Package ‘PairedData’

February 19, 2015

Type Package
Title Paired Data Analysis
Version 1.0.1
Date 2013-04-18
Author Stephane Champely <champely@univ-lyon1.fr>
Maintainer Stephane Champely <champely@univ-lyon1.fr>
Description This package provides many datasets and a set of graphics (based on ggplot2), statistics, effect sizes and hypothesis tests for analysing paired data with S4 class.
License GPL (>= 2)
Depends methods, graphics, MASS, gld, mvtnorm, lattice, ggplot2
Collate global1.R ClassP1.R
NeedsCompilation no
Repository CRAN
Date/Publication 2013-04-19 07:43:41

R topics documented:

PairedData-package ........................................ 2
Anorexia ..................................................... 3
anscombe2 ................................................ 4
Barley ....................................................... 5
Blink ........................................................ 6
Blink2 ....................................................... 7
BloodLead .................................................. 8
bonettseier.var.test ..................................... 9
ChickWeight ................................................ 10
Corn ......................................................... 11
Datalcoholic ............................................... 12
effect.size ............................................... 13
GDO ........................................................ 14
Grain ....................................................... 15
Grain2 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
grambsch.var.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
GrapeFruit . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
HorseBeginners . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19
IceSkating . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
imam.var.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21
Iron . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
lambda.table . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
levene.var.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
mcculloch.var.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
Meat . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27
paired . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28
paired-class . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29
paired.plotBA . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30
paired.plotCor . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31
paired.plotMcNeil . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32
paired.plotProfiles . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 33
plot . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 34
PrisonStress . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 35
rpaired.contaminated . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 36
rpaired.gld . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37
Rugby . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38
sandvikolsson.var.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38
Sewage . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 40
Shoulder . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 41
SkiExperts . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42
Sleep . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43
slidingchart . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43
summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 44
t.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 46
Tobacco . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 47
var.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48
wilcox.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49
winsor.cor.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51
yuen.t.test . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 52

Index

PairedData-package

A package for visualising and analysing paired data.

Description

A package for visualising and analysing paired (mainly numeric) data with S4 class.
**Details**

- **Package:** PairedData
- **Type:** Package
- **Version:** 1.0.1
- **Date:** 2013-04-19
- **License:** GPL
- **Depends:** methods

**Author(s)**

Stephane Champely

Maintainer: Stephane Champely<champely@univ-lyon1.fr>

---

**Anorexia**

*Anorexia data from Pruzek & Helmreich (2009)*

**Description**

This dataset presents 17 paired data corresponding to the weights of girls before and after treatment for anorexia. A more complete version can be found in the package MASS. There is actually a cluster of four points in this dataset.

**Usage**

```r
data(Anorexia)
```

**Format**

A dataframe with 17 rows and 2 numeric columns:

```
[,1] Prior numeric weight (lbs) before therapy
[,2] Post numeric weight (lbs) after therapy
```

**Source**


**References**

See Also

anorexia in MASS

Examples

data(Anorexia)

# Visualization of the cluster
with(Anorexia, plot(paired(Prior, Post), type="profile"))

# The effects of trimming or winsorizing
# with 4 outliers (n=17)
17*0.2
with(Anorexia, summary(paired(Prior, Post)))
17*0.25
with(Anorexia, summary(paired(Prior, Post), tr=0.25))

Description

This dataset presents four sets of paired samples (n=15), giving the same t statistic (t=2.11) and thus the same p-value whereas their situations are really diversified (differences in variances, clustering, heteroscedasticity). The importance of plotting data is thus stressed. The name is given from the famous Anscombe’s dataset created to study simple linear regression.

Usage

data(antscombe2)

Format

A dataframe with 15 rows, 8 numeric columns of paired data: (X1,Y1) ; (X2,Y2) ; (X3,Y3) ; (X4,Y4), and 1 factor column: Subjects, giving a label for the subjects.

Source

S. Champely, CRIS, Lyon 1 University, FRANCE

References

Examples

```r
data(anscombe2)
# p<0.05 for the paired t-test
with(anscombe2, plot(paired(X1,Y1), type="BA"))
with(anscombe2, t.test(paired(X1,Y1)))

# Same p but Var(X2)<Var(Y2) and correlation in the Bland-Altman plot
with(anscombe2, t.test(paired(X2,Y2)))
with(anscombe2, summary(paired(X2,Y2)))
with(anscombe2, plot(paired(X2,Y2), type="BA"))

# Same p but two clusters
with(anscombe2, plot(paired(X3,Y3), type="BA"))

# Same p but the difference is "linked" to the mean
with(anscombe2, plot(paired(X4,Y4), type="BA"))
```

**Description**

This dataset presents 12 paired data corresponding to the yields of Glabron and Velvet Barley, grown on different farms. The values from farm 12 are quite different.

**Usage**

data(Barley)

**Format**

A dataframe with 17 rows and 3 columns:

```
> Barley
       Glabron Velvet
[1,] Farm         factor numeric yields
[2,] Glabron numeric yields (bushels per acre)
[3,] Velvet numeric yields
```

**Source**


**References**

Examples

data(Barley)

# Visualizing a clear outlier
with(Barley, plot(paired(Glabron, Velvet), type="BA"))

# Results form the paired t test and paired Yuen test are similar
with(Barley, t.test(paired(Glabron, Velvet)))
with(Barley, yuen.t.test(paired(Glabron, Velvet)))

# Nevertheless the outlier inflates the location (numerator) and
# scale (denominator) standard statistics for the difference
with(Barley, summary(paired(Glabron, Velvet)))

Blink data from Preece (1982, Table 2)

Description

This dataset presents paired data corresponding to average blink-rate per minute of 12 subjects in an experiment of a visual motor task. They had to steer a pencil along a moving track. Each subject was tested under two conditions: a straight track and an oscillating one. Note that the values from subjects 1 and 2 are somewhat different.

Usage

data(Blink)

Format

A dataframe with 12 rows and 3 columns:

```
[,1] Subject factor
[,2] Straight numeric  blink rate in first condition
[,3] Oscillating numeric blink rate in second condition
```

Source


References

Examples

```r
data(Blink)

# Visualizing two "outliers"
with(Blink, plot(paired(Straight, Oscillating), type="profile"))

# Interestingly, the differences for the two outliers are quite "normal"
# so their influence on the t test is negligible
with(Blink, qqnorm(Straight-Oscillating))
with(Blink, qqline(Straight-Oscillating))
```

**Blink2**

*Blink data (2nd example) from Preece (1982, Table 3)*

**Description**

This dataset presents paired data corresponding to average blink-rate per minute of 12 subjects in an experiment of a visual motor task. They had to steer a pencil along a moving track. Each subject was tested under two conditions: a straight track and an oscillating one. Data about blink-rate during a pre-experimental resting are also available. Subjects 1 and 2 then appear less extreme than in the dataset Blink.

**Usage**

data(Blink2)

**Format**

A dataframe with 12 rows and 4 columns:

<table>
<thead>
<tr>
<th></th>
<th>Subject factor</th>
<th>Resting numeric</th>
<th>Straight numeric</th>
<th>Oscillating numeric</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Subject</td>
<td>Resting</td>
<td>Straight</td>
<td>Oscillating</td>
</tr>
<tr>
<td>2</td>
<td>factor</td>
<td>numeric</td>
<td>numeric</td>
<td>numeric</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>(blink rate in pre-experimental condition)</td>
<td>(blink rate in first condition)</td>
<td>(blink rate in second condition)</td>
</tr>
</tbody>
</table>

**Source**


**References**

BloodLead

Blood lead levels data from Pruzek & Helmreich (2009)

Description
This dataset presents matched paired data corresponding to blood lead levels for 33 children of parents who had worked in a lead related factory and 33 control children from their neighborhood. The two samples have different dispersions and their correlation is small.

Usage
data(BloodLead)

Format
A dataframe with 33 rows and 3 columns:

[.1] Pair factor matched pair of children
[.2] Exposed numeric blood lead levels (mg/dl) for exposed children
[.3] Control numeric blood lead levels for controls

Source

References

Examples
data(BloodLead)

# Control values are clearly less dispersed (and inferior)
# than exposed levels
with(BloodLead,plot(paired(Control,Exposed),type="McNeil"))
with(BloodLead,var.test(paired(Control,Exposed)))
with(BloodLead,grambsch.var.test(paired(Control,Exposed)))
with(BloodLead,bonettseier.var.test(paired(Control,Exposed)))

# Correlation is small (bad matching)
with(BloodLead,cor.test(Control,Exposed))
Bonett-Seier test of scale for paired samples

Description
Robust test of scale for paired samples based on the mean absolute deviations.

Usage
bonettseier.var.test(x, ...)

## Default S3 method:
bonettseier.var.test(x, y = NULL, alternative = c("two.sided", "less", "greater"), omega = 1, conf.level = 0.95,...)

## S3 method for class 'paired'
bonettseier.var.test(x, ...)

Arguments
x first sample or object of class paired.
y second sample.
alternative alternative hypothesis.
omega a priori ratio of means absolute deviations.
conf.level confidence level.
... further arguments to be passed to or from methods.

Value
A list with class "htest" containing the following components:
statistic the value of the z-statistic.
p.value the p-value for the test.
conf.int a confidence interval for the ratio of means absolute deviations appropriate to the specified alternative hypothesis.
estimate the estimated means absolute deviations.
null.value the specified hypothesized value of the ratio of means absolute deviations.
alternative a character string describing the alternative hypothesis.
method a character string indicating what type of test was performed.
data.name a character string giving the name(s) of the data.
Author(s)
Stephane CHAMPELY

References

See Also
var.test, grambsch.var.test

Examples
z<-rnorm(20)
x<-rnorm(20)+z
y<-(rnorm(20)+z)*2
bonettseier.var.test(x,y)

data(anscombe2)
p<--with(anscombe2,paired(X1,Y1))
bonettseier.var.test(p)

ChickWeight  Chick weight data from Preece (1982, Table 11)

Description
This dataset presents 10 paired data corresponding to the weights of chicks, two from ten families, reared in confinement or on open range.

Usage
data(ChickWeight)

Format
A dataframe with 10 rows and 3 columns:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Chicks</td>
<td>factor</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Confinement</td>
<td>numeric</td>
<td>chick weight (ounces)</td>
</tr>
<tr>
<td>3</td>
<td>OpenRange</td>
<td>numeric</td>
<td>chick weight</td>
</tr>
</tbody>
</table>

Source
References


Examples

data(Corn)

# Look at the interesting discussion in Preece (1982)
# about degree of precision and t test
with(Corn,plot(paired(Confinement,OpenRange)))
with(Corn,stem(Confinement~OpenRange, scale=2))

Corn data (Darwin)

Description

This dataset presents 15 paired data corresponding to the final height of corn data (Zea Mays), one produced by cross-fertilization and the other by self-fertilization. These data were used by Fisher (1936) and were published in Andrews and Herzberg (1985).

Usage

data(Corn)

Format

A dataframe with 15 rows and 4 columns:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>[.1]</td>
<td>pair</td>
<td>numeric</td>
</tr>
<tr>
<td>[.2]</td>
<td>pot</td>
<td>numeric</td>
</tr>
<tr>
<td>[.3]</td>
<td>Crossed</td>
<td>numeric</td>
</tr>
<tr>
<td>[.4]</td>
<td>Self</td>
<td>numeric</td>
</tr>
</tbody>
</table>

Source


References

Examples

```r
data(Corn)

# Visualizing two outliers
with(Corn, slidingchart(paired(Crossed, Self)))

# Very bad matching in these data
with(Corn, cor.test(Crossed, Self))
with(Corn, winsor.cor.test(Crossed, Self))

# So the two-sample test is slightly
# more interesting than the paired test
with(Corn, t.test(Crossed, Self, var.equal=TRUE))
with(Corn, t.test(Crossed, Self, paired=TRUE))

# The Pitman-Morgan test is influenced by the two outliers
with(Corn, var.test(paired(Crossed, Self)))
with(Corn, grambisch.var.test(paired(Crossed, Self)))
with(Corn, bonettseier.var.test(paired(Crossed, Self)))

# Lastly, is there a pot effect?
with(Corn, plot(paired(Crossed, Self)))
with(Corn, plot(paired(Crossed, Self), group=pot))
```

---

**Datalcoholic**

**Datalcoholic: a dataset of paired datasets**

**Description**

This dataset presents for teaching purposes 50 paired datasets available in different R packages.

**Usage**

```r
data(Datalcoholic)
```

**Format**

A dataframe with 4 columns.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[,1]</td>
<td>Dataset</td>
</tr>
<tr>
<td>[,2]</td>
<td>Package</td>
</tr>
<tr>
<td>[,3]</td>
<td>Topic</td>
</tr>
<tr>
<td>[,4]</td>
<td>NumberPairs</td>
</tr>
</tbody>
</table>

**Examples**

```r
data(Datalcoholic)
```
show(Dataalcoholic)

description

Robust and classical effects sizes for paired samples of the form: \((M_x-M_y)/S\) where \(M_x\) and \(M_y\) are location parameters for each sample and \(S\) is a scale parameter.

Usage

```r
## S4 method for signature 'paired'
effect.size(object, tr=0.2)
```

Arguments

- `object`: an object of class `paired`
- `tr`: percentage of trimming

Value

A table with two rows corresponding to classical (means) and robust (trimmed means, \(tr=0.2\)) delta-type effect sizes. The four columns correspond to:

- **Average**: Numerator is the difference in (trimmed) means, denominator is the average of the two (winsorised and rescaled to be consistent with the standard deviation when the distribution is normal) standard deviations.
- **Single \((x)\)**: Denominator is the (winsorised and rescaled) standard deviation of the first sample.
- **Single \((y)\)**: Denominator is the (winsorised and rescaled) standard deviation of the second sample.
- **Difference**: Numerator is the (trimmed) mean and denominator the (winsorised and rescaled) standard deviation of the differences \((x-y)\).

Author(s)

Stephane CHAMPELY

References

Examples

```
z <- rnorm(20)
x <- rnorm(20) + z
y <- rnorm(20) + z + 1
p <- paired(x, y)
effect.size(p)
```

GDO  

**Agreement study**

Description

This dataset gives the same measurements of muscle activation (EMG) in 3 days corresponding to a reproductibility study for 18 tennis players.

Usage

```
data(GDO)
```

Format

A dataframe with 18 rows and 4 columns.

<table>
<thead>
<tr>
<th></th>
<th>Subject</th>
<th>factor</th>
<th>anonymous subjects</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Day1</td>
<td>numeric</td>
<td>measurement first day</td>
</tr>
<tr>
<td>2</td>
<td>Day2</td>
<td>numeric</td>
<td>measurement second day</td>
</tr>
<tr>
<td>3</td>
<td>Day3</td>
<td>numeric</td>
<td>measurement third day</td>
</tr>
</tbody>
</table>

Source

Private communication. Samuel Rota, CRIS, Lyon 1 University, FRANCE

See Also

packages: agreement, irr and MethComp.

Examples

```
data(GDO)

# Building new vectors for performing
# a repeated measures ANOVA
# with a fixed Day effect
Activation <- c(GDO[,2], GDO[,3], GDO[,4])
Subject <- factor(rep(GDO[,1], 3))
Day <- factor(rep(c("D1", "D2", "D3"), rep(18, 3)))
aovGDO <- aov(Activation ~ Day + Error(Subject))
```
Grain data from Preece (1982, Table 5)

**Description**

This dataset presents 9 paired data corresponding to the grain yields of Great Northern and Big Four oats grown in "adjacent" plots.

**Usage**

data(Grain)

**Format**

A dataframe with 9 rows and 3 columns:

<table>
<thead>
<tr>
<th>.1</th>
<th>Year factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>.2</td>
<td>GreatNorthern numeric</td>
</tr>
<tr>
<td>.3</td>
<td>BigFour numeric</td>
</tr>
</tbody>
</table>

**Source**


**References**


**Examples**

data(Grain)

# Usual visualization for paired data (2 clusters?)
with(Grain, plot(paired(GreatNorthern,BigFour)))

# Are they actually "adjacent" plots?
# Why this variable Year?
# Is there any time trend?
with(Grain, plot(Year,GreatNorthern,type="o"))
with(Grain, plot(Year,BigFour,type="o"))
Description

This dataset presents 6 paired data corresponding to the grain yields of two wheat varieties grown on pairs of plots.

Usage

data(Grain2)

Format

A dataframe with 6 rows and 3 columns:

<table>
<thead>
<tr>
<th>.1</th>
<th>Plot factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>.2</td>
<td>Variety_1 numeric grain yield (bushels per acre)</td>
</tr>
<tr>
<td>.3</td>
<td>Variety_2 numeric grain yield</td>
</tr>
</tbody>
</table>

Source


References


Examples

data(Grain2)

# A very small data set
print(Grain2)

# The paired t test is the test of the differences
with(Grain2,t.test(Variety_1,Variety_2,paired=TRUE))
with(Grain2,t.test(Variety_1-Variety_2))

# The data are actually rounded to the nearest integer
# So they can be somewhere between +0.5 or -0.5
# and thus the differences between +1 or -1
# The possible t values can be simulated by:
simulating.t<-numeric(1000)
for(i in 1:1000){
simulating.t[i]<-with(Grain2,t.test(Variety_1-Variety_2+runif(6,-1,1)))$stat}
Description

Robust test of scale for paired samples.

Usage

grambsch.var.test(x, ...)

## Default S3 method:
grambsch.var.test(x, y = NULL, alternative = c("two.sided", "less", "greater"), ...)

## S3 method for class 'paired'
grambsch.var.test(x, ...)

Arguments

x first sample or an object of class paired.
y second sample.
alternative alternative hypothesis.
... further arguments to be passed to or from methods.

Details

Denoting $s=x+y$ and $d=x-y$, the test proposed by Grambsch (1994, and called by the author 'modified Pitman test') is based on the fact that $\text{var}(x)-\text{var}(y)=\text{cov}(x+y,x-y)=\text{cov}(s,d)$. The values $z=(s-\text{mean}(s))(d-\text{mean}(d))$ can be tested for null expectation using a classical t test in order to compare the two variances. Note that the p value is computed using the normal distribution.

Value

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

- statistic the value of the F-statistic.
- p.value the p-value for the test.
- null.value the specified hypothesized value of the ratio of variances (=1!)
- alternative a character string describing the alternative hypothesis.
- method a character string indicating what type of test was performed.
- data.name a character string giving the name(s) of the data.
GrapeFruit

Author(s)
Stephane CHAMPELY

References

See Also
var.test, bonettseier.var.test

Examples
z<-rnorm(20)
x<-rnorm(20)+z
y<-rnorm(20)+z*2
grambsch.var.test(x,y)
p<-paired(x,y)
grambesch.var.test(p)

GrapeFruit
Grape Fruit data from Preece (1982, Table 6)

Description
This dataset presents paired data corresponding to the percentage of solids recorded in the shaded and exposed halves of 25 grapefruits.

Usage
data(GrapeFruit)

Format
A dataframe with 25 rows and 3 columns:

<table>
<thead>
<tr>
<th></th>
<th>Fruit</th>
<th>numeric</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Shaded</td>
<td>numeric</td>
</tr>
<tr>
<td>2</td>
<td>Exposed</td>
<td>numeric</td>
</tr>
</tbody>
</table>

Source
References


Examples

```r
data(GrapeFruit)

# Visualizing a very strange paired distribution
with(GrapeFruit, plot(paired(Shaded, Exposed)))
with(GrapeFruit, plot(paired(Shaded, Exposed), type="BA"))
with(GrapeFruit, plot(paired(Shaded, Exposed), type="McNeil"))
with(GrapeFruit, plot(paired(Shaded, Exposed), type="profile"))

# As underlined by Preece (1982), have a look to
# the distribution of the final digits
show(GrapeFruit)
table(round((GrapeFruit$Shaded*10-floor(GrapeFruit$Shaded*10))*10))
table(round((GrapeFruit$Exposed*10-floor(GrapeFruit$Exposed*10))*10))
```

---

**HorseBeginners**  
*Actual and imaginary performances in equitation*

Description

This dataset gives the actual and motor imaginary performances (time) in horse-riding for 8 beginners.

Usage

```r
data(HorseBeginners)
```

Format

A dataframe with 8 rows and 3 columns.

```
[1,] Subject factor Anonymous subjects
[2,] Actual numeric Actual performance (sec.)
[3,] Imaginary numeric Imaginary performance (sec.)
```

Source

Private communication. Aymeric Guillot, CRIS, Lyon 1 University, FRANCE.
IceSkating

References


Examples

data(HorseBeginners)

# There is one outlier
with(HorseBeginners, plot(paired(Actual, Imaginary), type="profile"))

# This outlier has a great influence
# on the non robust Pitman-Morgan test of variances
with(HorseBeginners, var.test(paired(Actual, Imaginary)))
with(HorseBeginners[-1,], var.test(paired(Actual, Imaginary)))
with(HorseBeginners, grambisch.var.test(paired(Actual, Imaginary)))
with(HorseBeginners, bonetttseier.var.test(paired(Actual, Imaginary)))

---

IceSkating  Ice skating speed study

Description

This dataset gives the speed measurement (m/sec) for seven iceskating dancers using the return leg in flexion or in extension.

Usage

data(IceSkating)

Format

A dataframe with 7 rows and 3 columns.

| .1 | Subject factor anonymous subjects |
| .2 | Extension numeric speed when return leg in extension (m/sec) |
| .3 | Flexion numeric speed when return leg in flexion (m/sec) |

Source

Private communication. Karine Monteil, CRIS, Lyon 1 University, FRANCE.

References

Examples

data(IceSkating)

# Nothing particular in the paired plot
with(IceSkating,plot(paired(Extension,Flexion),type="McNeil"))

# The differences are normally distributed
with(IceSkating,qqnorm(Extension-Flexion))
with(IceSkating,qqline(Extension-Flexion))

# Usual t test
with(IceSkating,t.test(paired(Extension,Flexion)))


imam.var.test

Imam test of scale for paired samples

Description

Robust test of scale for paired samples based on absolute deviations from the trimmed means (or medians), called Imam test in Wilcox (1989).

Usage

imam.var.test(x, ...)

### Default S3 method:
imam.var.test(x, y = NULL,
alternative = c("two.sided", "less", "greater"),
mu = 0, conf.level = 0.95, location=c("trim","median"),
tr=0.1, ...)

### S3 method for class 'paired'
imam.var.test(x, ...)

Arguments

x first sample or object of class paired.
y second sample.
alternative alternative hypothesis.
mu the location parameter mu.
conf.level confidence level.
location location parameter for centering: trimmed mean or median.
tr percentage of trimming.
... further arguments to be passed to or from methods.
Details

The data are transformed as deviations from the trimmed mean: \( X = \text{abs}(x - \text{mean}(x, \text{tr}=0.1)) \) and \( Y = (y - \text{mean}(y, \text{tr}=0.1)) \). A paired t test is then carried out on the (global) ranks of \( X \) and \( Y \).

Value

A list with class "htest" containing the components of a paired t test.

Author(s)

Stephane CHAMPELY

References


See Also

var.test, grambsch.var.test

Examples

```r
z <- rnorm(28)
x <- rnorm(28) + z
y <- (rnorm(28) + z) * 2
imam.var.test(x, y)

# some variations
imam.var.test(x, y, tr=0.2)
imam.var.test(x, y, location="median")

data(anscombe2)
p <- with(anscombe2, paired(X1, Y1))
imam.var.test(p)
```

Iron

Iron data from Preece (1982, Table 10)

Description

This dataset presents 10 paired data corresponding to percentages of iron found in compounds with the help of two different methods (take a guess: A & B). It is quite interesting to study rounding effect on hypothesis test (have a look at the examples section).
Usage

data(Iron)

Format

A dataframe with 10 rows and 3 columns:

<table>
<thead>
<tr>
<th>[,1]</th>
<th>Compound</th>
</tr>
</thead>
<tbody>
<tr>
<td>[,2]</td>
<td>Method_A</td>
</tr>
<tr>
<td>[,3]</td>
<td>Method_B</td>
</tr>
<tr>
<td></td>
<td>factor</td>
</tr>
<tr>
<td></td>
<td>numeric</td>
</tr>
<tr>
<td></td>
<td>percentage of iron</td>
</tr>
</tbody>
</table>

Source


References


Examples

data(Iron)

# Visualizing, very nice correlation
# Is this an agreement problem or a comparison problem?
with(Iron, plot(paired(Method_A, Method_B)))

# Significant... p=0.045
with(Iron, t.test(paired(Method_A, Method_B)))

# Looking at data, rounded at 0.1 so they can be +0.05 or -0.05
show(Iron)

# Thus the differences can be +0.1 or -0.1
# Influence of rounding on the t-statistic
with(Iron, t.test(Method_A - Method_B + 0.1))
with(Iron, t.test(Method_A - Method_B - 0.1))

Parameters for Generalised Lambda Distributions

This dataset gives the parameters for specific 8 Generalized Tukey-lambda distributions with zero mean and unit variance useful for simulation studies as given in Bonett and Seier (2003).
Usage

data(lambda.table)

Format

A dataframe with 8 rows (distributions) and 4 columns (parameters).

References


levene.var.test  Levene test of scale for paired samples

Description

Robust test of scale for paired samples based on absolute deviations from the trimmed means (or medians), called extended Brown-Forsythe test in Wilcox (1989).

Usage

levene.var.test(x, ...)

## Default S3 method:
levene.var.test(x, y = NULL,
    alternative = c("two.sided", "less", "greater"),
    mu = 0, conf.level = 0.95, location = c("trim","median"),
    tr=0.1, ...)

## S3 method for class 'paired'
levene.var.test(x, ...)

Arguments

x  first sample or object of class paired.
y  second sample.
alternative  alternative hypothesis.
mu  the location parameter mu.
conf.level  confidence level.
location  location parameter for centering: trimmed mean or median.
tr  percentage of trimming.
...  further arguments to be passed to or from methods.
Details

The data are transformed as deviations from the trimmed mean: \( X = \text{abs}(x - \text{mean}(x, \text{tr}=0.1)) \) and \( Y = (y - \text{mean}(y, \text{tr}=0.1)) \). A paired t test is then carried out on \( X \) and \( Y \).

Value

A list with class "htest" containing the components of a paired t test.

Author(s)

Stephane CHAMPELY

References


See Also

var.test, grambsch.var.test

Examples

```r
z <- rnorm(20)
x <- rnorm(20) + z
y <- (rnorm(20) + z) * 2
levene.var.test(x, y)

# Some variations
levene.var.test(x, y, tr=0.2)
levene.var.test(x, y, location="median")

data(anscombe2)
p <- with(anscombe2, paired(X2, Y2))
levene.var.test(p)
```

---

mcculloch.var.test

McCulloch test of scale for paired samples

Description

Robust test of scale for paired samples based on spearman coefficient (the default, or kendall or pearson) of the transformed \( D = x - y \) and \( S = x + y \).
Usage

```r
mcculloch.var.test(x, ...)
```

```r
## Default S3 method:
mcculloch.var.test(x, y = NULL,
alternative = c("two.sided", "less", "greater"),method= c("spearman", "pearson", "kendall"),
exact = NULL,conf.level = 0.95,continuity = FALSE, ...)
```

```r
## S3 method for class 'paired'
mcculloch.var.test(x, ...)
```

Arguments

- `x` first sample or object of class paired.
- `y` second sample.
- `alternative` alternative hypothesis.
- `method` a character string indicating which correlation coefficient is to be used for the test. One of "spearman", "kendall", or "pearson", can be abbreviated.
- `exact` a logical indicating whether an exact p-value should be computed.
- `conf.level` confidence level.
- `continuity` logical: if true, a continuity correction is used for Spearman's rho when not computed exactly.
- `...` further arguments to be passed to or from methods.

Value

A list with class "htest" containing the components of a (Spearman) correlation test.

Author(s)

Stephane CHAMPELY

References


See Also

var.test, grambsch.var.test

Examples

```r
z<-rnorm(20)
x<-rnorm(20)+z
y<-(rnorm(20)+z)*2
mcculloch.var.test(x,y)
```
Meat

Meat data from Preece (1982, Table 4)

Description

This dataset presents 20 paired data corresponding to the percentage of fat in samples of meat using two different methods: AOAC and Babcock.

Usage

data(Meat)

Format

A dataframe with 20 rows and 3 columns:

<table>
<thead>
<tr>
<th>.1</th>
<th>.2</th>
<th>.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>AOAC</td>
<td>Babcock</td>
<td>MeatType</td>
</tr>
<tr>
<td>numeric</td>
<td>numeric</td>
<td>factor</td>
</tr>
</tbody>
</table>

percentage of fat

<table>
<thead>
<tr>
<th>.1</th>
<th>.2</th>
<th>.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>AOAC</td>
<td>Babcock</td>
<td>MeatType</td>
</tr>
<tr>
<td>numeric</td>
<td>numeric</td>
<td>factor</td>
</tr>
</tbody>
</table>

Source


References


Examples

data(Meat)

# Presence of clusters or...
with(Meat,plot(paired(AOAC,Babcock)))
# group effect according to Meat type?
with(Meat, plot(paired(AOAC, Babcock), group=MeatType))
with(Meat, plot(paired(AOAC, Babcock), group=MeatType, facet=FALSE))

<table>
<thead>
<tr>
<th>paired</th>
<th>Paired</th>
</tr>
</thead>
</table>

## Description
This function creates objects of class paired.

## Usage
`paired(x, y)`

## Arguments
- `x`: first vector.
- `y`: second vector.

## Details
The two vectors must share the same class. Moreover, for vectors of class factor, they must have the same levels.

## Value
An object of class paired.

## Author(s)
Stephane Champely

## Examples
```r
x<-rnorm(15)
y<-rnorm(15)
p1<-paired(x, y)
show(p1)

data(IceSkating)
p2<-with(IceSkating, paired(Extension, Flexion))
show(p2)
```
paired-class

Class "paired"

Description

An object of class paired is a dataframe with two columns sharing the same class (usually numeric).

Objects from the Class

Objects can be created by calls of the form `new("paired", ...)`. 

Slots

- `.Data`: Object of class "list" ~~
- `names`: Object of class "character" ~~
- `row.names`: Object of class "data.frameRowLabels" ~~
- `.S3Class`: Object of class "character" ~~

Extends

Class "data.frame", directly. Class "list", by class "data.frame", distance 2. Class "oldClass", by class "data.frame", distance 2. Class "vector", by class "data.frame", distance 3.

Methods

- `effect.size` signature(object = "paired"): ...
- `summary` signature(object = "paired"): ...
- `plot` signature(object = "paired"): ...

Author(s)

Stephane Champely

Examples

data(IceSkating)
p <- with(IceSkating, paired(Extension, Flexion))
show(p)
plot(p)
summary(p)
effect.size(p)
paired.plotBA  

*Bland-Altman plot*

**Description**

Produce a Bland-Altman plot for paired data, including a confidence region for the mean of the differences.

**Usage**

```r
paired.plotBA(df, condition1, condition2, groups = NULL,
              facet = TRUE, ...)
```

**Arguments**

- `df`: a data.frame.
- `condition1`: name of the variable corresponding to the first sample.
- `condition2`: name of the variable corresponding to the first sample.
- `groups`: name of the variable corresponding to the groups (optional).
- `facet`: faceting or grouping strategy for plotting?
- `...`: arguments to be passed to methods

**Value**

a graphical object of class ggplot.

**Author(s)**

Stephane CHAMPELY

**References**


**See Also**

tmd

**Examples**

```r
data(PrisonStress)
pair.plo.plotBA(PrisonStress, "PSSbefore","PSSafter")
```

# Extending the resulting ggplot object by faceting
```r
paired.plotBA(PrisonStress,"PSSbefore","PSSafter")+facet_grid(~Group)
```
paired.plotCor

Paired correlation plot

Description

Produce a squared scatterplot for paired data (same units for both axes), including the first bisector
line for reference.

Usage

paired.plotCor(df, condition1, condition2, groups = NULL,
facets = TRUE, ...)

Arguments

df
condition1
condition2
groups
facets
...

a data.frame.
name of the variable corresponding to the first sample.
name of the variable corresponding to the first sample.
name of the variable corresponding to the groups (optional).
faceting or grouping strategy for plotting?
arguments to be passed to methods

Value

a graphical object of class ggplot.

Author(s)

Stephane CHAMPELY

Examples

data(PrisonStress)
paired.plotCor(PrisonStress,"PSSbefore","PSSafter")

# Changing the theme of the ggplot object
paired.plotCor(PrisonStress,"PSSbefore","PSSafter")+theme_bw()
paired.plotMcNeil

Parallel lines plot

Description

Produce a parallel lines plot for paired data.

Usage

paired.plotMcNeil(df, condition1, condition2, groups = NULL, subjects, facet = TRUE, ...)

Arguments

df  a data frame.
condition1  name of the variable corresponding to the second sample.
condition2  name of the variable corresponding to the first sample.
groups  names of the variable corresponding to groups (optional).
subjects  names of the variable corresponding to subjects.
facet  faceting or grouping strategy for plotting?
...  further arguments to be passed to methods.

Value

a graphical object of class ggplot.

Author(s)

Stephane CHAMPELY

References


See Also

plotBA

Examples

data(PrisonStress)
paired.plotMcNeil(PrisonStress, "PSSbefore", "PSSafter", subjects="Subject")
paired.plotProfiles

Profile plot

Description
Produce a profile plot or before-after plot or 1-1 plot for paired data.

Usage
paired.plotProfiles(df, condition1, condition2, groups = NULL, subjects, facet = TRUE, ...)

Arguments
df a data frame.
condition1 name of the variable corresponding to the second sample.
condition2 name of the variable corresponding to the first sample.
groups names of the variable corresponding to groups (optional).
subjects names of the variable corresponding to subjects.
facet faceting or grouping strategy for plotting?
... further arguments to be passed to methods.

Value
a graphical object of class ggplot.

Author(s)
Stephane CHAMPELY

References

See Also
plotBA, plotMcNeil

Examples
data(PrisonStress)
paired.plotProfiles(PrisonStress, "PSSbefore", "PSSafter", subjects = "Subject", groups = "Group")

# Changing the line colour
paired.plotProfiles(PrisonStress, "PSSbefore", "PSSafter") + geom_line(colour = "red")
plot

--- Methods for Function plot ---

Description

Plot an object of class paired.

Usage

```r
## S4 method for signature 'paired'
plot(x, groups=NULL, subjects=NULL, facet=TRUE, type=c("correlation","BA","McNeil","profile"),...)
```

Arguments

- `x` a paired object created by the `paired` function.
- `groups` a factor (optional).
- `subjects` subjects name.
- `facet` faceting or grouping strategy for plotting?
- `type` type of the plot (correlation, Bland-Altman, McNeil or profile plot).
- `...` arguments to be passed to methods.

Value

an graphical object of class ggplot.

Examples

data(HorseBeginners)
```r
pd1<-with(HorseBeginners,paired(Actual,Imaginary))
plot(pd1)
plot(pd1,type="BA")
plot(pd1,type="McNeil")
plot(pd1,type="profile")
```
data(Shoulder)
```r
with(Shoulder,plot(paired(Left,Right),groups=Group))
with(Shoulder,plot(paired(Left,Right),groups=Group,facet=FALSE))
with(Shoulder,plot(paired(Left,Right),
    groups=Group,facet=FALSE,type="profile")+theme_bw())
```
**PrisonStress**  

**Stress in prison**

---

**Description**

This dataset gives the PSS (stress measurement) for 26 people in prison at the entry and at the exit. Part of these people were physically trained during their imprisonment.

**Usage**

data(PrisonStress)

**Format**

A dataframe with 26 rows and 4 columns.

- [.1] Subject factor anonymous subjects
- [.2] Group factor sport or control
- [.3] PSSbefore numeric stress measurement before training
- [.4] PSSafter numeric stress measurement after training

**Source**

Private communication. Charlotte Verdot, CRIS, Lyon 1 University, FRANCE

**References**


**Examples**

data(PrisonStress)

# The two groups are not randomized!  
# The control group is less stressed before the experiment  
with(PrisonStress,boxplot(PSSbefore~Group,ylab="Stress at the beginning of the study"))

# But more stressed at the end!  
with(PrisonStress,boxplot(PSSafter~Group,ylab="22 weeks later"))

# So the effects of physical training seems promising  
with(PrisonStress,plot(paired(PSSbefore,PSSafter),groups=Group,type="BA",facet=FALSE))

# Testing using gain scores analysis  
difference<-PrisonStress$PSSafter-PrisonStress$PSSbefore  
t.test(difference~PrisonStress$Group,var.equal=TRUE)
# Testing using ANCOVA
lmJail<-lm(PSSafter~PSSbefore+Group,data=PrisonStress)
anova(lmJail)

# Testing using repeated measures ANOVA
PSS<-c(PrisonStress$PSSbefore,PrisonStress$PSSafter)
Time<-factor(rep(c("Before","After"),c(26,26)))
Subject<-rep(PrisonStress$Subject,2)
Condition<-rep(PrisonStress$Group,2)
aovJail<-aov(PSS~Condition*Time+Error(Subject))
summary(aovJail)

---

**rpaired.contaminated**  
*Simulate paired samples*

**Description**

Simulate paired data with a given correlation (Kendall’s tau=(2/pi)arcsine(r)) and marginals being contaminated normal distributions: (1-eps)*F(x)+eps*F(x/K) where F is the cumulative standard normal distribution, eps the percentage of contamination and K a scale parameter. Moreover, this marginal can be multiplied by another scale parameter sigma but usually sigma=1.

**Usage**

```
rpaired.contaminated(n, d1 = c(0.1, 10, 1), d2 = c(0.1, 10, 1), r = 0.5)
```

**Arguments**

- `n`  
  sample size.
- `d1`  
  vector of 3 parameters for the first contaminated normal distribution (eps,K,sigma).
- `d2`  
  vector of 3 parameters for the second contaminated normal distribution.
- `r`  
  correlation.

**Value**

An object of class paired.

**Author(s)**

Stephane CHAMPELY

**References**

Simulate paired data with a given correlation (Kendall’s tau=(2/pi)arcsine(r)) and marginals being Generalized Tukey-Lambda (G-TL) distributions.

Usage

```
rpaired.gld(n, d1=c(0.000,0.1974,0.1349,0.1349), d2=c(0.000,0.1974,0.1349,0.1349), r)
```

Arguments

- `n`: sample size.
- `d1`: vector of four parameters for the first G-TL distribution.
- `d2`: vector of four parameters for the second G-TL distribution.
- `r`: correlation.

Value

An object of class paired.

Author(s)

Stephane CHAMPELY

References


See Also

`rpaired.contaminated`
**Examples**

```r
data(lambda.table)
p<e-rpaired.gld(n=30,d1=lambda.table[7,],d2=lambda.table[7,],r=0.5)
plot(p)
```

**Description**

This dataset gives the ratings on a continuous ten-points scale of two experts about 93 actions during several rugby union matches.

**Usage**

```r
data(Rugby)
```

**Format**

A dataframe with 93 rows and 3 columns.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>.1</td>
<td>EXPERT.1</td>
<td>numeric</td>
</tr>
<tr>
<td>.2</td>
<td>EXPERT.2</td>
<td>numeric</td>
</tr>
<tr>
<td>.3</td>
<td>Actions</td>
<td>factor</td>
</tr>
</tbody>
</table>

**Source**

Private communication. Mickael Campo, CRIS, Lyon 1 University, FRANCE.

**Examples**

```r
data(Rugby)
with(Rugby,plot(paired(EXPERT.1,EXPERT.2)))
with(Rugby,plot(paired(EXPERT.1,EXPERT.2),type="BA"))
```

**Description**

Robust test of scale for paired samples based on the absolute deviations from the trimmed means (or medians).
Usage

sandvikolsson.var.test(x, ...)

## Default S3 method:
sandvikolsson.var.test(x, y = NULL,
    alternative = c("two.sided", "less", "greater"),
    mu = 0, exact = NULL, correct = TRUE,
    conf.int = FALSE, conf.level = 0.95, location=c("trim","median"),tr=0.1, ...)

## S3 method for class 'paired'
sandvikolsson.var.test(x, ...)

Arguments

x  
first sample or object of class paired.

y  
second sample.

alternative  
alternative hypothesis.

mu  
the location parameter mu.

exact  
a logical indicating whether an exact p-value should be computed.

correct  
a logical indicating whether to apply continuity correction in the normal approx-
imation for the p-value.

conf.int  
a logical indicating whether a confidence interval should be computed.

conf.level  
confidence level.

location  
location parameter for centering: trimmed mean or median.

tr  
percentage of trimming.

...  
further arguments to be passed to or from methods.

Details

The data are transformed as deviations from the trimmed mean: \( X = \text{abs}(x - \text{mean}(x, \text{tr}=0.1)) \) and \( Y = (y - \text{mean}(y, \text{tr}=0.1)) \). A wilcoxon signed-rank test is then carried out on \( X \) and \( Y \).

Value

A list with class "htest" containing the components of a wilcoxon signed-rank test.

Author(s)

Stephane CHAMPELY

References

See Also

var.test, grambsch.var.test

Examples

z <- rnorm(20)
x <- rnorm(20) + z
y <- (rnorm(20) + z) * 2
sandvikolsson.var.test(x, y)

p <- paired(x, y)
sandvikolsson.var.test(p)

# some variations
sandvikolsson.var.test(p, tr = 0.2)
sandvikolsson.var.test(p, location = "median")

Sewage

Chlorinating sewage data from Preece (1982, Table 9)

Description

This dataset presents 8 paired data corresponding to log coliform densities per ml for 2 sewage chlorination methods on each of 8 days.

Usage

data(Sewage)

Format

A dataframe with 8 rows and 3 columns:

<table>
<thead>
<tr>
<th>.1</th>
<th>Day</th>
<th>numeric</th>
</tr>
</thead>
<tbody>
<tr>
<td>.2</td>
<td>Method_A</td>
<td>numeric</td>
</tr>
<tr>
<td>.3</td>
<td>Method_B</td>
<td>numeric</td>
</tr>
</tbody>
</table>

Source


References

Examples

data(Sewage)

# Visualising
with(Sewage, plot(paired(Method_A, Method_B), type="profile"))

# Basic paired t-test
with(Sewage, t.test(paired(Method_A, Method_B)))

# Influence of the 0.1 rounding on the t-test
with(Sewage, t.test(Method_A-Method_B-0.1))
with(Sewage, t.test(Method_A-Method_B+0.1))

Shoulde

Shoulder flexibility in swimmers

Description

This dataset gives the flexibility for the right and left shoulders in 15 swimmers and 15 sedentary people.

Usage

data(Shoulder)

Format

A dataframe with 30 rows and 4 columns.

<table>
<thead>
<tr>
<th>.1</th>
<th>Subject factor</th>
<th>anonymous subjects</th>
</tr>
</thead>
<tbody>
<tr>
<td>.2</td>
<td>Group factor</td>
<td>swimmer or control</td>
</tr>
<tr>
<td>.3</td>
<td>Right numeric</td>
<td>right shoulder flexibility (deg.)</td>
</tr>
<tr>
<td>.4</td>
<td>Left numeric</td>
<td>left shoulder flexibility (deg.)</td>
</tr>
</tbody>
</table>

Source

Private communication. Karine Monteil, CRIS, Lyon 1 University, FRANCE.

References

Examples

```r
data(Shoulder)

# Is there some heteroscedasticity?
with(Shoulder, plot(paired(Left, Right)))

# Swimmers are indeed quite different
with(Shoulder, plot(paired(Right, Left), groups=Group))

# A first derived variable to compare the amplitude in flexibility
with(Shoulder, boxplot((Left+Right)/2~Group, ylab="mean shoulder flexibility"))

# A second derived variable to study shoulder asymmetry
with(Shoulder, boxplot((abs(Left-Right))~Group, ylab="asymmetry in shoulder flexibility"))
```

Description

This dataset gives the actual and motor imaginary performances (time) in ski for 12 experts.

Usage

```r
data(SkiExperts)
```

Format

A dataframe with 12 rows and 3 columns.

| .[1] | Subject factor anonymous subjects |
| .[2] | Actual numeric actual performance (sec.) |
| .[3] | Imaginary numeric imaginary performance (sec.) |

Source

Private communication. Aymeric Guillot, CRIS, Lyon 1 University, FRANCE.

References


Examples

```r
data(SkiExperts)
```
# Visualising
with(SkiExperts,plot(paired(Actual,Imaginary),type="profile"))

# No underestimation of imaginary time for experts
with(SkiExperts,t.test(paired(Actual,Imaginary)))

# But a very interesting increase in dispersion in their
# predicted times
with(Ski Experts, var.test(paired(Actual, Imaginary)))

---

Sleep

Sleep hours data from Preece (1982, Table 16)

Description

This dataset presents paired data corresponding to the sleep hours gained by 10 patients (these are differences indeed) using two isomers (Dextro- and Laevo-). These data from Student were studied by Fischer (1925). Read the paper of Preece (1982, section 9) for a complete understanding of this quite complex situation.

Usage

data(Sleep)

Format

A dataframe with 10 rows and 2 columns:

<table>
<thead>
<tr>
<th></th>
<th>Dextro</th>
<th>numeric</th>
<th>sleep hour gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[2]</td>
<td>Laevo</td>
<td>numeric</td>
<td>sleep hour gain</td>
</tr>
</tbody>
</table>

Source


References


---

slidingchart

Sliding square plot
Description

Draw a sliding square plot for paired data which mixes the usual scatterplot with the tukey mean-difference plot.

Usage

```r
## S4 method for signature 'paired'
slidingchart(object, ...)
```

Arguments

- `object`: an object of class paired.
- `...`: arguments to be passed to methods.

Author(s)

Stephane CHAMPELY

References


See Also

plot

Examples

```r
data(PrisonStress)
with(PrisonStress, slidingchart(paired(PSSbefore, PSSafter)))
```

Summary

Summary statistics for paired samples

Description

Classical and robust statistics (location, scale and correlation) for paired samples.

Usage

```r
## S4 method for signature 'paired'
summary(object, tr=0.2)
```
summary

Arguments

object      an object of class paired.
tr          percentage of trimming.

Value

A list with a first table corresponding to location and scale statistics and a second table to Pearson and winsorized correlation.

The first table contains four rows corresponding to calculations for x, y, x-y and (x+y)/2 variables. The location and scale statistics are given in columns.

n          sample size.
mean       mean.
median     median.
trim       trimmed mean (tr=0.2)
sd         standard deviation.
IQR        interquartile range (standardised to be consistent with the sd in the normal case)
median ad  median of absolute deviations (standardised)
mean ad     mean of absolute deviations (standardised)
sdHwI      winsorised standard deviation (tr=0.2 and standardised)
min         minimum value.
max         maximum value.

Author(s)

Stephane CHAMPELY

Examples

z<-rnorm(20)
x<-rnorm(20)+z
y<-rnorm(20)+z+1
p<-paired(x,y)
summary(p)
Description
A method designed for objects of class paired.

Usage

```r
## S3 method for class 'paired'
t.test(x, ...)
```

Arguments

- `x` An object of class paired.
- `...` further arguments to be passed to or from methods.

Value
A list with class "htest" containing the following components:

- `statistic` the value of the t-statistic.
- `parameter` the degrees of freedom for the t-statistic.
- `p.value` the p-value for the test.
- `conf.int` a confidence interval for the mean appropriate to the specified alternative hypothesis.
- `estimate` the estimated difference in mean.
- `null.value` the specified hypothesized value of mean difference.
- `alternative` a character string describing the alternative hypothesis.
- `method` a character string indicating what type of test was performed (always paired here)
- `data.name` a character string giving the name(s) of the data.

Author(s)
Stephane Champely

See Also
yuen.t.test

Examples

```r
data(PrisonStress)
with(PrisonStress, t.test(paired(PSSbefore, PSSafter)))
```
**Description**

This dataset presents 8 paired data corresponding to numbers of lesions caused by two virus preparations inoculated into the two halves of each tobacco leaves.

**Usage**

```r
data(Tobacco)
```

**Format**

A dataframe with 8 rows and 3 columns:

<table>
<thead>
<tr>
<th></th>
<th>Plant factor</th>
<th>Preparation_1</th>
<th>Preparation_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1]</td>
<td></td>
<td>numeric</td>
<td>numeric</td>
</tr>
<tr>
<td>[2]</td>
<td></td>
<td>number of lesions</td>
<td>number of lesions</td>
</tr>
</tbody>
</table>

**Source**


**References**


**Examples**

```r
data(Tobacco)

# A clear outlier
with(Tobacco, plot(paired(Preparation_1,Preparation_2)))

# Comparison of normal and robust tests
with(Tobacco, t.test(paired(Preparation_1,Preparation_2)))
with(Tobacco, yuen.t.test(paired(Preparation_1,Preparation_2)))

with(Tobacco, var.test(paired(Preparation_1,Preparation_2)))
with(Tobacco, gumbscss.var.test(paired(Preparation_1,Preparation_2)))
```
with(Tobacco.cor.test(Preparation_1,Preparation_2))
with(Tobacco.winsor.cor.test(Preparation_1,Preparation_2))

# Maybe a transformation
require(MASS)
with(Tobacco.eqscplot(log(Preparation_1),log(Preparation_2)))
abline(0,1,col="red")

---

**var.test**

Tests of variance(s) for normal distribution(s)

---

**Description**

Classical tests of variance for one-sample, two-independent samples or paired samples.

**Usage**

```r
## Default S3 method:
var.test(x, y = NULL, ratio = 1, alternative = c("two.sided", "less", "greater"), paired = FALSE, conf.level = 0.95, ...)

## S3 method for class 'paired'
var.test(x, ...)

## Default S3 method:
pitman.morgan.test(x, y = NULL, alternative = c("two.sided", "less", "greater"), ratio = 1, conf.level = 0.95, ...)
```

**Arguments**

- `x` first sample or an object of class paired or an object of class lm.
- `y` second sample or an object of class lm.
- `ratio` a priori ratio of variances (two-samples) or variance (one-sample).
- `alternative` alternative hypothesis.
- `paired` independent (the default) or paired samples.
- `conf.level` confidence level.
- `...` further arguments to be passed to or from methods.

**Value**

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

- `statistic` the value of the X-squared statistic (one-sample) or F-statistic (two-samples).
- `parameter` the degrees of freedom for the statistic.
- `p.value` the p-value for the test.


**conf.int**

a confidence interval for the parameter appropriate to the specified alternative hypothesis.

**estimate**

the estimated variance(s).

**null.value**

the specified hypothesized value of the parameter.

**alternative**

a character string describing the alternative hypothesis.

**method**

a character string indicating what type of test was performed.

**data.name**

a character string giving the name(s) of the data.

**Author(s)**

Stephane CHAMPELY

**References**


**See Also**

bonette.seier.var.test, grambsch.var.test

**Examples**

data(HorseBeginners)

# one sample test
var.test(HorseBeginners$Actual, ratio=15)

# two independent samples test
var.test(HorseBeginners$Actual, HorseBeginners$Imaginary)

# two dependent samples test
var.test(HorseBeginners$Actual, HorseBeginners$Imaginary, paired=TRUE)
p <- with(HorseBeginners, paired(Actual, Imaginary))
var.test(p)

```
 wilcox.test

 Wilcoxon's signed rank test for paired data

Description

A method designed for objects of class paired.

Usage

## S3 method for class 'paired'
wilcox.test(x, ...)
```
Arguments

x  An object of class paired.

...  further arguments to be passed to or from methods.

Value

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

statistic  the value of V statistic.

parameter  the parameter(s) for the exact distribution of the test statistic.

p.value  the p-value for the test.

null.value  the true location shift mu.

alternative  a character string describing the alternative hypothesis.

method  a character string indicating what type of test was performed (always paired here)

data.name  a character string giving the name(s) of the data.

conf.int  a confidence interval for the location parameter. (Only present if argument conf.int = TRUE.)

estimate  an estimate of the location parameter. (Only present if argument conf.int = TRUE.)

Author(s)

Stephane Champely

See Also

yuen.test

Examples

data(PrisonStress)
with(PrisonStress, wilcox.test(PSSbefore, PSSafter))
with(PrisonStress, wilcox.test(PSSbefore, PSSafter, paired=TRUE))
with(PrisonStress, wilcox.test(paired(PSSbefore, PSSafter)))
Description
Test for association between paired samples, using winsorized correlation coefficient.

Usage
winsor.cor.test(x, ...)  

## Default S3 method:  
winsor.cor.test(x, y, tr=0.2,alternative = c("two.sided", "less", "greater"), ...)  

## S3 method for class 'paired'  
winsor.cor.test(x,x, tr=0.2,alternative = c("two.sided", "less", "greater"), ...)  

Arguments
 x an object of class paired or the first variable.  
y second variable.  
tr percentage of winsorizing.  
alternative a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less". You can specify just the initial letter.  
... further arguments to be passed to or from methods.  

Value
A list with class "htest" containing the following components:  
statistic the value of the t-statistic.  
parameter the degrees of freedom for the t-statistic.  
p.value the p-value for the test.  
estimate the winsorized correlation.  
null.value the specified hypothesized value of the winsorized correlation (=0).  
alternative a character string describing the alternative hypothesis.  
data.name a character string giving the name(s) of the data.  

Author(s)
Stephane Champely

See Also
cor.test
Examples

```r
data(PrisonStress)
with(PrisonStress, winsor.cor.test(PSSbefore, PSSafter))
with(PrisonStress, winsor.cor.test(paired(PSSbefore, PSSafter)))
```

---

### yuen.t.test

**Yuen's trimmed mean test**

#### Description

Yuen's test for one, two or paired samples.

#### Usage

```r
yuen.t.test(x, ...)
```

- **x**: first sample or object of class paired.
- **y**: second sample.
- **tr**: percentage of trimming.
- **alternative**: alternative hypothesis.
- **mu**: a number indicating the true value of the trimmed mean (or difference in trimmed means if you are performing a two sample test).
- **paired**: a logical indicating whether you want a paired yuen's test.
- **conf.level**: confidence level.
- **formula**: a formula of the form `y ~ f` where `y` is a numeric variable giving the data values and `f` a factor with TWO levels giving the corresponding groups.
- **data**: an optional matrix or data frame (or similar: see `model.frame`) containing the variables in the formula formula. By default the variables are taken from environment(formula).
- **subset**: an optional vector specifying a subset of observations to be used.
- **na.action**: a function which indicates what should happen when the data contain NAs. Defaults to `getOption("na.action")`.
- **...**: further arguments to be passed to or from methods.
Value

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

- **statistic**: the value of the t-statistic.
- **parameter**: the degrees of freedom for the t-statistic.
- **p.value**: the p-value for the test.
- **conf.int**: a confidence interval for the trimmed mean appropriate to the specified alternative hypothesis.
- **estimate**: the estimated trimmed mean or difference in trimmed means depending on whether it was a one-sample test or a two-sample test.
- **null.value**: the specified hypothesized value of the trimmed mean or trimmed mean difference depending on whether it was a one-sample test or a two-sample test.
- **alternative**: a character string describing the alternative hypothesis.
- **method**: a character string indicating what type of test was performed.
- **data.name**: a character string giving the name(s) of the data.

Author(s)

Stephane CHAMPELY, but some part are mere copy of the code of Wilcox (WRS)

References


See Also

t.test

Examples

```r
z <- rnorm(20)
x <- rnorm(20) + z
y <- rnorm(20) + z + 1

# two-sample test
yuen.t.test(x, y)

# one-sample test
yuen.t.test(y, mu = 1, tr = 0.25)

# paired-sample tests
yuen.t.test(x, y, paired = TRUE)

p <- paired(x, y)
yuen.t.test(p)
```
Index

*Topic classes
  paired-class, 29

*Topic datasets
  Anorexia, 3
  anscombe2, 4
  Barley, 5
  Blink, 6
  Blink2, 7
  BloodLead, 8
  ChickWeight, 10
  Corn, 11
  Datalcoholic, 12
  GDO, 14
  Grain, 15
  Grain2, 16
  GrapeFruit, 18
  HorseBeginners, 19
  IceSkating, 20
  Iron, 22
  lambda.table, 23
  Meat, 27
  PrisonStress, 35
  Rugby, 38
  Sewage, 40
  Shoulder, 41
  SkiExperts, 42
  Sleep, 43
  Tobacco, 47

*Topic htest
  bonettseier.var.test, 9
  grambsch.var.test, 17
  imam.var.test, 21
  levene.var.test, 24
  mcculloch.var.test, 25
  sandvikolsson.var.test, 38
  var.test, 48
  yuen.t.test, 52

*Topic methods
  plot, 34
  summary, 44

*Topic package
  PairedData-package, 2

  Anorexia, 3
  anscombe2, 4
  Barley, 5
  Blink, 6
  Blink2, 7
  BloodLead, 8
  bonettseier.var.test, 9
  ChickWeight, 10
  Corn, 11
  data.frame, 29
  Datalcoholic, 12
  effect.size, 13
  effect.size, paired-method (effect.size), 13
  GDO, 14
  Grain, 15
  Grain2, 16
  grambsch.var.test, 17
  GrapeFruit, 18
  HorseBeginners, 19
  IceSkating, 20
  imam.var.test, 21
  Iron, 22
  lambda.table, 23
  levene.var.test, 24
  list, 29
  mcculloch.var.test, 25
  Meat, 27
oldClass, 29

paired, 28
paired-class, 29
paired.plotBA, 30
paired.plotCor, 31
paired.plotMcNeil, 32
paired.plotProfiles, 33
PairedData (PairedData-package), 2
PairedData-package, 2
pitman.morgan.test.default (var.test), 48
plot, 34
plot.paired-method (plot), 34
plot.paired (plot), 34
PrisonStress, 35

rpaired.contaminated, 36
rpaired.gld, 37
Rugby, 38
sandvikolsson.var.test, 38
Sewage, 40
Shoulder, 41
SkiExperts, 42
Sleep, 43
slidingchart, 43
slidingchart.paired-method (slidingchart), 43
summary, 44
summary.paired-method (summary), 44
summary.paired (summary), 44

t.test, 46
Tobacco, 47

var.test, 48
vector, 29

wilcox.test, 49
winsor.cor.test, 51

yuen.t.test, 52