# Package 'pairwiseComparisons'

June 23, 2020

**Type** Package

**Title** Multiple Pairwise Comparison Tests

**Version** 1.1.2

**Maintainer** Indrajeet Patil <patilindrajeet.science@gmail.com>

**Description** Multiple pairwise comparison tests on tidy data for
one-way analysis of variance for both between-subjects and
within-subjects designs. Currently, it supports only the most common
types of statistical analyses and tests: parametric (Welch's and
Student's t-test), nonparametric (Durbin-Conover and Dunn test),
robust (Yuen's trimmed means test), and Bayes Factor (Student's
t-test).

**License** GPL-3 | file LICENSE

**URL** <https://indrajeetpatil.github.io/pairwiseComparisons/>,
<https://github.com/IndrajeetPatil/pairwiseComparisons>

**BugReports** <https://github.com/IndrajeetPatil/pairwiseComparisons/issues>

**Depends** R (>= 3.6.0)

**Imports** broomExtra, dplyr, dunn.test, forcats, ipmisc (>= 3.0.1),
PMCMRplus, purrr, rlang, stats, tidyBF (>= 0.2.1), tidyr,
utils, WRS2 (>= 1.1-0)

**Suggests** knitr, rmarkdown, spelling, testthat

**Encoding** UTF-8

**Language** en-US

**LazyData** true

**RoxygenNote** 7.1.0.9000

**NeedsCompilation** no

**Author** Indrajeet Patil [cre, aut, cph]
(<https://orcid.org/0000-0003-1995-6531>)

**Repository** CRAN

**Date/Publication** 2020-06-23 10:50:02 UTC

# R topics documented:

---

bugs_long                    *Tidy version of the "Bugs" dataset.*

---

### Description

Tidy version of the "Bugs" dataset.

### Usage

```
bugs_long
```

### Format

A data frame with 372 rows and 6 variables

- subject. Dummy identity number for each participant.

- gender. Participant's gender (Female, Male).

- region. Region of the world the participant was from.

- education. Level of education.

- condition. Condition of the experiment the participant gave rating for (**LDLF**: low freighteningness and low disgustingness; **LFHD**: low freighteningness and high disgustingness; **HFHD**: high freighteningness and low disgustingness; **HFHD**: high freighteningness and high disgustingness).

- desire. The desire to kill an arthropod was indicated on a scale from 0 to 10.

### Details

This data set, "Bugs", provides the extent to which men and women want to kill arthropods that vary in freighteningness (low, high) and disgustingness (low, high). Each participant rates their attitudes towards all anthropods. Subset of the data reported by Ryan et al. (2013).

### Source

https://www.sciencedirect.com/science/article/pii/S0747563213000277

## Examples

```
dim(bugs_long)
head(bugs_long)
dplyr::glimpse(bugs_long)
```

---

iris_long                   *Edgar Anderson's Iris Data in long format.*

---

## Description

Edgar Anderson's Iris Data in long format.

## Usage

```
iris_long
```

## Format

A data frame with 600 rows and 5 variables

- id. Dummy identity number for each flower (150 flowers in total).

- Species. The species are *Iris setosa*, *versicolor*, and *virginica*.

- condition. Factor giving a detailed description of the attribute (Four levels: "Petal.Length", "Petal.Width", "Sepal.Length", "Sepal.Width").

- attribute. What attribute is being measured ("Sepal" or "Pepal").

- measure. What aspect of the attribute is being measured ("Length" or "Width").

- value. Value of the measurement.

## Details

This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica.

This is a modified dataset from datasets package.

## Examples

```
dim(iris_long)
head(iris_long)
dplyr::glimpse(iris_long)
```

---

| movies_long | *Movie information and user ratings from IMDB.com (long format).* |
|---|---|

---

#### Description

Movie information and user ratings from IMDB.com (long format).

#### Usage

```
movies_long
```

#### Format

A data frame with 1,579 rows and 8 variables

- title. Title of the movie.
- year. Year of release.
- budget. Total budget (if known) in US dollars
- length. Length in minutes.
- rating. Average IMDB user rating.
- votes. Number of IMDB users who rated this movie.
- mpaa. MPAA rating.
- genre. Different genres of movies (action, animation, comedy, drama, documentary, romance, short).

#### Details

Modified dataset from `ggplot2movies` package.

The internet movie database, <http://imdb.com/>, is a website devoted to collecting movie data supplied by studios and fans. It claims to be the biggest movie database on the web and is run by amazon. More about information imdb.com can be found online, [http://imdb.com/help/show_leaf?about](http://imdb.com/help/show_leaf?about), including information about the data collection process, [http://imdb.com/help/show_leaf?infosource](http://imdb.com/help/show_leaf?infosource).

Movies were are identical to those selected for inclusion in movies_wide but this dataset has been constructed such that every movie appears in one and only one genre category.

#### Source

<https://CRAN.R-project.org/package=ggplot2movies>

#### Examples

```
dim(movies_long)
head(movies_long)
dplyr::glimpse(movies_long)
```

| movies_wide | *Movie information and user ratings from IMDB.com (wide format).* |
|---|---|

#### Description

Movie information and user ratings from IMDB.com (wide format).

#### Usage

```
movies_wide
```

#### Format

A data frame with 1,579 rows and 13 variables

- title. Title of the movie.
- year. Year of release.
- budget. Total budget in millions of US dollars
- length. Length in minutes.
- rating. Average IMDB user rating.
- votes. Number of IMDB users who rated this movie.
- mpaa. MPAA rating.
- action, animation, comedy, drama, documentary, romance, short. Binary variables representing if movie was classified as belonging to that genre.
- NumGenre. The number of different genres a film was classified in an integer between one and four

#### Details

Modified dataset from `ggplot2movies` package.

The internet movie database, <http://imdb.com/>, is a website devoted to collecting movie data supplied by studios and fans. It claims to be the biggest movie database on the web and is run by amazon. More information about imdb.com can be found online, [http://imdb.com/help/show_leaf?about](http://imdb.com/help/show_leaf?about), including information about the data collection process, [http://imdb.com/help/show_leaf?infosource](http://imdb.com/help/show_leaf?infosource).

Movies were selected for inclusion if they had a known length and had been rated by at least one imdb user. Small categories such as documentaries and NC-17 movies were removed.

#### Source

<https://CRAN.R-project.org/package=ggplot2movies>

#### Examples

```
dim(movies_wide)
head(movies_wide)
dplyr::glimpse(movies_wide)
```

---

pairwise_caption                 *Expression containing details about pairwise comparison test*

---

### Description

This returns an expression containing details about the pairwise comparison test and the *p*-value adjustment method. These details are typically included in `ggstatsplot` package plots as a caption.

### Usage

```
pairwise_caption(caption, test.description, p.adjust.method)
```

### Arguments

caption          Additional text to be included in the plot.

test.description

                 Text describing the details of the test.

p.adjust.method

                 Adjustment method for *p*-values for multiple comparisons. Possible methods
                 are: ″holm″ (default), ″hochberg″, ″hommel″, ″bonferroni″, ″BH″, ″BY″,
                 ″fdr″, ″none″.

### Examples

```
library(pairwiseComparisons)
pairwise_caption("my caption", "Student's t-test", "holm")
```

---

pairwise_comparisons      *Multiple pairwise comparison tests with tidy data*

---

### Description

Calculate parametric, non-parametric, and robust pairwise comparisons between group levels with corrections for multiple testing.

### Usage

```
pairwise_comparisons(
  data,
  x,
  y,
  type = "parametric",
  paired = FALSE,
  var.equal = FALSE,
  tr = 0.1,
```

```
    bf.prior = 0.707,
    p.adjust.method = "holm",
    k = 2L,
    ...
)

pairwise_p(
    data,
    x,
    y,
    type = "parametric",
    paired = FALSE,
    var.equal = FALSE,
    tr = 0.1,
    bf.prior = 0.707,
    p.adjust.method = "holm",
    k = 2L,
    ...
)
```

## Arguments

| | |
|---|---|
| data | A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will **not** be accepted. |
| x | The grouping variable from the dataframe data. |
| y | The response (a.k.a. outcome or dependent) variable from the dataframe data. |
| type | Type of statistic expected ("parametric" or "nonparametric" or "robust" or "bayes").Corresponding abbreviations are also accepted: "p" (for parametric), "np" (nonparametric), "r" (robust), or "bf"resp. |
| paired | Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE. |
| var.equal | a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used. |
| tr | Trim level for the mean when carrying out robust tests. If you get error stating "Standard error cannot be computed because of Winsorized variance of 0 (e.g., due to ties). Try to decrease the trimming level.", try to play around with the value of tr, which is by default set to 0.1. Lowering the value might help. |
| bf.prior | A number between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes factors. |
| p.adjust.method | |
| | Adjustment method for *p*-values for multiple comparisons. Possible methods are: "holm" (default), "hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none". |
| k | Number of digits after decimal point (should be an integer) (Default: k = 2L). |
| ... | Current ignored. |

**Value**

A tibble dataframe containing two columns corresponding to group levels being compared with each other (group1 and group2) and p.value column corresponding to this comparison. The dataframe will also contain a p.value.label column containing a *label* for this *p*-value, in case this needs to be displayed in geom_ggsignif. In addition to these common columns across the different types of statistics, there will be additional columns specific to the type of test being run.

The significance column asterisks indicate significance levels of *p*-values in the American Psychological Association (APA) mandated format:

- ns : > 0.05
- * : < 0.05
- ** : < 0.01
- *** : < 0.001

**Examples**

```
# for reproducibility
set.seed(123)
library(pairwiseComparisons)

#------------------ between-subjects design ---------------------------

# parametric
# if `var.equal = TRUE`, then Student's t-test will be run
pairwise_comparisons(
  data = mtcars,
  x = cyl,
  y = wt,
  type = "parametric",
  var.equal = TRUE,
  paired = FALSE,
  p.adjust.method = "none"
)

# if `var.equal = FALSE`, then Games-Howell test will be run
pairwise_comparisons(
  data = mtcars,
  x = cyl,
  y = wt,
  type = "parametric",
  var.equal = FALSE,
  paired = FALSE,
  p.adjust.method = "bonferroni"
)

# non-parametric (Dunn test)
pairwise_comparisons(
  data = mtcars,
  x = cyl,
```

```
  y = wt,
  type = "nonparametric",
  paired = FALSE,
  p.adjust.method = "none"
)

# robust (Yuen's trimmed means t-test)
pairwise_comparisons(
  data = mtcars,
  x = cyl,
  y = wt,
  type = "robust",
  paired = FALSE,
  p.adjust.method = "fdr"
)

# Bayes Factor (Student's t-test)
pairwise_comparisons(
  data = mtcars,
  x = cyl,
  y = wt,
  type = "bayes",
  paired = FALSE
)

#------------------ within-subjects design ---------------------------

# parametric (Student's t-test)
pairwise_comparisons(
  data = bugs_long,
  x = condition,
  y = desire,
  type = "parametric",
  paired = TRUE,
  p.adjust.method = "BH"
)

# non-parametric (Durbin-Conover test)
pairwise_comparisons(
  data = bugs_long,
  x = condition,
  y = desire,
  type = "nonparametric",
  paired = TRUE,
  p.adjust.method = "BY"
)

# robust (Yuen's trimmed means t-test)
pairwise_comparisons(
  data = bugs_long,
  x = condition,
  y = desire,
  type = "robust",
```

```
  paired = TRUE,
  p.adjust.method = "hommel"
)

# Bayes Factor (Student's t-test)
pairwise_comparisons(
  data = bugs_long,
  x = condition,
  y = desire,
  type = "bayes",
  paired = TRUE
)
```

---

p_adjust_text                    *Preparing text to describe which* p-*value adjustment method was used*

---

### Description

Preparing text to describe which *p*-value adjustment method was used

### Usage

```
p_adjust_text(p.adjust.method)
```

### Arguments

p.adjust.method

Adjustment method for *p*-values for multiple comparisons. Possible methods
are: "holm" (default), "hochberg", "hommel", "bonferroni", "BH", "BY",
"fdr", "none".

### Value

Standardized text description for what method was used.

### Examples

```
library(pairwiseComparisons)
p_adjust_text("none")
p_adjust_text("BY")
```

# Index