

Package ‘`infern`’

February 13, 2018

Type Package

Title Inferential Statistics

Version 0.3.0

Description

Select set of parametric and non-parametric statistical tests. ‘`infern`’ builds upon the solid set of statistical tests provided in ‘`stats`’ package by including additional data types as inputs, expanding and restructuring the test results. The tests included are t tests, variance tests, proportion tests, chi square tests, Levene’s test, McNemar Test, Cochran’s Q test and Runs test.

Depends R(>= 3.2.4)

License MIT + file LICENSE

URL <https://rsquaredacademy.github.io/infern/>,
<https://github.com/rsquaredacademy/infern>

BugReports <https://github.com/rsquaredacademy/infern/issues>

Imports dplyr, magrittr, purrr, Rcpp, rlang, shiny, tibble, tidyr

Suggests covr, descriptr, haven, jsonlite, knitr, lubridate, readr,
readxl, rmarkdown, scales, shinyBS, shinythemes, stringr,
testthat, tools

Encoding UTF-8

LazyData true

VignetteBuilder knitr

RoxygenNote 6.0.1

LinkingTo Rcpp

NeedsCompilation yes

Author Aravind Hebbali [aut, cre]

Maintainer Aravind Hebbali <hebbali.aravind@gmail.com>

Repository CRAN

Date/Publication 2018-02-13 11:12:37 UTC

R topics documented:

exam	2
hsb	3
infern	3
infer_binom_calc	4
infer_chisq_assoc_test	5
infer_chisq_gof_test	6
infer_cochran_qtest	7
infer_launch_shiny_app	8
infer_levene_test	9
infer_mcnemar_test	10
infer_oneway_anova	12
infer_os_prop_test	13
infer_os_t_test	15
infer_os_var_test	16
infer_runs_test	18
infer_ts_ind_ttest	19
infer_ts_paired_ttest	21
infer_ts_prop_test	23
infer_ts_var_test	24
treatment	26
treatment2	26
Index	27

exam	<i>Dummy data set for Cochran's Q test</i>
------	--

Description

A dataset containing information about results of three exams.

Usage

```
exam
```

Format

A data frame with 15 rows and 3 variables:

exam1 result of exam1

exam2 result of exam2

exam3 result of exam3

Source

<http://www.spss-tutorials.com/spss-cochran-q-test/>

hsb	<i>High School and Beyond Data Set</i>
-----	--

Description

A dataset containing demographic information and standardized test scores of high school students.

Usage

hsb

Format

A data frame with 200 rows and 10 variables:

id id of the student

female gender of the student

race ethnic background of the student

ses socio-economic status of the student

schtyp school type

prog program type

read scores from test of reading

write scores from test of writing

math scores from test of math

science scores from test of science

socst scores from test of social studies

Source

<http://www.ats.ucla.edu/stat/spss/whatstat/whatstat.htm>

inferr	<i>inferr package</i>
--------	-----------------------

Description

Parametric and non parametric statistical tests

Details

See the README on [GitHub](#)

infer_binom_calc *Binomial Test*

Description

Test whether the proportion of successes on a two-level categorical dependent variable significantly differs from a hypothesized value.

Usage

```
infer_binom_calc(n, success, prob = 0.5, ...)
```

```
infer_binom_test(data, variable, prob = 0.5)
```

Arguments

n	number of observations
success	number of successes
prob	assumed probability of success on a trial
...	additional arguments passed to or from other methods
data	a data.frame or a tibble
variable	factor; column in data

Value

binom_test returns an object of class "binom_test". An object of class "binom_test" is a list containing the following components:

n	number of observations
k	number of successes
exp_k	expected number of successes
obs_p	assumed probability of success
exp_p	expected probability of success
lower	lower one sided p value
upper	upper one sided p value

Deprecated Functions

binom_calc() and binom_test() have been deprecated. Instead use infer_binom_calc() and infer_binom_test().

References

Hoel, P. G. 1984. Introduction to Mathematical Statistics. 5th ed. New York: Wiley.

See Also[binom.test](#)**Examples**

```
# using calculator
infer_binom_calc(32, 13, prob = 0.5)

# using data set
infer_binom_test(hsb, female, prob = 0.5)
```

```
infer_chisq_assoc_test
```

Chi Square Test of Association

Description

Chi Square test of association to examine if there is a relationship between two categorical variables.

Usage

```
infer_chisq_assoc_test(data, x, y)
```

Arguments

data	a data.frame or tibble
x	factor; column in data
y	factor; column in data

Value

infer_chisq_assoc_test returns an object of class "infer_chisq_assoc_test". An object of class "infer_chisq_assoc_test" is a list containing the following components:

chi	chi square
chilr	likelihood ratio chi square
chimh	mantel haenszel chi square
chiy	continuity adjusted chi square
sig	p-value of chi square
siglr	p-value of likelihood ratio chi square
sigmh	p-value of mantel haenszel chi square
sigy	p-value of continuity adjusted chi square
phi	phi coefficient
cc	contingency coefficient
cv	cramer's v
ds	product of dimensions of the table of x and y
df	degrees of freedom

Deprecated Function

chisq_test() has been deprecated. Instead use infer_chisq_assoc_test().

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[chisq.test](#)

Examples

```
infer_chisq_assoc_test(hsb, female, schtyp)
```

```
infer_chisq_assoc_test(hsb, female, ses)
```

infer_chisq_gof_test *Chi Square Goodness of Fit Test*

Description

Test whether the observed proportions for a categorical variable differ from hypothesized proportions

Usage

```
infer_chisq_gof_test(data, x, y, correct = FALSE)
```

Arguments

data	a data.frame or tibble
x	factor; column in data
y	expected proportions
correct	logical; if TRUE continuity correction is applied

Value

infer_chisq_gof_test returns an object of class "infer_chisq_gof_test". An object of class "infer_chisq_gof_test" is a list containing the following components:

chisquare	chi square statistic
pvalue	p-value
df	chi square degrees of freedom
ssize	number of observations

names	levels of x
level	number of levels of x
obs	observed frequency/proportion
exp	expected frequency/proportion
deviation	deviation of observed from frequency
std	standardized residuals
varname	name of categorical variable

Deprecated Function

chisq_gof() has been deprecated. Instead use infer_chisq_gof_test()

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[chisq.test](#)

Examples

```
infer_chisq_gof_test(hsb, race, c(20, 20, 20, 140))

# apply continuity correction
infer_chisq_gof_test(hsb, race, c(20, 20, 20, 140), correct = TRUE)
```

infer_cochran_qtest *Cochran Q Test*

Description

Test if the proportions of 3 or more dichotomous variables are equal in the same population.

Usage

```
infer_cochran_qtest(data, ...)
```

Arguments

data	a data.frame or tibble
...	columns in data

Value

infer_cochran_qtest returns an object of class "infer_cochran_qtest". An object of class "infer_cochran_qtest" is a list containing the following components:

n	number of observations
df	degrees of freedom
q	cochran's q statistic
pvalue	p-value

Deprecated Function

cochran_test() has been deprecated. Instead use infer_cochran_qtest().

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

Examples

```
infer_cochran_qtest(exam, exam1, exam2, exam3)
```

```
infer_launch_shiny_app
```

Launch Shiny App

Description

Launches shiny app

Usage

```
infer_launch_shiny_app()
```

Examples

```
## Not run:  
infer_launch_shiny_app()  
  
## End(Not run)
```

infer_levene_test *Levene's test for equality of variances*

Description

infer_levene_test reports Levene's robust test statistic for the equality of variances and the two statistics proposed by Brown and Forsythe that replace the mean in Levene's formula with alternative location estimators. The first alternative replaces the mean with the median. The second alternative replaces the mean with the 10

Usage

```
infer_levene_test(data, ...)

## Default S3 method:
infer_levene_test(data, ..., group_var = NULL,
  trim_mean = 0.1)
```

Arguments

data	a data.frame or tibble
...	numeric; columns in data
group_var	factor; column in data
trim_mean	trimmed mean

Value

infer_levene_test returns an object of class "infer_levene_test". An object of class "infer_levene_test" is a list containing the following components:

bf	Brown and Forsythe f statistic
p_bf	p-value for Brown and Forsythe f statistic
lev	Levene's f statistic
p_lev	p-value for Levene's f statistic
bft	Brown and Forsythe f statistic using trimmed mean
p_bft	p-value for Brown and Forsythe f statistic using trimmed mean
avgs	mean for each level of the grouping variable
sds	standard deviations for each level of the grouping variable
avg	combined mean
sd	combined standard deviation
n	number of observations
n_df	numerator degrees of freedom
d_df	denominator degrees of freedom

levs	levels of the grouping variable
lens	number of observations for each level of the grouping variable
type	alternative hypothesis

Deprecated Function

levene_test() has been deprecated. Instead use infer_levene_test().

References

- Bland, M. 2000. An Introduction to Medical Statistics. 3rd ed. Oxford: Oxford University Press.
- Brown, M. B., and A. B. Forsythe. 1974. Robust tests for the equality of variances. Journal of the American Statistical Association 69: 364–367.
- Carroll, R. J., and H. Schneider. 1985. A note on Levene’s tests for equality of variances. Statistics and Probability Letters 3: 191–194.

Examples

```
# using grouping variable
infer_levene_test(hsb, read, group_var = race)

# using variables
infer_levene_test(hsb, read, write, socst)
```

infer_mcnemar_test *McNemar Test*

Description

Test if the proportions of two dichotomous variables are equal in the same population.

Usage

```
infer_mcnemar_test(data, x = NULL, y = NULL)
```

Arguments

data	a data.frame or tibble
x	factor; column in data
y	factor; column in data

Value

infer_mcnemar_test returns an object of class "infer_mcnemar_test". An object of class "infer_mcnemar_test" is a list containing the following components:

statistic	chi square statistic
df	degrees of freedom
pvalue	p-value
exactp	exact p-value
cstat	continuity correction chi square statistic
cpvalue	continuity correction p-value
kappa	kappa coefficient; measure of interrater agreement
std_err	asymptotic standard error
kappa_cil	95% kappa lower confidence limit
kappa_ciu	95% kappa upper confidence limit
cases	cases
controls	controls
ratio	ratio of proportion with factor
odratio	odds ratio
tbl	two way table

Deprecated Function

mcnemar_test() has been deprecated. Instead use infer_mcnemar_test().

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[mcnemar.test](#)

Examples

```
# using variables from data
library(dplyr)
hb <- mutate(hsb,
             himath = if_else(math > 60, 1, 0),
             hiread = if_else(read > 60, 1, 0)
            )
infer_mcnemar_test(hb, himath, hiread)

# test if the proportion of students in himath and hiread group is same
himath <- ifelse(hsb$math > 60, 1, 0)
hiread <- ifelse(hsb$read > 60, 1, 0)
```

```
infer_mcnemar_test(table(himath, hiread))

# using matrix
infer_mcnemar_test(matrix(c(135, 18, 21, 26), nrow = 2))
```

infer_oneway_anova *One Way ANOVA*

Description

One way analysis of variance

Usage

```
infer_oneway_anova(data, x, y, ...)
```

Arguments

data	a data.frame or a tibble
x	numeric; column in data
y	factor; column in data
...	additional arguments passed to or from other methods

Value

owanova returns an object of class "owanova". An object of class "owanova" is a list containing the following components:

between	between group sum of squares
within	within group sum of squares
total	total sum of squares
df_btw	between groups degress of freedom
df_within	within groups degress of freedom
df_total	total degress of freedom
ms_btw	between groups mean square
ms_within	within groups mean square
f	f value
p	p value
r2	r squared value
ar2	adjusted r squared value
sigma	root mean squared error
obs	number of observations
tab	group statistics

Deprecated Functions

owanova() has been deprecated. Instead use infer_oneway_anova().

References

Kutner, M. H., Nachtsheim, C., Neter, J., & Li, W. (2005). Applied linear statistical models. Boston: McGraw-Hill Irwin.

See Also

[anova](#)

Examples

```
infer_oneway_anova(mtcars, mpg, cyl)
infer_oneway_anova(hsb, write, prog)
```

infer_os_prop_test *One Sample Test of Proportion*

Description

infer_os_prop_test compares proportion in one group to a specified population proportion.

Usage

```
infer_os_prop_test(data, variable = NULL, prob = 0.5, phat = 0.5,
  alternative = c("both", "less", "greater", "all"))

## Default S3 method:
infer_os_prop_test(data, variable = NULL, prob = 0.5,
  phat = 0.5, alternative = c("both", "less", "greater", "all"))
```

Arguments

data	numeric vector of length 1 or a data.frame or tibble
variable	factor; column in data
prob	hypothesised proportion
phat	observed proportion
alternative	a character string specifying the alternative hypothesis, must be one of "both" (default), "greater", "less" or "all". You can specify just the initial letter.

Value

`infer_os_prop_test` returns an object of class "`infer_os_prop_test`". An object of class "`infer_os_prop_test`" is a list containing the following components:

<code>n</code>	number of observations
<code>phat</code>	proportion of 1's
<code>p</code>	assumed probability of success
<code>z</code>	z statistic
<code>sig</code>	p-value for z statistic
<code>alt</code>	alternative hypothesis
<code>obs</code>	observed number of 0's and 1's
<code>exp</code>	expected number of 0's and 1's
<code>deviation</code>	deviation of observed from expected
<code>std</code>	standardized residuals

Deprecated Function

`prop_test()` has been deprecated. Instead use `infer_os_prop_test()`.

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[prop.test](#) [binom.test](#)

Examples

```
# use as a calculator
infer_os_prop_test(200, prob = 0.5, phat = 0.3)

# using data set
infer_os_prop_test(hsb, female, prob = 0.5)
```

infer_os_t_test	<i>One Sample t Test</i>
-----------------	--------------------------

Description

infer_os_t_test performs t tests on the equality of means. It tests the hypothesis that a sample has a mean equal to a hypothesized value.

Usage

```
infer_os_t_test(data, x, mu = 0, alpha = 0.05, alternative = c("both",
  "less", "greater", "all"), ...)
```

Arguments

data	a data.frame or tibble
x	numeric; column in data
mu	a number indicating the true value of the mean
alpha	acceptable tolerance for type I error
alternative	a character string specifying the alternative hypothesis, must be one of "both" (default), "greater", "less" or "all". You can specify just the initial letter
...	additional arguments passed to or from other methods

Value

infer_os_t_test returns an object of class "infer_os_t_test". An object of class "infer_os_t_test" is a list containing the following components:

mu	a number indicating the true value of the mean
n	number of observations
df	degrees of freedom
Mean	observed mean of x
stddev	standard deviation of x
std_err	estimate of standard error
test_stat	t statistic
confint	confidence interval for the mean
mean_diff	mean difference
mean_diff_l	lower confidence limit for mean difference
mean_diff_u	upper confidence limit for mean difference
p_l	lower one-sided p-value
p_u	upper one-sided p-value
p	two sided p-value
conf	confidence level
type	alternative hypothesis
var_name	name of x

Deprecated Function

`ttest()` has been deprecated. Instead use `infer_os_t_test()`.

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[t.test](#)

Examples

```
# lower tail
infer_os_t_test(hsb, write, mu = 50, alternative = 'less')

# upper tail
infer_os_t_test(hsb, write, mu = 50, alternative = 'greater')

# both tails
infer_os_t_test(hsb, write, mu = 50, alternative = 'both')

# all tails
infer_os_t_test(hsb, write, mu = 50, alternative = 'all')
```

`infer_os_var_test` *One Sample Variance Comparison Test*

Description

`infer_os_var_test` performs tests on the equality of standard deviations (variances). It tests that the standard deviation of a sample is equal to a hypothesized value.

Usage

```
infer_os_var_test(data, x, sd, confint = 0.95, alternative = c("both",
  "less", "greater", "all"), ...)
```

Arguments

<code>data</code>	a data.frame or tibble
<code>x</code>	numeric; column in data
<code>sd</code>	hypothesised standard deviation
<code>confint</code>	confidence level
<code>alternative</code>	a character string specifying the alternative hypothesis, must be one of "both" (default), "greater", "less" or "all". You can specify just the initial letter
<code>...</code>	additional arguments passed to or from other methods

Value

`infer_os_var_test` returns an object of class "`infer_os_var_test`". An object of class "`infer_os_var_test`" is a list containing the following components:

<code>n</code>	number of observations
<code>sd</code>	hypothesised standard deviation of <code>x</code>
<code>sigma</code>	observed standard deviation
<code>se</code>	estimated standard error
<code>chi</code>	chi-square statistic
<code>df</code>	degrees of freedom
<code>p_lower</code>	lower one-sided p-value
<code>p_upper</code>	upper one-sided p-value
<code>p_two</code>	two-sided p-value
<code>xbar</code>	mean of <code>x</code>
<code>c_lwr</code>	lower confidence limit of standard deviation
<code>c_upr</code>	upper confidence limit of standard deviation
<code>var_name</code>	name of <code>x</code>
<code>conf</code>	confidence level
<code>type</code>	alternative hypothesis

Deprecated Function

`os_vartest()` has been deprecated. Instead use `infer_os_var_test()`.

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[var.test](#)

Examples

```
# lower tail
infer_os_var_test(mtcars, mpg, 5, alternative = 'less')

# upper tail
infer_os_var_test(mtcars, mpg, 5, alternative = 'greater')

# both tails
infer_os_var_test(mtcars, mpg, 5, alternative = 'both')

# all tails
infer_os_var_test(mtcars, mpg, 5, alternative = 'all')
```

infer_runs_test *Test for Random Order*

Description

runtest tests whether the observations of x are serially independent i.e. whether they occur in a random order, by counting how many runs there are above and below a threshold. By default, the median is used as the threshold. A small number of runs indicates positive serial correlation; a large number indicates negative serial correlation.

Usage

```
infer_runs_test(data, x, drop = FALSE, split = FALSE, mean = FALSE,
  threshold = NA)
```

Arguments

data	a data.frame or tibble
x	numeric; column in data
drop	logical; if TRUE, values equal to the threshold will be dropped from x
split	logical; if TRUE, data will be recoded in binary format
mean	logical; if TRUE, mean will be used as threshold
threshold	threshold to be used for counting runs, specify 0 if data is coded as a binary.

Value

infer_runs_test returns an object of class "infer_runs_test". An object of class "infer_runs_test" is a list containing the following components:

n	number of observations
threshold	within group sum of squares
n_below	number below the threshold
n_above	number above the threshold
mean	expected number of runs
var	variance of the number of runs
n_runs	number of runs
z	z statistic
p	p-value of z

Deprecated Function

runs_test() has been deprecated. Instead use infer_runs_test().

References

- Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.
- Edgington, E. S. 1961. Probability table for number of runs of signs of first differences in ordered series. Journal of the American Statistical Association 56: 156–159.
- Madansky, A. 1988. Prescriptions for Working Statisticians. New York: Springer.
- Swed, F. S., and C. Eisenhart. 1943. Tables for testing randomness of grouping in a sequence of alternatives. Annals of Mathematical Statistics 14: 66–87.

Examples

```
infer_runs_test(hsb, read)

infer_runs_test(hsb, read, drop = TRUE)

infer_runs_test(hsb, read, split = TRUE)

infer_runs_test(hsb, read, mean = TRUE)

infer_runs_test(hsb, read, threshold = 0)
```

infer_ts_ind_ttest *Two Independent Sample t Test*

Description

infer_ts_ind_ttest compares the means of two independent groups in order to determine whether there is statistical evidence that the associated population means are significantly different.

Usage

```
infer_ts_ind_ttest(data, x, y, confint = 0.95, alternative = c("both",
  "less", "greater", "all"), ...)
```

Arguments

data	a data frame
x	factor; a column in data
y	numeric; a column in data
confint	confidence level
alternative	a character string specifying the alternative hypothesis, must be one of "both" (default), "greater", "less" or "all". You can specify just the initial letter
...	additional arguments passed to or from other methods

Value

`infer_ts_ind_ttest` returns an object of class "`infer_ts_ind_ttest`". An object of class "`infer_ts_ind_ttest`" is a list containing the following components:

<code>levels</code>	levels of x
<code>obs</code>	number of observations of y for each level of x
<code>n</code>	total number of observations
<code>mean</code>	mean of y for each level of x
<code>sd</code>	standard deviation of y for each level of x
<code>se</code>	estimate of standard error of y for each level of x
<code>lower</code>	lower limit for the mean of y for each level of x
<code>upper</code>	upper limit for the mean of y for each level of x
<code>combined</code>	a data frame; mean, standard deviation, standard error and confidence limit of mean of y
<code>mean_diff</code>	difference in mean of y for the two groups of x
<code>se_dif</code>	estimate of the standard error for difference in mean of y for the two groups of x
<code>sd_dif</code>	degrees of freedom
<code>conf_diff</code>	confidence interval for <code>mean_diff</code>
<code>df_pooled</code>	degrees of freedom for the pooled method
<code>df_satterthwaite</code>	degrees of freedom for the Satterthwaite method
<code>t_pooled</code>	t statistic for the pooled method
<code>t_satterthwaite</code>	t statistic for the Satterthwaite method
<code>sig_pooled</code>	two-sided p-value for the pooled method
<code>sig_pooled_l</code>	lower one-sided p-value for the pooled method
<code>sig_pooled_u</code>	upper one-sided p-value for the pooled method
<code>sig</code>	two-sided p-value for the Satterthwaite method
<code>sig_l</code>	lower one-sided p-value for the Satterthwaite method
<code>sig_u</code>	upper one-sided p-value for the Satterthwaite method
<code>num_df</code>	numerator degrees of freedom for folded f test
<code>den_df</code>	denominator degrees of freedom for folded f test
<code>f</code>	f value for the equality of variances test
<code>f_sig</code>	p-value for the folded f test
<code>var_y</code>	name of y
<code>confint</code>	confidence level
<code>alternative</code>	alternative hypothesis

Deprecated Function

ind_ttest() has been deprecated. Instead use infer_ts_ind_ttest().

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[t.test](#)

Examples

```
# lower tail
infer_ts_ind_ttest(hsb, female, write, alternative = 'less')

# upper tail
infer_ts_ind_ttest(hsb, female, write, alternative = 'greater')

# both tails
infer_ts_ind_ttest(hsb, female, write, alternative = 'both')

# all tails
infer_ts_ind_ttest(hsb, female, write, alternative = 'all')
```

infer_ts_paired_ttest *Paired t test*

Description

infer_ts_paired_ttest tests that two samples have the same mean, assuming paired data.

Usage

```
infer_ts_paired_ttest(data, x, y, confint = 0.95, alternative = c("both",
  "less", "greater", "all"))
```

Arguments

data	a data.frame or tibble
x	numeric; column in data
y	numeric; column in data
confint	confidence level
alternative	a character string specifying the alternative hypothesis, must be one of "both" (default), "greater", "less" or "all". You can specify just the initial letter.

Value

infer_ts_paired_ttest returns an object of class "infer_ts_paired_ttest". An object of class "infer_ts_paired_ttest" is a list containing the following components:

Obs	number of observations
b	mean, standard deviation and standard error of x, y and their difference
tstat	t statistic
p_lower	lower one-sided p-value
p_upper	upper one-sided p-value
p_two_tail	two sided p-value
corr	Correlation of x and y
corsig	p-value of correlation test
conf_int1	confidence interval for mean of x
conf_int2	confidence interval for mean of y
conf_int_diff	confidence interval for mean of difference of x and y
df	degrees of freedom
confint	confidence level
alternative	alternative hypothesis
var_names	names of x and y
xy	string used in printing results of the test

Deprecated Function

paired_ttest() has been deprecated. Instead use infer_ts_paired_ttest().

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[t.test](#)

Examples

```
# lower tail
infer_ts_paired_ttest(hsb, read, write, alternative = 'less')

# upper tail
infer_ts_paired_ttest(hsb, read, write, alternative = 'greater')

# both tails
infer_ts_paired_ttest(hsb, read, write, alternative = 'both')

# all tails
infer_ts_paired_ttest(hsb, read, write, alternative = 'all')
```

infer_ts_prop_test *Two Sample Test of Proportion*

Description

Tests on the equality of proportions using large-sample statistics. It tests that a sample has the same proportion within two independent groups or two samples have the same proportion.

Usage

```
infer_ts_prop_test(data, var1, var2, alternative = c("both", "less",
  "greater", "all"), ...)
```

```
infer_ts_prop_grp(data, var, group, alternative = c("both", "less", "greater",
  "all"))
```

```
infer_ts_prop_calc(n1, n2, p1, p2, alternative = c("both", "less", "greater",
  "all"), ...)
```

Arguments

data	a data.frame or tibble
var1	factor; column in data
var2	factor; column in data
alternative	a character string specifying the alternative hypothesis, must be one of "both" (default), "greater", "less" or "all". You can specify just the initial letter
...	additional arguments passed to or from other methods
var	factor; column in data
group	factor; column in data
n1	sample 1 size
n2	sample 2 size
p1	sample 1 proportion
p2	sample 2 proportion

Value

an object of class "infer_ts_prop_test". An object of class "infer_ts_prop_test" is a list containing the following components:

n1	sample 1 size
n2	sample 2 size
phat1	sample 1 proportion
phat2	sample 2 proportion

z	z statistic
sig	p-value for z statistic
alt	alternative hypothesis

Deprecated Functions

ts_prop_test(), ts_prop_grp() and ts_prop_calc() have been deprecated. Instead use infer_ts_prop_test(), infer_ts_prop_grp() and infer_ts_prop_calc().

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[prop.test](#)

Examples

```
# using variables
# lower tail
infer_ts_prop_test(treatment, treatment1, treatment2,
alternative = 'less')

# using groups
# lower tail
infer_ts_prop_grp(treatment2, outcome, female,
alternative = 'less')

# using sample size and proportions
# lower tail
infer_ts_prop_calc(n1 = 30, n2 = 25, p1 = 0.3, p2 = 0.5, alternative = 'less')
```

infer_ts_var_test *Two Sample Variance Comparison Test*

Description

infer_ts_var_test performs tests on the equality of standard deviations (variances).

Usage

```
infer_ts_var_test(data, ..., group_var = NULL, alternative = c("less",
"greater", "all"))
```


Arguments

data	a data.frame or tibble
...	numeric; column(s) in data
group_var	factor; column in data
alternative	a character string specifying the alternative hypothesis, must be one of "both" (default), "greater", "less" or "all". You can specify just the initial letter.

Value

infer_ts_var_test returns an object of class "infer_ts_var_test". An object of class "infer_ts_var_test" is a list containing the following components:

f	f statistic
lower	lower one-sided p-value
upper	upper one-sided p-value
two_tail	two-sided p-value
vars	variances for each level of the grouping variable
avgs	means for each level of the grouping variable
sds	standard deviations for each level of the grouping variable
ses	standard errors for each level of the grouping variable
avg	combined mean
sd	combined standard deviation
se	estimated combined standard error
n1	numerator degrees of freedom
n2	denominator degrees of freedom
lens	number of observations for each level of grouping variable
len	number of observations
lev	levels of the grouping variable
type	alternative hypothesis

Deprecated Function

var_test() has been deprecated. Instead use infer_ts_var_test().

References

Sheskin, D. J. 2007. Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition. : Chapman & Hall/CRC.

See Also

[var.test](#)

Examples

```
# using grouping variable
infer_ts_var_test(hsb, read, group_var = female, alternative = 'less')

# using two variables
infer_ts_var_test(hsb, read, write, alternative = 'less')
```

treatment	<i>Dummy data set for 2 Sample Proportion test</i>
-----------	--

Description

A dataset containing information about two treatments

Usage

```
treatment
```

Format

A data frame with 50 rows and 2 variables:

treatment1 result of treatment type 1

treatment2 result of treatment type 2

treatment2	<i>Dummy data set for 2 Sample Proportion test</i>
------------	--

Description

A dataset containing information about treatment outcomes

Usage

```
treatment2
```

Format

A data frame with 200 rows and 2 variables:

outcome outcome of treatment

female gender of patient, 0 for male and 1 for female

Index

*Topic **datasets**

exam, [2](#)
hsb, [3](#)
treatment, [26](#)
treatment2, [26](#)

anova, [13](#)

binom.test, [5](#), [14](#)
binom_calc (infer_binom_calc), [4](#)
binom_test (infer_binom_calc), [4](#)

chisq.test, [6](#), [7](#)
chisq_gof (infer_chisq_gof_test), [6](#)
chisq_test (infer_chisq_assoc_test), [5](#)
cochran_test (infer_cochran_qtest), [7](#)

exam, [2](#)

hsb, [3](#)

ind_ttest (infer_ts_ind_ttest), [19](#)
infer_binom_calc, [4](#)
infer_binom_test (infer_binom_calc), [4](#)
infer_chisq_assoc_test, [5](#)
infer_chisq_gof_test, [6](#)
infer_cochran_qtest, [7](#)
infer_launch_shiny_app, [8](#)
infer_levene_test, [9](#)
infer_mcnemar_test, [10](#)
infer_oneway_anova, [12](#)
infer_os_prop_test, [13](#)
infer_os_t_test, [15](#)
infer_os_var_test, [16](#)
infer_runs_test, [18](#)
infer_ts_ind_ttest, [19](#)
infer_ts_paired_ttest, [21](#)
infer_ts_prop_calc
 (infer_ts_prop_test), [23](#)
infer_ts_prop_grp (infer_ts_prop_test),
 [23](#)

infer_ts_prop_test, [23](#)
infer_ts_var_test, [24](#)
inferr, [3](#)
inferr-package (inferr), [3](#)

launch_inferr (infer_launch_shiny_app),
 [8](#)
levene_test (infer_levene_test), [9](#)

mcnemar.test, [11](#)
mcnemar_test (infer_mcnemar_test), [10](#)

os_vartest (infer_os_var_test), [16](#)
owanova (infer_oneway_anova), [12](#)

paired_ttest (infer_ts_paired_ttest), [21](#)
prop.test, [14](#), [24](#)
prop_test (infer_os_prop_test), [13](#)

runs_test (infer_runs_test), [18](#)

t.test, [16](#), [21](#), [22](#)
treatment, [26](#)
treatment2, [26](#)
ts_prop_calc (infer_ts_prop_test), [23](#)
ts_prop_grp (infer_ts_prop_test), [23](#)
ts_prop_test (infer_ts_prop_test), [23](#)
ttest (infer_os_t_test), [15](#)

var.test, [17](#), [25](#)
var_test (infer_ts_var_test), [24](#)