median", "mean"), line.it = FALSE,
line.type = c("45 degrees", "QQ"), col.line = 1, lwd = 1,
outliers=FALSE, alpha=0.05, ...)
### Arguments

- **y**: the input data.
- **plot.it**: logical. Should the result be plotted?
- **square.it**: Logical. Should the plot scales be square? True is the default.
- **scale**: the choice of a scale estimator, i.e. the classical or robust estimate of the standard deviation.
- **location**: the choice of a location estimator, i.e. the mean or median.
- **line.it**: logical. Should the line be plotted? No line is the default.
- **line.type**: If line.it=TRUE, the choice of a line to be plotted, i.e. the 45 degree line or the line passing through the first and third quartiles of the data.
- **col.line**: the color of the line (if plotted).
- **lwd**: the line width (if plotted).
- **outliers**: logical. Should the outliers be listed in the output?
- **alpha**: significance level of outliers. If outliers=TRUE, then all observations that are less than the 100*alpha-th standard normal percentile or greater than the 100*(1-alpha)-th standard normal percentile will be listed in the output.
- **...**: Other parameters from `plot`.

### Details

An RQQ plot is a modified QQ plot where data are robustly standardized by the median and robust measure of spread (rather than mean and classical standard deviation as in the basic QQ plots) and then are plotted against the expected standard normal order statistics (see Gel, Miao and Gastwirth, 2005). Under normality, the plot of the standardized observations should follow the 45 degrees line, or QQ line. Both the median and robust standard deviation are significantly less sensitive to outliers than mean and classical standard deviation and therefore are more preferable in many practical situations to assess graphically deviations from normality (if any). We choose median and MAD as a robust measure of location and spread for our RQQ plots since this standardization typically provides a clearer graphical diagnostics of normality. In particular, deviations from the QQ line are usually more noticeable in RQQ plots in the case of outliers and heavy tails. Users can also choose to plot the "45 Degrees" line or the "1st and 3rd Quantile" line. No line is default.

### Value

A list with the following numeric components.

- **x**: The x coordinates of the points that were/would be plotted.
- **y**: The original 'y' vector, i.e., the corresponding y coordinates including 'NA's.

### Author(s)

W. Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao
References


See Also

`rjb.test, sj.test, qqnorm, qqplot, qline`

Examples

```r
## Simulate 100 observations: using rnorm() for
## normally distributed data, Y=N(0,1)
y = rnorm(100)
rqq(y)

## Using michigan data
data(michigan)
rqq(michigan)
```

runs.test  

### Runs Test for Randomness

This function performs the runs test for randomness. Users can choose whether to plot the correlation graph or not, and whether to test against two-sided, negative or positive correlation. NAs from the data are omitted.

Usage

```r
runs.test(y, plot.it = FALSE, alternative = c("two.sided",
        "positive.correlated", "negative.correlated"))
```

Arguments

- `y`: a numeric vector of data values.
- `plot.it`: logical flag. If 'TRUE', then the graph will be plotted. If 'FALSE', then it is not plotted.
- `alternative`: a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "negative.correlated" or "positive.correlated".
Details

On the graph observations which are less than the sample median are represented by letter "A" in red color, and observations which are greater or equal to the sample median are represented by letter "B" in blue color.

Value

A list with the following components.

- `statistic` the value of the standardized Runs statistic.
- `p.value` the p-value for the test.
- `data.name` a character string giving the names of the data.
- `alternative` a character string describing the alternative hypothesis.

Author(s)

Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


See Also

- `bartels.test`

Examples

```r
## Simulate 100 observations from an autoregressive model
## of the first order (AR(1))
y = arima.sim(n = 100, list(ar = c(0.5)))

## Test y for randomness
runs.test(y)

## Sample Output
##
## Runs Test - Two sided
## data: y
## Standardized Runs Statistic = -2.8142, p-value = 0.004889
```
sj.test  

Test of Normality - SJ Test

Description
This function performs the robust directed test of normality which is based on the ratio of the classical standard deviation $s$ to the robust standard deviation $J$ (Average Absolute Deviation from the Median (MAAD)) of the sample data.

Usage
```
sj.test(x, crit.values = c("t.approximation", "empirical"), N = 0)
```

Arguments
- `x` a numeric vector of data values.
- `crit.values` a character string specifying how the critical values should be obtained, i.e. approximated by the t-distribution (default) or empirically.
- `N` number of Monte Carlo simulations for the empirical critical values

Value
A list with the following numeric components.
- `statistic` the standardized test statistic
- `p.value` the p-value.
- `parameter` the ratio of the classical standard deviation $S$ to the robust standard deviation $J$.
- `data.name` a character string giving the name of the data.

Author(s)
Wallace Hui, Yulia R. Gel, Joseph L. Gastwirth, Weiwen Miao

References


See Also
`rqq`, `rjb.test`, `jarque.bera.test` (in `tseries` package)
Examples

```r
data(bias)
sj.test(bias)
```

```r
## Test of Normality - SJ Test
##
## data:  bias
## Standardized SJ Statistic = 2.5147, ratio of S to J = 1.068, p-value = 0.0216
```

```r
symmetry.test(x = c("MGG", "CM", "M"),
               option = c("both", "left", "right"),
               side = c("both", "left", "right"),
               boot = TRUE, B = 1000, q = 8/9)
```

**Description**

This function performs test for symmetry about an unknown median. Users can choose between the Cabilio-Masaro test (Cabilio and Masaro, 1996), the Mira test (Mira, 1999), or the MGG test (Miao, Gel, and Gastwirth, 2006); and using asymptotic distribution of respective statistics or a distribution from \( m \)-out-of-\( n \) bootstrap. Additionally to the general distribution asymmetry, the function allows to test for negative or positive skeweness (see the argument `side`). NAs from the data are omitted.

**Usage**

```r
symmetry.test(x, option = c("MGG", "CM", "M"),
              side = c("both", "left", "right"),
              boot = TRUE, B = 1000, q = 8/9)
```

**Arguments**

- `x`: data to be tested for symmetry.
- `option`: test statistic to be applied. Options include statistic by Miao, Gel, and Gastwirth (2006) (default), Cabilio and Masaro (1996), and by Mira (1999).
- `side`: choice from the three possible alternative hypotheses: general distribution asymmetry (`side = "both"`, default), left skewness (`side = "left"`), or right skewness (`side = "right"`).
- `boot`: logical value indicates whether \( m \)-out-of-\( n \) bootstrap will be used to obtain critical values (default), or asymptotic distribution of the chosen statistic.
- `B`: number of bootstrap replications to perform (default is 1000).
- `q`: scalar from 0 to 1 to define a set of possible \( m \) for the \( m \)-out-of-\( n \) bootstrap. Default \( q = 8/9 \). Possible \( m \) are then set as the values unique(\( \text{round}(n^{*}(q^{*}j)) \)) greater than 4, where \( n = \text{length}(x) \) and \( j = c(0:20) \).

**Details**

If the bootstrap option is used (\( \text{boot} = \text{TRUE} \)), a bootstrap distribution is obtained for each candidate subsample size \( m \). Then, a heuristic method (Bickel et al., 1997; Bickel and Sakov, 2008) is used for the choice of optimal \( m \). Particularly, we use the Wasserstein metric (Ruschendorf, 2001) to calculate distances between different bootstrap distributions and select \( m \), which corresponds to the minimal distance.
Value

A list of class htest containing the following components:

- **method**: name of the method.
- **data.name**: name of the data.
- **statistic**: value of the test statistic.
- **p.value**: \( p \)-value of the test.
- **alternative**: alternative hypothesis.
- **estimate**: bootstrap optimal \( m \) (given in the output only if bootstrap was used, i.e., \( \text{boot} = \text{TRUE} \)).

Author(s)

Joseph L. Gastwirth, Yulia R. Gel, Wallace Hui, Vyacheslav Lyubchich, Weiwen Miao, Xingyu Wang (in alphabetical order)

References


Examples

```r
data(zuni)
symmetry.test(zuni[,"Revenue"], boot = FALSE)

## Symmetry test by Miao, Gel, and Gastwirth (2006)
##
## data:  zuni[, "Revenue"]
## Test statistic = 5.0321, p-value = 4.851e-07
## alternative hypothesis: the distribution is asymmetric.
```
The Zuni data from the law case: Zuni Public School v. United States Department of Education

Description
Number of students and available revenue per pupil in each school district in New Mexico

Usage
zuni

Format
A multivariate time series with 89 observations on 4 variables: District, Revenue and Mem (number of students). The object is of class "mts".

Details
The Zuni data come from a law case "The Zuni Public School District No. 89, Gallup-McKinley County Public School District No. 1, Petitioners v. United States Department of Education" concerning whether the revenue per pupil data satisfied a standard for "equal" expenditures per-pupil in a state. This classification determines whether most of the federal money given to the state under the law goes to the state or to the local school districts.

Source

Examples
data(zuni)
hist(zuni[,"Revenue"], br=40, col="blue", main="The Zuni Revenue")
Index

*Topic datasets
  bias, 3
  blackhire, 4
  data1963, 9
  michigan, 24
  popdata, 30
  pot, 31
  zuni, 43

*Topic distribution
  bartels.test, 2
  j.maad, 11
  runs.test, 38
  symmetry.test, 41

*Topic dplot
  lorenz.curve, 19

*Topic htest
  bartels.test, 2
  brunner.munzel.test, 4
  cd, 6
  cmh.test, 7
  gini.index, 6, 10
  laplace.test, 12
  levene.test, 13
  lnested.test, 15
  ltrend.test, 15
  mma.test, 15
  neuhauser.hothorn.test, 15
  nig.parameter, 29
  popdata, 30
  pot, 31
  rjb.test, 12, 32
  rlm.test, 33
  robust.mmm.test, 35
  runs.test, 38
  sj.test, 40
  symmetry.test, 41

*Topic nonparametric
  brunner.munzel.test, 4

*Topic robust
  rqq, 36

*Topic ts
  sj.test, 40
  bartels.test, 2, 39
  bias, 3
  blackhire, 4
  brunner.munzel.test, 4
  cd, 6, 10, 12
  cmh.test, 7
  data1963, 9
  gini.index, 6, 10, 12, 20
  j.maad, 6, 10, 11
  laplace.test, 12
  levene.test, 13, 19, 23, 26, 28, 36
  lnested.test, 15, 16, 23, 26, 28, 36
  lorenz.curve, 10, 19
  ltrend.test, 15, 19, 21, 26, 28, 36
  michigan, 24
  mma.test, 15, 19, 23, 25, 28, 36
  neuhauser.hothorn.test, 15, 19, 23, 26, 28, 36
  nig.parameter, 29
  popdata, 30
  pot, 31
  rjb.test, 12, 32, 34, 38, 40
  rlm.test, 33
  robust.mmm.test, 15, 19, 23, 26, 28, 36
  rqq, 12, 33, 34, 36, 40
  runs.test, 3, 38
  sj.test, 12, 33, 34, 38, 40
  symmetry.test, 41
  zuni, 43