

Package ‘twoway’

June 26, 2020

Type Package

Title Analysis of Two-Way Tables

Version 0.6.3

Maintainer Michael Friendly <friendly@yorku.ca>

Description Carries out analyses of two-way tables with one observation per cell, together with graphical displays for an additive fit and a diagnostic plot for removable 'non-additivity' via a power transformation of the response. It implements Tukey's Exploratory Data Analysis (1973) <ISBN: 978-0201076165> methods, including a 1-degree-of-freedom test for row*column 'non-additivity', linear in the row and column effects.

URL <https://github.com/friendly/twoway>

BugReports <https://github.com/friendly/twoway/issues>

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

Language en-US

Suggests knitr, rmarkdown

NeedsCompilation no

Author Michael Friendly [aut, cre] (<<https://orcid.org/0000-0002-3237-0941>>),
Richard M. Heiberger [aut],
John Fox [ctb]

Repository CRAN

Date/Publication 2020-06-26 06:10:02 UTC

R topics documented:

anova.twoway	2
Arizona	3

as.data.frame.twoway	3
as.twoway	4
drugs	5
hstart	5
insectCounts	6
ladder_power	7
meanfit	8
medianfit	8
plot.twoway	9
print.twoway	11
residuals.twoway	12
sentRT	13
taskRT	13
to_long	14
twoway	15
twoway.formula	16

Index	18
--------------	-----------

anova.twoway	<i>ANOVA summary for a two-way table, including Tukey Additivity Test</i>
--------------	---

Description

Test for a 1-df interaction in two-way ANOVA table by the Tukey test.

Usage

```
## S3 method for class 'twoway'
anova(object, ...)
```

Arguments

object	a class("twoway") object
...	other arguments passed down, but not used here

Details

At present, this function simply gives the results of the ANOVAs for the additive model, the model including the 1 df term for non-additivity, and an anova() comparison of the two. The analysis is based on row and column means.

Author(s)

Michael Friendly

Examples

```
data(sentRT)
sent.2way <- twoway(sentRT)
anova(sent.2way)
```

Arizona	<i>Mean monthly temperatures in Arizona</i>
---------	---

Description

This is the data set used by Tukey (1977) for the initial examples of twoway tables

Format

a matrix of 7 rows (Month) and 3 columns (City) where the value is mean monthly temperature in degrees F. The matrix has a responseName attribute, "Temperature"

References

Tukey, J. W. (1977). *Exploratory Data Analysis*, Reading MA: Addison-Wesley. Exhibit 1 of chapter 10, p. 333

Examples

```
data(Arizona)
(AR.2way <-twoway(Arizona, method="median"))

## plot(AR.2way)
```

as.data.frame.twoway	<i>Convert a twoway object to a data frame This function converts a "twoway" object to a data.frame</i>
----------------------	---

Description

The rows and columns of the data table are strung out in standard R order in a vector, joined with row and column labels. Additional columns are added, representing the calculated values used in the two-way display.

Usage

```
## S3 method for class 'twoway'
as.data.frame(x, ...)
```

Arguments

`x` a "twoway" object
`...` other arguments, presently ignored

Value

a data.frame with $r \times c$ rows corresponding to the input data table, and the following columns

row row labels
col column labels
data the data value in the cell
fit the fitted value,
roweff the row effect
coleff the column effect
nonadd the 1 df for non-additivity value

Examples

```
data(sentRT)
sent.2way <- twoway(sentRT)
as.data.frame(sent.2way)
```

<code>as.twoway</code>	<i>Create an initial twoway object representing the data before fitting</i>
------------------------	---

Description

Create an initial twoway object representing the data before fitting
 Method for matrix input

Usage

```
as.twoway(x, ...)
```

S3 method for class 'matrix'

```
as.twoway(x, ..., name = deparse(substitute(x)),
  responseName = name, varNames = names(dimnames(x)))
```

Arguments

`x` a numeric matrix or numeric data frame with rownames
`...` other arguments, unused here
`name` Name of the data matrix
`responseName` Name of the response variable
`varNames` Names of the row and column variables

Value

An object of class c("tway") with all effects(roweff, col, overall) set to zero, and method="Initial"

Author(s)

Richard M. Heiberger

Richard M. Heiberger

Examples

```
data(taskRT)
as.tway(taskRT)
```

drugs

Scores for 5 subjects after being given each of 4 drugs

Description

The original source is Winer (1971), p. 268. This was used as an example in Friendly (1991).

References

Friendly, M. (1991). *SAS System for Statistical Graphics* Cary, NC: SAS Institute, Output 7.28

Examples

```
data(drugs)
tway(drugs)
```

hstart

Number of U.S. housing starts by month for the years 1965 – 1973

Description

Number of U.S. housing starts by month for the years 1965 – 1973

Format

a 9 x 12 matrix, where the entries are the number of housing starts, in thousands

References

Becker, Chambers & Wilks (1988), *The New S Language*, Brooks Cole. Friendly, M. (1991). *SAS System for Statistical Graphics* Cary, NC: SAS Institute, p.380

Examples

```
hstart.2way <- twoway(hstart, method="mean")
plot(hstart.2way)
```

insectCounts	<i>Counts of an insect for the combinations of 4 treatments and 6 areas of a field</i>
--------------	--

Description

Counts of numbers of an insect, *Leptinotarsa decemlineata* (the Colorado potato beetle), each of which is the sum for two plots treated alike, for all combinations of 4 treatments and 6 areas of the field chosen to be relatively homogeneous.

Format

a 4 x 6 matrix, where the rows are treatments and the columns are areas of a field.

Details

These data are used in Tukey (1977) Exhibit 1 of Ch 11 and throughout the chapter as examples of median polish. Because the data are counts, either a sqrt or log transformation would be reasonable.

References

Tukey, J. W. (1977). *Exploratory Data Analysis*, Reading MA: Addison-Wesley. Exhibit 1 of chapter 111

Examples

```
insect.2way <- twoway(insectCounts, method="median")
print(insect.2way, digits=2)

plot(insect.2way)
plot(insect.2way, which="diagnose")

# try sqrt transformation
insect.sqrt <- twoway(sqrt(insectCounts), method="median")
print(insect.sqrt, digits=2)

plot(insect.sqrt)
plot(insect.sqrt, which="diagnose")
```

ladder_power	<i>Find the nearest ladder-of-powers representation of a power transformation</i>
--------------	---

Description

The input power value is rounded to the nearest integer or fractional powers, $\pm 1/3, 1/2$. The function is presently designed just for display purposes.

Usage

```
ladder_power(p)
```

Arguments

p	A numeric power, for use as a transformation of a response, y , of the form y^p , where $p=0$ is interpreted to mean $\log(y)$
---	--

Details

In use, the transformation via the ladder of powers usually attaches a minus sign to the transformation when the power < 0 , so that the order of the response values are preserved under the transformation. Thus, a result of power = -0.5 is interpreted to mean $-1/\sqrt{y}$.

Value

a named list of two elements: power, the ladder-of-power value, and name, the name for the transformation

References

Tukey, J. W. (1977). *Exploratory Data Analysis*, Reading MA: Addison-Wesley.

Examples

```
ladder_power(0.6)
ladder_power(-0.6)
```

meanfit	<i>Fit a two-way table using row and column means</i>
---------	---

Description

Fit a two-way table using row and column means

Usage

```
meanfit(x, ..., na.rm=FALSE)
```

Arguments

x	a numeric matrix or data frame
...	other arguments passed down
na.rm	logical. Should missing values be removed?

Value

An object of class c("twoway") with the following named components:

overall the fitted constant term.

roweff the fitted row effects.

coleff the fitted column effects.

residuals the residuals.

name the name of the dataset.

rownames the names for the rows

colnames the names for the columns

method "median"

medianfit	<i>Fit a two-way table using median polish</i>
-----------	--

Description

Fit a two-way table using median polish

Usage

```
medianfit(x, trace.iter = FALSE, ...)
```


Arguments

`x` a numeric matrix or data frame
`trace.iter` whether to give verbose output of iteration history in median polish.
`...` other arguments passed down

Value

An object of class `c("twoway", "medpolish")` with the following named components:

overall the fitted constant term.

roweff the fitted row effects.

coleff the fitted column effects.

residuals the residuals.

name the name of the dataset.

rownames the names for the rows

colnames the names for the columns

method "median"

plot.twoway

Plot methods for two-way tables

Description

Plots either the fitted values and residuals under additivity or a diagnostic plot for removable non-additivity by a power transformation

Usage

```
## S3 method for class 'twoway'
plot(x, which = c("fit", "diagnose"), ...,
     na.rm=any(is.na(x$residuals)))

## S3 method for class 'twoway.fit'
plot(x, main = paste0("Tukey two-way fit plot for ",
  x$name, " (method: ", x$method, ")"), xlab = expression(hat(mu) *
  " + Column Effect - Row Effect"), ylab = expression("Fit = " * hat(mu)
  * " + Column Effect + Row Effect"), rfactor = 1, rcolor = c("blue",
  "red"), lwd = 3, ylim = NULL, ...,
  na.rm=any(is.na(x$residuals)))

## S3 method for class 'twoway.diagnose'
plot(x, annotate = TRUE, jitter = FALSE,
     smooth = FALSE, pch = 16, ...)
```

Arguments

<code>x</code>	a <code>class("twoway")</code> object
<code>which</code>	one of "fit" or "diagnose"
<code>...</code>	other arguments, passed to <code>plot</code>
<code>na.rm</code>	logical. Should missing values be removed?
<code>main</code>	plot title
<code>xlab</code>	X axis label
<code>ylab</code>	Y axis label
<code>rfactor</code>	draw lines for $\text{abs}(\text{residuals}) > \text{rfactor} * \text{sqrt}(\text{MSPE})$
<code>rcolor</code>	a vector of length 2 giving the color of lines for positive and negative residuals
<code>lwd</code>	line width for residual lines in the fit plot
<code>ylim</code>	Y axis limits
<code>annotate</code>	A logical value; if TRUE, the slope and power are displayed in the diagnostic plot
<code>jitter</code>	A logical value; if TRUE, the comparison values in the plot are jittered to avoid overplotting
<code>smooth</code>	A logical value; if TRUE, a smoothed <code>loess</code> curve is added to the plot
<code>pch</code>	Plot character for point symbols in the diagnostic plot

Details

For the `which="fit"` plot, the basic result comes from a plot of the row effects against the column fitted values, which appears as a rectangular grid in these coordinates. Rotating this 45 degrees counterclockwise give a plot in which the vertical coordinate is the fitted value for the two-way table, and the horizontal coordinate is the column fit minus the row effect. The spacing of the grid lines for the rows and columns of the table show the relative magnitudes of the row/column means or medians.

For the `which="diagnose"` plot, the interaction residuals from an additive model, $y_{ij} = \mu + \alpha_i + \beta_j$, are plotted against the estimated components $\alpha_i \beta_j / \mu$. If this plot shows a substantially non-zero slope, b , this analysis suggests that a power transformation, $y \rightarrow y^{(1-b)}$ might reduce the apparent interaction effects.

For both plots, if you want to directly compare the result of `method="mean"` and `method="median"`, it is essential to set the same `xlim` and `ylim` axes in the call.

Value

The diagnostic plot invisibly returns a list with elements `c("slope", "power")`

Examples

```
data(taskRT)
tw <- twoway(taskRT)
tw
twmed <- twoway(taskRT, method="median")
twmed
```

```

plot(tw, xlim=c(2,7), ylim=c(2,7)) ## use the same xlim and ylim, for comparison
plot(twmed, xlim=c(2,7), ylim=c(2,7))

plot(tw, which="diagnose", xlim=c(-.19, .19), ylim=c(-.5, .55))
plot(twmed, which="diagnose", xlim=c(-.19, .19), ylim=c(-.5, .55))

data(insectCounts)
twi <- twoway(insectCounts)
twimed <- twoway(insectCounts, method="median")

plot(twi, xlim=c(-250, 700), ylim=c(-180, 900))
plot(twimed, xlim=c(-250, 700), ylim=c(-180, 900))

plot(twi, which="diagnose", xlim=c(-160, 170), ylim=c(-200, 400)) ## power = .1
plot(twimed, which="diagnose", xlim=c(-160, 170), ylim=c(-200, 400)) ## power = .3

```

```
print.twoway          Print method for two-way tables
```

Description

Print method for two-way tables

Usage

```
## S3 method for class 'twoway'
print(x, digits = getOption("digits"), border = 2,
      zapsmall = TRUE, ...)
```

Arguments

x	a numeric matrix
digits	number of digits to print
border	if 0, the components "twoway" object ("overall", "roweff", "coleff", "residuals") are printed separately; if 1, the row, column and overall effects are joined to the residuals in a single table. if 2, row, column, overall and residuals are joined, and decorated with horizontal and vertical rules
zapsmall	a logical value; if TRUE small residuals are printed as 0.
...	other arguments passed down

Author(s)

Michael Friendly, Richard Heiberger

Examples

```

data(taskRT)
task.2way <- twoway(taskRT)
print(task.2way)
print(task.2way, border=0)

data(sentRT)
sent.2way <- twoway(sentRT)
print(sent.2way)
print(sent.2way, border=1)

```

residuals.twoway	<i>Extract residuals from a twoway object</i>
------------------	---

Description

Extract residuals from a twoway object
 Extract fitted values from a twoway object

Usage

```

## S3 method for class 'twoway'
residuals(object, nonadd = FALSE, ...)

## S3 method for class 'twoway'
fitted(object, nonadd = FALSE, ...)

```

Arguments

object	A class="twoway" object
nonadd	If TRUE, the 1 degree of freedom term for non-additivity is subtracted from the additive residuals
...	other arguments (unused)

Value

A numeric matrix of residuals corresponding to the data supplied to twoway
 A numeric matrix of fitted values corresponding to the data supplied to twoway

Examples

```

data(taskRT)
task.2way <- twoway(taskRT)
residuals(task.2way)
residuals(task.2way, nonadd=TRUE)

```

```

sum(residuals(task.2way)^2)           # SSE for additive model
sum(residuals(task.2way, nonadd=TRUE)^2) # SSPE, non-additive model
data(taskRT)
task.2way <- twoway(taskRT)
fitted(task.2way)
fitted(task.2way, nonadd=TRUE)

```

sentRT	<i>Reaction times for T/F judgments</i>
--------	---

Description

A demonstration 3 x 3 two-way table composed of reaction times for three subjects making T/F judgments on three types of sentences

References

Friendly, M. (1991). *SAS System for Statistical Graphics* Cary, NC: SAS Institute, Table 7.2

Examples

```

data(sentRT)
twoway(sentRT)

```

taskRT	<i>Data on reaction times for various tasks and topics</i>
--------	--

Description

A demonstration 3 x 4 two-way table composed of reaction times for tasks varying in difficulty, with content on different topics.

Format

A matrix of 3 rows and 4 columns, where the rows are the task difficulty levels and the columns are the the topics. The cell values are average reaction times (in sec.). The matrix has a responseName attribute, "RT"

Examples

```

data(taskRT)
twoway(taskRT)
twoway(taskRT, method="median")

```

to_long	<i>Reshape a data.frame or matrix to a long data.frame</i>
---------	--

Description

Reshape a data.frame or matrix to a long data.frame

Reshape a data.frame or matrix to a wide data.frame

Usage

```
to_long(wide, rowname = NULL, colname = NULL,
        responseName = deparse(substitute(wide)), varNames = c("Row", "Col"))
```

```
to_wide(long, row = 1, col = 2, response = 3)
```

Arguments

wide	A data.frame or matrix in wide form
rowname	Name for the row variable
colname	Name for the column variable
responseName	Name for the response variable. If wide is a matrix with an attribute that begins with "response", that value is taken as the responseName. Otherwise, the name of the wide object is used.
varNames	Default names for the row and column variables if not passed as rowname or colname
long	A data.frame in long form
row	Column index or quoted name of the row variable
col	Column index or quoted name of the column variable
response	Column index or quoted name of the response variable

Value

A data.frame in long format

Author(s)

Michael Friendly and Richard M. Heiberger

Michael Friendly and Richard M. Heiberger

Examples

```
Arizona.long <- to_long(Arizona, varNames=c("Month", "City"))
Arizona.long
```

```
Arizona.long <- to_long(Arizona, varNames=c("Month", "City"))
# back the other way
to_wide(Arizona.long)
```

twoway

Analysis of a two-way table with one observation per cell

Description

Fits an additive model using either row and column means or Tukey's median polish procedure

Usage

```
twoway(x, ...)

## Default S3 method:
twoway(x, method = c("mean", "median"), ...,
       name = deparse(substitute(x)), responseName = attr(x, "response"),
       varNames = names(dimnames(x)))
```

Arguments

x	a numeric matrix or data frame.
...	other arguments passed down
method	one of "mean" or "median"
name	name for the input dataset
responseName	name for the response variable
varNames	names for the Row and Column variables

Details

The `rownames(x)` are used as the levels of the row factor and the `colnames(x)` are the levels of the column factor. For a numeric matrix, the function uses the `names(dimnames(x))` as the names of these variables, and, if present, a `responseName` attribute as the name for the response variable.

Value

An object of class `c("twoway")` with the following named components:

- overall** the fitted constant term.
- roweff** the fitted row effects.
- coleff** the fitted column effects.
- residuals** the residuals.
- name** the name of the dataset.
- rownames** the names for the rows
- colnames** the names for the columns
- method** the fitting method
- varNames** the names of the row and column variables

responseName the name of the response variable
compValue the comparison values, for the diagnostic plot
slope the slope value, for the diagnostic plot
power the suggested power transformation, 1-slope

An object of class "twoway", but supplemented by additional components used for labeling

Author(s)

Michael Friendly

References

Tukey, J. W. (1977). *Exploratory Data Analysis*, Reading MA: Addison-Wesley. Friendly, M. (1991). *SAS System for Statistical Graphics* Cary, NC: SAS Institute

See Also

[codetwoway.formula](#), [codemedpolish](#)
[medianfit](#), [meanfit](#)

Examples

```
data(taskRT)
twoway(taskRT)
```

<code>twoway.formula</code>	<i>Formula method for twoway analysis using a dataset in long format</i>
-----------------------------	--

Description

The formula method reshapes the data set from long to wide format and calls the default method.

Usage

```
## S3 method for class 'formula'
twoway(formula, data, subset, na.action, ...)
```

Arguments

<code>formula</code>	A formula of the form <code>response ~ rowvar + colvar</code> , where response is numeric
<code>data</code>	The name of the data set, containing a row vector, column factor and a numeric response
<code>subset</code>	An expression to subset the data (unused)
<code>na.action</code>	What to do with NAs? (unused)
<code>...</code>	other arguments, passed down

Author(s)

Michael Friendly and Richard Heiberger

References

the conversion of long to wide in a formula method was suggested on <https://stackoverflow.com/questions/50469320/how-to-write-a-formula-method-that-converts-long-to-wide>

Examples

```
longRT <- to_long(taskRT)
twoway(RT ~ Task + Topic, data=longRT)
```

Index

*Topic **data**

- Arizona, [3](#)
- drugs, [5](#)
- hstart, [5](#)
- insectCounts, [6](#)
- sentRT, [13](#)
- taskRT, [13](#)

[anova.twoway](#), [2](#)

Arizona, [3](#)

[as.data.frame.twoway](#), [3](#)

[as.twoway](#), [4](#)

[drugs](#), [5](#)

[fitted.twoway \(residuals.twoway\)](#), [12](#)

[hstart](#), [5](#)

[insectCounts](#), [6](#)

[ladder_power](#), [7](#)

[loess](#), [10](#)

[meanfit](#), [8](#), [16](#)

[medianfit](#), [8](#), [16](#)

[medpolish](#), [16](#)

[plot.twoway](#), [9](#)

[print.twoway](#), [11](#)

[residuals.twoway](#), [12](#)

[sentRT](#), [13](#)

[taskRT](#), [13](#)

[to_long](#), [14](#)

[to_wide \(to_long\)](#), [14](#)

[twoway](#), [15](#)

[twoway.formula](#), [16](#), [16](#)