

# Package ‘statsExpressions’

August 28, 2019

**Type** Package

**Title** Expressions with Statistical Details

**Version** 0.1.1

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

**Description** Statistical processing backend for 'ggstatsplot',  
this package creates expressions with details from statistical tests.  
Currently, it supports only the most common types of statistical  
tests: parametric, nonparametric, robust, and bayesian versions of  
t-test/anova, correlation analyses, contingency table analysis.

**License** GPL-3 | file LICENSE

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

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

**Depends** R (>= 3.5.0)

**Imports** BayesFactor (>= 0.9.12-4.2), boot (>= 1.3-22), broomExtra (>= 0.0.5), crayon (>= 1.3.4), dplyr (>= 0.8.3), ellipsis (>= 0.2.0.1), ez (>= 4.4-0), groupedstats, magrittr (>= 1.5), MCMCpack (>= 1.4-4), methods, psych, rcompanion (>= 2.3.0), rlang (>= 0.4.0), stats, tibble (>= 2.1.3), tidyr (>= 0.8.3), WRS2 (>= 1.0-0)

**Suggests** ggplot2, knitr, rmarkdown, spelling, testthat

**VignetteBuilder** knitr

**Encoding** UTF-8

**Language** en-US

**LazyData** true

**RoxygenNote** 6.1.1.9000

**NeedsCompilation** no

**Author** Indrajeet Patil [cre, aut, ctb]  
(<<https://orcid.org/0000-0003-1995-6531>>),  
Chuck Powell [ctb] (<<https://orcid.org/0000-0002-3606-2188>>)

Repository CRAN

Date/Publication 2019-08-28 08:20:03 UTC

## R topics documented:

bf_contingency_tab . . . . .	2
bf_corr_test . . . . .	5
bf_expr . . . . .	6
bf_extractor . . . . .	7
bf_oneway_anova . . . . .	8
bf_ttest . . . . .	9
bugs_long . . . . .	12
expr_anova_bayes . . . . .	13
expr_anova_nonparametric . . . . .	14
expr_anova_parametric . . . . .	16
expr_anova_robust . . . . .	18
expr_contingency_tab . . . . .	20
expr_corr_test . . . . .	22
expr_template . . . . .	24
expr_t_bayes . . . . .	26
expr_t_nonparametric . . . . .	27
expr_t_onesample . . . . .	29
expr_t_parametric . . . . .	32
expr_t_robust . . . . .	34
intent_morality . . . . .	35
iris_long . . . . .	37
long_to_wide_converter . . . . .	38
movies_long . . . . .	38
movies_wide . . . . .	39
Titanic_full . . . . .	40
VR_dilemma . . . . .	41
<b>Index</b>	<b>43</b>

---

bf_contingency_tab	<i>Bayesian contingency table analysis.</i>
--------------------	---

---

## Description

Bayesian contingency table analysis.

**Usage**

```
bf_contingency_tab(data, x, y = NULL, counts = NULL, ratio = NULL,
  sampling.plan = "indepMulti", fixed.margin = "rows",
  prior.concentration = 1, caption = NULL, output = "null", k = 2,
  ...)
```

```
bf_onesample_proptest(data, x, y = NULL, counts = NULL, ratio = NULL,
  sampling.plan = "indepMulti", fixed.margin = "rows",
  prior.concentration = 1, caption = NULL, output = "null", k = 2,
  ...)
```

**Arguments**

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	an m by n matrix of counts (integers m,n > 1)
y	The variable to use as the <b>columns</b> in the contingency table. Default is NULL. If NULL, one-sample proportion test (a goodness of fit test) will be run for the main variable. Otherwise an appropriate association test will be run.
counts	A string naming a variable in data containing counts, or NULL if each row represents a single observation (Default).
ratio	A vector of proportions: the expected proportions for the proportion test (should sum to 1). Default is NULL, which means the null is equal theoretical proportions across the levels of the nominal variable. This means if there are two levels this will be ratio = c(0.5, 0.5) or if there are four levels this will be ratio = c(0.25, 0.25, 0.25, 0.25), etc.
sampling.plan	Character describing the sampling plan. Possible options are "indepMulti" (independent multinomial; default), "poisson", "jointMulti" (joint multinomial), "hypergeom" (hypergeometric). For more, see ?BayesFactor::contingencyTableBF().
fixed.margin	For the independent multinomial sampling plan, which margin is fixed ("rows" or "cols"). Defaults to "rows".
prior.concentration	Specifies the prior concentration parameter, set to 1 by default. It indexes the expected deviation from the null hypothesis under the alternative, and corresponds to Gunel and Dickey's (1974) "a" parameter.
caption	Text to display as caption (will be displayed on top of the Bayes Factor caption/message).
output	Can either be "null" (or "caption" or "H0" or "h0"), which will return expression with evidence in favor of the null hypothesis, or "alternative" (or "title" or "H1" or "h1"), which will return expression with evidence in favor of the alternative hypothesis, or "results", which will return a dataframe with results all the details).
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
...	further arguments to be passed to or from methods.

**Note**

Bayes Factor for goodness of fit test is based on gist provided by Richard Morey: <https://gist.github.com/richarddmorey/a4cd3a2051f373db917550d67131dba4>.

**Author(s)**

Indrajeet Patil

**See Also**

[bf\\_corr\\_test](#), [bf\\_oneway\\_anova](#), [bf\\_ttest](#)

**Examples**

```
# ----- association tests -----  
  
# for reproducibility  
set.seed(123)  
library(statsExpressions)  
  
# to get caption (in favor of null)  
bf_contingency_tab(  
  data = mtcars,  
  x = am,  
  y = cyl,  
  fixed.margin = "cols"  
)  
  
# to get caption (in favor of alternative)  
bf_contingency_tab(  
  data = mtcars,  
  x = am,  
  y = cyl,  
  fixed.margin = "rows",  
  output = "alternative"  
)  
  
# to see results  
bf_contingency_tab(  
  data = mtcars,  
  x = am,  
  y = cyl,  
  sampling.plan = "jointMulti",  
  fixed.margin = "rows",  
  prior.concentration = 1  
)  
  
# ----- goodness of fit tests -----  
  
bf_contingency_tab(  

```

```

data = mtcars,
x = am,
prior.concentration = 10
)

```

bf\_corr\_test

*Bayesian correlation test.***Description**

Bayesian correlation test.

**Usage**

```
bf_corr_test(data, x, y, bf.prior = 0.707, caption = NULL,
output = "null", k = 2, ...)
```

**Arguments**

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The column in data containing the explanatory variable to be plotted on the x-axis. Can be entered either as a character string (e.g., "x") or as a bare expression (e.g, x).
y	The column in data containing the response (outcome) variable to be plotted on the y-axis. Can be entered either as a character string (e.g., "y") or as a bare expression (e.g, y).
bf.prior	A numeric value between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes Factors.
caption	Text to display as caption (will be displayed on top of the Bayes Factor caption/message).
output	Can either be "null" (or "caption" or "H0" or "h0"), which will return expression with evidence in favor of the null hypothesis, or "alternative" (or "title" or "H1" or "h1"), which will return expression with evidence in favor of the alternative hypothesis, or "results", which will return a dataframe with results all the details).
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
...	Additional arguments (currently ignored).

**Author(s)**

Indrajeet Patil

**See Also**

[bf\\_contingency\\_tab](#), [bf\\_oneway\\_anova](#), [bf\\_ttest](#)

## Examples

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

# to get caption (default)
bf_corr_test(
  data = anscombe,
  x = x1,
  y = y4,
  bf.prior = 1
)

# to see results
bf_corr_test(
  data = anscombe,
  x = x1,
  y = y4,
  bf.prior = 0.8,
  output = "results"
)
```

---

bf\_expr

*Prepare caption with expression for Bayes Factor results*


---

## Description

Convenience function to write a caption message with bayes factors in favor of the null hypothesis.

## Usage

```
bf_expr(bf.df, k = 2, output = "null", caption = NULL, ...)
```

## Arguments

bf.df	A dataframe containing two columns log_e_bf01 (for evidence in favor of null hypothesis) and bf.prior. If dataframe contains more than two rows, only the first row will be used.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
output	Can either be "null" (or "caption" or "H0" or "h0"), which will return expression with evidence in favor of the null hypothesis, or "alternative" (or "title" or "H1" or "h1"), which will return expression with evidence in favor of the alternative hypothesis, or "results", which will return a dataframe with results all the details).
caption	Text to display as caption (will be displayed on top of the Bayes Factor caption/message).
...	Additional arguments (ignored).

## Examples

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

# dataframe containing results
bf_results <-
  statsExpressions::bf_extractor(BayesFactor::correlationBF(
    x = iris$Sepal.Length,
    y = iris$Petal.Length
  )) %>%
  dplyr::mutate(.data = ., bf.prior = 0.707)

# creating caption (for null)
statsExpressions::bf_expr(
  bf.df = bf_results,
  output = "null",
  k = 3,
  caption = "Note: Iris dataset"
)

# creating caption (for alternative)
statsExpressions::bf_expr(
  bf.df = bf_results,
  output = "alternative"
)
```

---

bf\_extractor

*Extract Bayes Factors from BayesFactor model object.*

---

## Description

Extract Bayes Factors from BayesFactor model object.

## Usage

```
bf_extractor(bf.object, ...)
```

## Arguments

bf.object	An object from BayesFactor package.
...	Currently ignored.

## Author(s)

Indrajeet Patil

**Examples**

```
# getting only Bayes Factors
statsExpressions::bf_extractor(
  BayesFactor::anovaBF(
    formula = Sepal.Length ~ Species,
    data = iris,
    progress = FALSE
  )
)
```

---

bf_oneway_anova	<i>Bayesian one-way analysis of variance.</i>
-----------------	---

---

**Description**

Bayesian one-way analysis of variance.

**Usage**

```
bf_oneway_anova(data, x, y, bf.prior = 0.707, caption = NULL,
  output = "null", paired = FALSE, k = 2, ...)
```

**Arguments**

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
bf.prior	A numeric value between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes Factors.
caption	Text to display as caption (will be displayed on top of the Bayes Factor caption/message).
output	Can either be "null" (or "caption" or "H0" or "h0"), which will return expression with evidence in favor of the null hypothesis, or "alternative" (or "title" or "H1" or "h1"), which will return expression with evidence in favor of the alternative hypothesis, or "results", which will return a dataframe with results all the details).
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
...	Additional arguments.

**Author(s)**

Indrajeet Patil



**See Also**

[bf\\_contingency\\_tab](#), [bf\\_corr\\_test](#), [bf\\_ttest](#)

**Examples**

```
# between-subjects -----
# to get caption (default)
bf_oway_anova(
  data = iris,
  x = Species,
  y = Sepal.Length,
  bf.prior = 0.8,
  paired = FALSE
)

# to get results dataframe
bf_oway_anova(
  data = iris,
  x = Species,
  y = Sepal.Length,
  bf.prior = 0.8,
  output = "results"
)

# within-subjects -----
bf_oway_anova(
  data = bugs_long,
  x = condition,
  y = desire,
  paired = TRUE
)
```

---

bf\_ttest

*Bayes Factor for t-test*


---

**Description**

Bayes Factor for  $t$ -test

**Usage**

```
bf_ttest(data, x, y = NULL, test.value = 0, paired = FALSE,
  bf.prior = 0.707, caption = NULL, output = "null", k = 2, ...)
```

```
bf_one_sample_ttest(data, x, y = NULL, test.value = 0,
```

```
paired = FALSE, bf.prior = 0.707, caption = NULL,
output = "null", k = 2, ...)

bf_two_sample_ttest(data, x, y = NULL, test.value = 0,
paired = FALSE, bf.prior = 0.707, caption = NULL,
output = "null", k = 2, ...)
```

### Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	Either the grouping variable from the dataframe data if it's a two-sample <i>t</i> -test or a numeric variable if it's a one-sample <i>t</i> -test.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
test.value	A number specifying the value of the null hypothesis (Default: 0).
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
bf.prior	A numeric value between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes Factors.
caption	Text to display as caption (will be displayed on top of the Bayes Factor caption/message).
output	Can either be "null" (or "caption" or "H0" or "h0"), which will return expression with evidence in favor of the null hypothesis, or "alternative" (or "title" or "H1" or "h1"), which will return expression with evidence in favor of the alternative hypothesis, or "results", which will return a dataframe with results all the details).
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
...	Additional arguments (currently ignored).

### Details

If y is NULL, a one-sample *t*-test will be carried out, otherwise a two-sample *t*-test will be carried out.

### Author(s)

Indrajeet Patil

### See Also

[bf\\_contingency\\_tab](#), [bf\\_corr\\_test](#), [bf\\_oneway\\_anova](#)

**Examples**

```
# ----- two-samples tests -----

# for reproducibility
set.seed(123)
library(statsExpressions)

# to get caption (default)
bf_ttest(
  data = mtcars,
  x = am,
  y = wt,
  paired = FALSE,
  bf.prior = 0.880
)

# to see results
bf_ttest(
  data = mtcars,
  x = am,
  y = wt,
  paired = FALSE,
  output = "results"
)

# for paired sample test
bf_ttest(
  data = dplyr::filter(
    statsExpressions::intent_morality,
    condition %in% c("accidental", "attempted"),
    harm == "Poisoning"
  ),
  x = condition,
  y = rating,
  paired = TRUE,
  output = "results"
)

# ----- one-samples test -----

# to get caption (default)
bf_ttest(
  data = iris,
  x = Sepal.Length,
  test.value = 5.85,
  bf.prior = 0.8,
  output = "caption", k = 2
)

# to get results dataframe
bf_ttest(
```

```
data = iris,  
x = Sepal.Length,  
test.value = 5.85,  
bf.prior = 0.8,  
output = "results"  
)
```

---

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 frighteningness and low disgustingness; **LFHD**: low frighteningness and high disgustingness; **HFHD**: high frighteningness and low disgustingness; **HFHD**: high frighteningness 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 frighteningness (low, high) and disgustingness (low, high). Each participant rates their attitudes towards all arthropods. 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)
```

---

expr\_anova\_bayes      *Making expression containing Bayesian one-way ANOVA results.*

---

### Description

Making expression containing Bayesian one-way ANOVA results.

### Usage

```
expr_anova_bayes(data, x, y, paired = FALSE, bf.prior = 0.707, k = 2,
  ...)
```

### Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
bf.prior	A numeric value between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes Factors.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
...	Additional arguments (currently ignored).

### Value

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

### Author(s)

**Indrajeet Patil**

### Examples

```
set.seed(123)

# between-subjects -----
# with defaults
statsExpressions::expr_anova_bayes(
  data = ggplot2::msleep,
  x = vore,
  y = sleep_rem
)
```

```

# modifying the defaults
statsExpressions::expr_anova_bayes(
  data = ggplot2::msleep,
  x = vore,
  y = sleep_rem,
  k = 3,
  bf.prior = 0.8
)

# repeated measures -----
statsExpressions::expr_anova_bayes(
  data = WRS2::WineTasting,
  x = Wine,
  y = Taste,
  paired = TRUE,
  k = 4
)

```

---

expr\_anova\_nonparametric

*Making text subtitle for nonparametric ANOVA.*

---

## Description

Making text subtitle for nonparametric ANOVA.

## Usage

```

expr_anova_nonparametric(data, x, y, paired = FALSE,
  conf.type = "perc", conf.level = 0.95, k = 2, nboot = 100,
  stat.title = NULL, messages = TRUE, ...)

```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
conf.type	A vector of character strings representing the type of intervals required. The value should be any subset of the values "norm", "basic", "perc", "bca". For more, see ?boot::boot.ci.
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
k	Number of digits after decimal point (should be an integer) (Default: k = 2).

nboot	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).
stat.title	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a stat.title argument will be something like "Student's t-test: ".
messages	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
...	Additional arguments (currently ignored).

### Details

For paired designs, the effect size is Kendall's coefficient of concordance ( $W$ ), while for between-subjects designs, the effect size is epsilon-squared (for more, see `?rcompanion::epsilonSquared` and `?rcompanion::kendallW`).

### Value

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

### Author(s)

Indrajeet Patil

### Examples

```
# setup
set.seed(123)
library(statsExpressions)

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

# creating the subtitle
statsExpressions::expr_anova_nonparametric(
  data = bugs_long,
  x = condition,
  y = desire,
  paired = TRUE,
  conf.level = 0.99,
  k = 2
)

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

statsExpressions::expr_anova_nonparametric(
  data = ggplot2::msleep,
  x = vore,
  y = sleep_rem,
  paired = FALSE,
  conf.level = 0.99,
```

```

    conf.type = "perc"
  )

```

---

expr\_anova\_parametric *Making expression containing parametric ANOVA results*

---

## Description

Making expression containing parametric ANOVA results

## Usage

```

expr_anova_parametric(data, x, y, paired = FALSE,
  effsize.type = "unbiased", partial = TRUE, conf.level = 0.95,
  nboot = 100, var.equal = FALSE, sphericity.correction = TRUE,
  k = 2, stat.title = NULL, messages = TRUE, ...)

```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
effsize.type	Type of effect size needed for <i>parametric</i> tests. The argument can be "biased" (equivalent to "d" for Cohen's <i>d</i> for <b>t-test</b> ; "partial_eta" for partial eta-squared for <b>anova</b> ) or "unbiased" (equivalent to "g" Hedge's <i>g</i> for <b>t-test</b> ; "partial_omega" for partial omega-squared for <b>anova</b> ).
partial	Logical that decides if partial eta-squared or omega-squared are returned (Default: TRUE). If FALSE, eta-squared or omega-squared will be returned. Valid only for objects of class <code>lm</code> , <code>aov</code> , <code>anova</code> , or <code>aovlist</code> .
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
nboot	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).
var.equal	a logical variable indicating whether to treat the variances in the samples as equal. If TRUE, then a simple F test for the equality of means in a one-way analysis of variance is performed. If FALSE, an approximate method of Welch (1951) is used, which generalizes the commonly known 2-sample Welch test to the case of arbitrarily many samples.
sphericity.correction	Logical that decides whether to apply correction to account for violation of sphericity in a repeated measures design ANOVA (Default: TRUE).



k	Number of digits after decimal point (should be an integer) (Default: k = 2).
stat.title	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a stat.title argument will be something like "Student's t-test: ".
messages	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
...	Additional arguments (currently ignored).

**Value**

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

**Note**

For repeated measures designs (paired = TRUE), only omega-squared and partial eta-squared effect sizes are supported.

**Author(s)**

**Indrajeet Patil**

**Examples**

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

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

# with defaults
statsExpressions::expr_anova_parametric(
  data = ggplot2::msleep,
  x = vore,
  y = sleep_rem,
  paired = FALSE,
  k = 3
)

# modifying the defaults
statsExpressions::expr_anova_parametric(
  data = ggplot2::msleep,
  x = vore,
  y = sleep_rem,
  paired = FALSE,
  effsize.type = "biased",
  partial = FALSE,
  var.equal = TRUE,
  nboot = 10
)
```

```
# ----- repeated measures -----

statsExpressions::expr_anova_parametric(
  data = iris_long,
  x = condition,
  y = value,
  paired = TRUE,
  k = 4,
  nboot = 10
)
```

---

expr_anova_robust	<i>Expression containing results from heteroscedastic one-way ANOVA for trimmed means</i>
-------------------	---

---

## Description

Expression containing results from heteroscedastic one-way ANOVA for trimmed means

## Usage

```
expr_anova_robust(data, x, y, paired = FALSE, tr = 0.1, nboot = 100,
  conf.level = 0.95, conf.type = "norm", k = 2, stat.title = NULL,
  messages = TRUE, ...)
```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
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.
nboot	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
conf.type	A vector of character strings representing the type of intervals required. The value should be any subset of the values "norm", "basic", "perc", "bca". For more, see ?boot::boot.ci.

k	Number of digits after decimal point (should be an integer) (Default: k = 2).
stat.title	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a stat.title argument will be something like "Student's t-test: ".
messages	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
...	Additional arguments (currently ignored).

**Value**

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

**Author(s)**

Indrajeet Patil

**Examples**

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

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

# going with the defaults
statsExpressions::expr_anova_robust(
  data = ggplot2::midwest,
  x = state,
  y = percbelowpoverty,
  paired = FALSE,
  nboot = 10
)

# changing defaults
expr_anova_robust(
  data = ggplot2::midwest,
  x = state,
  y = percollege,
  paired = FALSE,
  conf.level = 0.99,
  tr = 0.2,
  nboot = 10
)

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

statsExpressions::expr_anova_robust(
  data = iris_long,
  x = condition,
```

```

y = value,
paired = TRUE,
tr = 0.2,
k = 3
)

```

---

expr\_contingency\_tab *Making expression for contingency table and goodness of fit tests*

---

## Description

Making expression for contingency table and goodness of fit tests

## Usage

```

expr_contingency_tab(data, x, y = NULL, counts = NULL, ratio = NULL,
  nboot = 100, paired = FALSE, stat.title = NULL,
  legend.title = NULL, conf.level = 0.95, conf.type = "norm",
  simulate.p.value = FALSE, B = 2000, bias.correct = FALSE, k = 2,
  messages = TRUE, ...)

```

```

expr_onesample_proptest(data, x, y = NULL, counts = NULL,
  ratio = NULL, nboot = 100, paired = FALSE, stat.title = NULL,
  legend.title = NULL, conf.level = 0.95, conf.type = "norm",
  simulate.p.value = FALSE, B = 2000, bias.correct = FALSE, k = 2,
  messages = TRUE, ...)

```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The variable to use as the <b>rows</b> in the contingency table.
y	The variable to use as the <b>columns</b> in the contingency table. Default is NULL. If NULL, one-sample proportion test (a goodness of fit test) will be run for the main variable. Otherwise an appropriate association test will be run.
counts	A string naming a variable in data containing counts, or NULL if each row represents a single observation (Default).
ratio	A vector of proportions: the expected proportions for the proportion test (should sum to 1). Default is NULL, which means the null is equal theoretical proportions across the levels of the nominal variable. This means if there are two levels this will be <code>ratio = c(0.5, 0.5)</code> or if there are four levels this will be <code>ratio = c(0.25, 0.25, 0.25, 0.25)</code> , etc.
nboot	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).

paired	Logical indicating whether data came from a within-subjects or repeated measures design study (Default: FALSE). If TRUE, McNemar's test subtitle will be returned. If FALSE, Pearson's chi-square test will be returned.
stat.title	Title for the effect being investigated with the chi-square test. The default is NULL, i.e. no title will be added to describe the effect being shown. An example of a stat.title argument will be something like "main x condition" or "interaction".
legend.title	Title text for the legend.
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
conf.type	A vector of character strings representing the type of intervals required. The value should be any subset of the values "norm", "basic", "perc", "bca". For more, see ?boot::boot.ci.
simulate.p.value	a logical indicating whether to compute p-values by Monte Carlo simulation.
B	an integer specifying the number of replicates used in the Monte Carlo test.
bias.correct	If TRUE, a bias correction will be applied to Cramer's V.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
messages	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
...	Additional arguments (currently ignored).

## Details

For more details about how the effect sizes and their confidence intervals were computed, see documentation in ?rcompanion::cramerV, ?rcompanion::cramerVFit, and ?rcompanion::cohenG.

## Value

Expression for contingency analysis (Pearson's chi-square test for independence for between-subjects design or McNemar's test for within-subjects design) or goodness of fit test for a single categorical variable.

## Author(s)

Indrajeet Patil

## References

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

## Examples

```
# ----- association tests -----

set.seed(123)
library(statsExpressions)

# without counts data
statsExpressions::expr_contingency_tab(
  data = mtcars,
  x = am,
  y = cyl,
  paired = FALSE,
  nboot = 15
)

# ----- goodness of fit tests -----

# for reproducibility
set.seed(123)

# with counts
expr_contingency_tab(
  data = as.data.frame(HairEyeColor),
  x = Eye,
  counts = Freq,
  ratio = c(0.2, 0.2, 0.3, 0.3)
)
```

---

expr\_corr\_test

*Making expression for correlation analysis*

---

## Description

Making expression for correlation analysis

## Usage

```
expr_corr_test(data, x, y, nboot = 100, beta = 0.1, type = "pearson",
  bf.prior = 0.707, conf.level = 0.95, conf.type = "norm", k = 2,
  stat.title = NULL, messages = TRUE, ...)
```

## 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 column in data containing the explanatory variable to be plotted on the x-axis. Can be entered either as a character string (e.g., "x") or as a bare expression (e.g, x).
y	The column in data containing the response (outcome) variable to be plotted on the y-axis. Can be entered either as a character string (e.g., "y") or as a bare expression (e.g, y).
nboot	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).
beta	bending constant (Default: 0.1). For more, see ?WRS2::pbcor.
type	Type of association between paired samples required ("parametric": Pearson's product moment correlation coefficient" or "nonparametric": Spearman's rho" or "robust": percentage bend correlation coefficient" or "bayes": Bayes Factor for Pearson's r"). Corresponding abbreviations are also accepted: "p" (for parametric/pearson's), "np" (nonparametric/spearman), "r" (robust), "bf" (for bayes factor), resp.
bf.prior	A numeric value between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes Factors.
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
conf.type	A vector of character strings representing the type of intervals required. The value should be any subset of the values "norm", "basic", "perc", "bca". For more, see ?boot::boot.ci.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
stat.title	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a stat.title argument will be something like "Student's t-test: ".
messages	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
...	Additional arguments (currently ignored).

### Value

Expression containing results from correlation test with confidence intervals for the correlation coefficient estimate.

### Author(s)

Indrajeet Patil

### References

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

## Examples

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

# without changing defaults
statsExpressions::expr_corr_test(
  data = ggplot2::midwest,
  x = area,
  y = percblack,
  type = "parametric"
)

# changing defaults
statsExpressions::expr_corr_test(
  data = ggplot2::midwest,
  x = area,
  y = percblack,
  nboot = 25,
  beta = 0.2,
  type = "robust",
  k = 1
)
```

---

expr\_template

*Template for subtitles with statistical details for tests*

---

## Description

Template for subtitles with statistical details for tests

## Usage

```
expr_template(no.parameters, stat.title = NULL, statistic.text,
  statistic, parameter = NULL, parameter2 = NULL, p.value,
  effsize.text, effsize.estimate, effsize.LL, effsize.UL, n,
  conf.level = 0.95, k = 2L, k.parameter = 0L, k.parameter2 = 0L,
  n.text = NULL)
```

## Arguments

- `no.parameters` An integer that specifies that the number of parameters for the statistical test. Can be 0 for non-parametric tests, 1 for tests based on  $t$ -statistic or chi-squared statistic, 2 for tests based on  $F$ -statistic.
- `stat.title` A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a `stat.title` argument will be something like "Student's  $t$ -test: ".



statistic.text	A character that specifies the relevant test statistic. For example, for tests with <i>t</i> -statistic, <code>statistic.text = "t"</code> . If you want to use <code>plotmath</code> , you will have to quote the argument (e.g., <code>quote(italic("t"))</code> ).
statistic	The numeric value of a statistic.
parameter	The numeric value of a parameter being modeled (often degrees of freedom for the test). Default is NULL to accommodate non-parametric tests.
parameter2	Relevant only if the statistic in question has two degrees of freedom (default: NULL).
p.value	The two-sided p-value associated with the observed statistic.
effsize.text	A character that specifies the relevant effect size. For example, for Cohen's <i>d</i> statistic, <code>effsize.text = "d"</code> . If you want to use <code>plotmath</code> , you will have to quote the argument (e.g., <code>quote(italic("d"))</code> ).
effsize.estimate, effsize.LL, effsize.UL	The estimated value of the effect size, its lower bound, and its upper.
n	An integer specifying the sample size used for the test.
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
k.parameter, k.parameter2	Number of decimal places to display for the parameters (default: 0).
n.text	A character that specifies the design, which will determine what the <i>n</i> stands for. For example, for repeated measures, this can be <code>quote(italic("n")["pairs"])</code> , while for independent subjects design this can be <code>quote(italic("n")["obs"])</code> . If NULL, defaults to generic <code>quote(italic("n"))</code> .

### Author(s)

Indrajeet Patil

### Examples

```
set.seed(123)

# subtitle for *t*-statistic with Cohen's *d* as effect size
statsExpressions::expr_template(
  no.parameters = 1L,
  statistic.text = quote(italic("t")),
  statistic = 5.494,
  parameter = 29.234,
  p.value = 0.00001,
  effsize.text = quote(italic("d")),
  effsize.estimate = -1.980,
  effsize.LL = -2.873,
  effsize.UL = -1.088,
  n = 32L,
  conf.level = 0.95,
  k = 3L,
```

```

    k.parameter = 3L
  )

```

---

 expr\_t\_bayes

*Making expression containing Bayesian t-test results*


---

## Description

Making expression containing Bayesian *t*-test results

## Usage

```

expr_t_bayes(data, x, y, bf.prior = 0.707, paired = FALSE, k = 2,
  ...)

```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
bf.prior	A numeric value between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes Factors.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
...	Additional arguments (currently ignored).

## Author(s)

Indrajeet Patil

## References

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

## Examples

```

# for reproducibility
set.seed(123)

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

statsExpressions::expr_t_bayes(
  data = mtcars,

```

```

    x = am,
    y = wt,
    paired = FALSE
  )

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

statsExpressions::expr_t_bayes(
  data = dplyr::filter(
    statsExpressions::intent_morality,
    condition %in% c("accidental", "attempted"),
    harm == "Poisoning"
  ),
  x = condition,
  y = rating,
  paired = TRUE
)

```

---

expr\_t\_nonparametric *Making expression for Mann-Whitney U-test/Wilcoxon test results*

---

## Description

Making expression for Mann-Whitney *U*-test/Wilcoxon test results

## Usage

```
expr_t_nonparametric(data, x, y, paired = FALSE, k = 2,
  conf.level = 0.95, conf.type = "norm", nboot = 100,
  stat.title = NULL, messages = TRUE, ...)
```

```
expr_mann_nonparametric(data, x, y, paired = FALSE, k = 2,
  conf.level = 0.95, conf.type = "norm", nboot = 100,
  stat.title = NULL, messages = TRUE, ...)
```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).

conf.type	A vector of character strings representing the type of intervals required. The value should be any subset of the values "norm", "basic", "perc", "bca". For more, see ?boot::boot.ci.
nboot	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).
stat.title	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a stat.title argument will be something like "Student's t-test: ".
messages	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
...	Additional arguments (currently ignored).

### Details

For the two independent samples case, the Mann-Whitney  $U$ -test is calculated and  $W$  is reported from `stats::wilcox.test`. For the paired samples case the Wilcoxon signed rank test is run and  $V$  is reported.

Since there is no single commonly accepted method for reporting effect size for these tests we are computing and reporting  $r$  (computed as  $Z/\sqrt{N}$ ) along with the confidence intervals associated with the estimate. Note that  $N$  here corresponds to total *sample size* for independent/between-subjects designs, and to total number of *pairs* (and **not observations**) for repeated measures/within-subjects designs.

*Note:* The `stats::wilcox.test` function does not follow the same convention as `stats::t.test`. The sign of the  $V$  test statistic will always be positive since it is **the sum of the positive signed ranks**. Therefore  $V$  will vary in magnitude but not significance based solely on the order of the grouping variable. Consider manually reordering your factor levels if appropriate as shown in the second example below.

### Author(s)

Indrajeet Patil, Chuck Powell

### References

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

### Examples

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

# ----- between-subjects design -----
# simple function call
statsExpressions::expr_t_nonparametric(
  data = sleep,
```

```

    x = group,
    y = extra
  )

  # creating a smaller dataset
  msleep_short <- dplyr::filter(
    .data = ggplot2::msleep,
    vore %in% c("carni", "herbi")
  )

  # modifying few things
  statsExpressions::expr_t_nonparametric(
    data = msleep_short,
    x = vore,
    y = sleep_rem,
    nboot = 200,
    conf.level = 0.99,
    conf.type = "bca"
  )

  # The order of the grouping factor matters when computing *V*
  # Changing default alphabetical order manually
  msleep_short$vore <- factor(msleep_short$vore,
    levels = c("herbi", "carni")
  )

  # note the change in the reported *V* value but the identical
  # value for *p* and the reversed effect size
  statsExpressions::expr_t_nonparametric(
    data = msleep_short,
    x = vore,
    y = sleep_rem
  )

  # ----- within-subjects design -----
  # using dataset included in the package
  statsExpressions::expr_t_nonparametric(
    data = VR_dilemma,
    x = modality,
    y = score,
    paired = TRUE,
    conf.level = 0.90,
    conf.type = "perc",
    nboot = 200,
    k = 5
  )

```

---

 expr\_t\_onesample

*Expression for one sample t-test and its non-parametric and robust equivalents*

---

**Description**

Expression for one sample *t*-test and its non-parametric and robust equivalents

**Usage**

```
expr_t_onesample(data, x, type = "parametric", test.value = 0,
  bf.prior = 0.707, robust.estimator = "onestep", effsize.type = "g",
  effsize.noncentral = TRUE, conf.level = 0.95, conf.type = "norm",
  nboot = 100, k = 2, stat.title = NULL, messages = TRUE, ...)
```

**Arguments**

<code>data</code>	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
<code>x</code>	A numeric variable.
<code>type</code>	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.
<code>test.value</code>	A number specifying the value of the null hypothesis (Default: 0).
<code>bf.prior</code>	A numeric value between 0.5 and 2 (default 0.707), the prior width to use in calculating Bayes Factors.
<code>robust.estimator</code>	If type = "robust", a robust estimator to be used ("onestep" (Default), "mom", or "median"). For more, see ?WRS2::onesampb.
<code>effsize.type</code>	Type of effect size needed for <i>parametric</i> tests. The argument can be "biased" ("d" for Cohen's <i>d</i> ) or "unbiased" ("g" Hedge's <i>g</i> for <b>t-test</b> ). The default is "g".
<code>effsize.noncentral</code>	Logical indicating whether to use non-central <i>t</i> -distributions for computing the confidence interval for Cohen's <i>d</i> or Hedge's <i>g</i> (Default: TRUE).
<code>conf.level</code>	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
<code>conf.type</code>	A vector of character strings representing the type of intervals required. The value should be any subset of the values "norm", "basic", "perc", "bca". For more, see ?boot::boot.ci.
<code>nboot</code>	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).
<code>k</code>	Number of digits after decimal point (should be an integer) (Default: k = 2).
<code>stat.title</code>	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a <code>stat.title</code> argument will be something like "Student's t-test: ".
<code>messages</code>	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
<code>...</code>	Additional arguments (currently ignored).

**Value**

Expression containing results from a one-sample test. The exact test and the effect size details contained will be dependent on the type argument.

**Author(s)**

Indrajeet Patil

**References**

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

**Examples**

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

# ----- parametric -----

statsExpressions::expr_t_onesample(
  data = ggplot2::msleep,
  x = brainwt,
  test.value = 0.275,
  type = "parametric"
)

# ----- non-parametric -----

statsExpressions::expr_t_onesample(
  data = ggplot2::msleep,
  x = brainwt,
  test.value = 0.275,
  type = "nonparametric"
)

# ----- robust -----

statsExpressions::expr_t_onesample(
  data = ggplot2::msleep,
  x = brainwt,
  test.value = 0.275,
  type = "robust"
)

# ----- Bayes Factor -----

statsExpressions::expr_t_onesample(
  data = ggplot2::msleep,
  x = brainwt,
  test.value = 0.275,
```

```

  type = "bayes",
  bf.prior = 0.8
)

```

---

expr\_t\_parametric      *Making expression containing t-test results*

---

## Description

Making expression containing *t*-test results

## Usage

```

expr_t_parametric(data, x, y, paired = FALSE, effsize.type = "g",
  effsize.noncentral = TRUE, conf.level = 0.95, var.equal = FALSE,
  k = 2, stat.title = NULL, ...)

```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
effsize.type	Type of effect size needed for <i>parametric</i> tests. The argument can be "biased" ("d" for Cohen's <i>d</i> ) or "unbiased" ("g" Hedge's <i>g</i> for <b>t-test</b> ). The default is "g".
effsize.noncentral	Logical indicating whether to use non-central <i>t</i> -distributions for computing the confidence interval for Cohen's <i>d</i> or Hedge's <i>g</i> (Default: TRUE).
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
var.equal	a logical variable indicating whether to treat the variances in the samples as equal. If TRUE, then a simple F test for the equality of means in a one-way analysis of variance is performed. If FALSE, an approximate method of Welch (1951) is used, which generalizes the commonly known 2-sample Welch test to the case of arbitrarily many samples.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
stat.title	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a stat.title argument will be something like "Student's t-test: ".
...	Additional arguments (currently ignored).



## Details

Cohen's  $d$  is calculated in the traditional fashion as the difference between means or mean minus  $\mu$  divided by the estimated standardized deviation. By default Hedge's correction is applied  $(N-3)/(N-2.25)$  to produce  $g$ . For independent samples  $t$ -test, there are two possibilities implemented. If the  $t$ -test did not make a homogeneity of variance assumption, (the Welch test), the variance term will mirror the Welch test, otherwise a pooled and weighted estimate is used. If a paired samples  $t$ -test was requested, then effect size desired is based on the standard deviation of the differences.

The computation of the confidence intervals defaults to a use of non-central Student- $t$  distributions (`effsize.noncentral = TRUE`); otherwise a central distribution is used.

When computing confidence intervals the variance of the effect size  $d$  or  $g$  is computed using the conversion formula reported in Cooper et al. (2009)

- $((n1+n2)/(n1*n2) + .5*d^2/df) * ((n1+n2)/df)$  (independent samples)
- $\text{sqrt}(((1/n) + (d^2/n)) * 2 * (1-r))$  (paired case)

## Value

Expression containing details from results of a two-sample test and effect size plus confidence intervals.

## Author(s)

Indrajeet Patil, Chuck Powell

## References

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

## Examples

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

# creating a smaller dataset
msleep_short <- dplyr::filter(
  .data = ggplot2::msleep,
  vore %in% c("carni", "herbi")
)

# with defaults
statsExpressions::expr_t_parametric(
  data = msleep_short,
  x = vore,
  y = sleep_rem
)

# changing defaults
statsExpressions::expr_t_parametric(
  data = msleep_short,
```

```

x = vore,
y = sleep_rem,
var.equal = TRUE,
k = 2,
effsize.type = "d"
)

```

---

expr\_t\_robust

*Expression containing results from a robust t-test*

---

## Description

Expression containing results from a robust *t*-test

## Usage

```

expr_t_robust(data, x, y, tr = 0.1, paired = FALSE, nboot = 100,
  conf.level = 0.95, conf.type = "norm", k = 2, stat.title = NULL,
  messages = TRUE, ...)

```

## Arguments

data	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
x	The grouping variable from the dataframe data.
y	The response (a.k.a. outcome or dependent) variable from the dataframe data.
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.
paired	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.
nboot	Number of bootstrap samples for computing confidence interval for the effect size (Default: 100).
conf.level	Scalar between 0 and 1. If unspecified, the defaults return 95% lower and upper confidence intervals (0.95).
conf.type	A vector of character strings representing the type of intervals required. The value should be any subset of the values "norm", "basic", "perc", "bca". For more, see ?boot::boot.ci.
k	Number of digits after decimal point (should be an integer) (Default: k = 2).
stat.title	A character describing the test being run, which will be added as a prefix in the subtitle. The default is NULL. An example of a stat.title argument will be something like "Student's t-test: ".
messages	Decides whether messages references, notes, and warnings are to be displayed (Default: TRUE).
...	Additional arguments (currently ignored).

**Author(s)**

Indrajeet Patil

**References**

For more details, see- [https://indrajeetpatil.github.io/statsExpressions/articles/stats\\_details.html](https://indrajeetpatil.github.io/statsExpressions/articles/stats_details.html)

**Examples**

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

# with defaults
statsExpressions::expr_t_robust(
  data = sleep,
  x = group,
  y = extra
)

# changing defaults
statsExpressions::expr_t_robust(
  data = ToothGrowth,
  x = supp,
  y = len,
  nboot = 10,
  k = 1,
  tr = 0.2
)

# within-subjects design
statsExpressions::expr_t_robust(
  data = dplyr::filter(
    statsExpressions::intent_morality,
    condition %in% c("accidental", "attempted"),
    harm == "Poisoning"
  ),
  x = condition,
  y = rating,
  paired = TRUE,
  nboot = 25
)
```

**Description**

Moral judgments about third-party moral behavior.

**Usage**

```
intent_morality
```

**Format**

A data frame with 4016 rows and 8 variables

- id. Participant id.
- gender. Participant's gender.
- item. Which story/vignette participants read for a given condition.
- harm. What kind of harm was involved in the item.
- belief. What kind of belief the actor had (neutral or negative/harmful).
- outcome. What kind of outcome the actor caused (neutral or negative/harmful).
- condition. Type of harm, composed of belief and outcome.
- question. Type of moral judgment asked (wrongness or punishment).
- rating. Moral judgment rating on a scale of 1 to 7.

**Details**

This dataset contains data from a recent study about how people judge behavior of others when they unintentionally or intentionally cause harm to others.

Participants responded to four different vignettes that contains four different types of conditions-

- accidental harm. neutral belief, harmful/negative outcome
- intentional harm. harmful/negative belief, harmful/negative outcome
- attempted harm. harmful/negative belief, neutral outcome
- neutral harm. neutral belief, neutral outcome

Additionally, participants saw one of the four variants for each of the four items. Each of the item had a different type of harm.

**Source**

<https://www.nature.com/articles/s41598-017-05299-9>

**Examples**

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

---

`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 "Petal").
- `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.

**Source**

<https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/iris.html>

**Examples**

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

---

`long_to_wide_converter`*Converts long-format dataframe to wide-format dataframe*

---

**Description**

This conversion is helpful mostly for repeated measures design.

**Usage**

```
long_to_wide_converter(data, x, y, paired = TRUE)
```

**Arguments**

<code>data</code>	A dataframe (or a tibble) from which variables specified are to be taken. A matrix or tables will <b>not</b> be accepted.
<code>x</code>	The grouping variable from the dataframe <code>data</code> .
<code>y</code>	The response (a.k.a. outcome or dependent) variable from the dataframe <code>data</code> .
<code>paired</code>	Logical that decides whether the experimental design is repeated measures/within-subjects or between-subjects. The default is FALSE.

**Author(s)**

[Indrajeet Patil](#)

**Examples**

```
statsExpressions::long_to_wide_converter(  
  data = iris_long,  
  x = condition,  
  y = value,  
  paired = TRUE  
)
```

---

`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	<i>Movie information and user ratings from IMDB.com (wide format).</i>
-------------	--

---

**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)
```

---

Titanic\_full

*Titanic dataset.*

---

**Description**

Titanic dataset.



**Usage**

```
Titanic_full
```

**Format**

A data frame with 2201 rows and 5 variables

- id. Dummy identity number for each person.
- Class. 1st, 2nd, 3rd, Crew.
- Sex. Male, Female.
- Age. Child, Adult.
- Survived. No, Yes.

**Details**

This data set provides information on the fate of passengers on the fatal maiden voyage of the ocean liner 'Titanic', summarized according to economic status (class), sex, age and survival.

This is a modified dataset from datasets package.

**Source**

<https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/Titanic.html>

**Examples**

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

---

VR\_dilemma

*Virtual reality moral dilemmas.*

---

**Description**

Virtual reality moral dilemmas.

**Usage**

```
VR_dilemma
```

**Format**

A data frame with 68 rows and 4 variables

- `id`. Dummy identity number for each participant.
- `order`. The order in which the participants completed the two sessions: "text\_first" (0) or "text\_second" (1).
- `modality`. Describes how the moral dilemmas were presented to the participants: either in text format ("text") or in Virtual Reality ("vr").
- `score`. Proportion of "utilitarian" decisions. In other words, of the 4 decisions, how many affirmative were responses. Range: 0 (all utilitarian) - 1 (none utilitarian).

**Details**

Dataset from a study where participants completed identical moral dilemmas in two different sessions held on separate days: in one session, they read text description of the scenario, while in another session they completed the same scenarios in Virtual Reality (videos: <https://www.youtube.com/watch?v=ebdU3HhhYs8>). The study investigated if there was a discrepancy between how people judged the same scenarios while reading them in text versus experiencing them in virtual reality.

**Source**

<https://psyarxiv.com/ry3ap/>

**Examples**

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

# Index

## \*Topic **datasets**

- bugs\_long, 12
  - intent\_morality, 35
  - iris\_long, 37
  - movies\_long, 38
  - movies\_wide, 39
  - Titanic\_full, 40
  - VR\_dilemma, 41
- 
- bf\_contingency\_tab, 2, 5, 9, 10
  - bf\_corr\_test, 4, 5, 9, 10
  - bf\_expr, 6
  - bf\_extractor, 7
  - bf\_one\_sample\_ttest (bf\_ttest), 9
  - bf\_onesample\_proptest
    - (bf\_contingency\_tab), 2
  - bf\_oneway\_anova, 4, 5, 8, 10
  - bf\_ttest, 4, 5, 9, 9
  - bf\_two\_sample\_ttest (bf\_ttest), 9
  - bugs\_long, 12
- 
- expr\_anova\_bayes, 13
  - expr\_anova\_nonparametric, 14
  - expr\_anova\_parametric, 16
  - expr\_anova\_robust, 18
  - expr\_contingency\_tab, 20
  - expr\_corr\_test, 22
  - expr\_mann\_nonparametric
    - (expr\_t\_nonparametric), 27
  - expr\_onesample\_proptest
    - (expr\_contingency\_tab), 20
  - expr\_t\_bayes, 26
  - expr\_t\_nonparametric, 27
  - expr\_t\_onesample, 29
  - expr\_t\_parametric, 32
  - expr\_t\_robust, 34
  - expr\_template, 24
- 
- intent\_morality, 35
  - iris\_long, 37
  - long\_to\_wide\_converter, 38
  - movies\_long, 38
  - movies\_wide, 39
  - Titanic\_full, 40
  - VR\_dilemma, 41