Package ‘confintr’

January 28, 2022

Type Package

Title Confidence Intervals

Version 0.1.2

Date 2022-01-28

Maintainer Michael Mayer <mayermichael79@gmail.com>

Description Calculates classic and/or bootstrap confidence intervals for many parameters such as the population mean, variance, interquartile range (IQR), median absolute deviation (MAD), skewness, kurtosis, Cramer's V, odds ratio, R-squared, quantiles (incl. median), proportions, different types of correlation measures, difference in means, quantiles and medians. Many of the classic confidence intervals are described in Smithson, M. (2003, ISBN: 978-0761924999). Bootstrap confidence intervals are calculated with the R package 'boot'. Both one- and two-sided intervals are supported.

License GPL (>= 2)

Depends R (>= 3.1.0)

Encoding UTF-8

RoxygenNote 7.1.1

URL https://github.com/mayer79/confintr

BugReports https://github.com/mayer79/confintr/issues

Suggests rmarkdown, knitr, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Imports boot, stats

NeedsCompilation no

Author Michael Mayer [aut, cre]

Repository CRAN

Date/Publication 2022-01-28 20:00:02 UTC

R topics documented:

`ci_boot` *Bootstrap Confidence Interval*

Description

Internel wrapper to `boot::boot.ci`. Contains the logic for one-sided and asymmetric confidence intervals.

Usage

```
ci_boot(S, boot_type = c("norm", "basic", "stud", "perc", "bca"), probs, ...)
```

Arguments

`S`	Result of `boot`.
`boot_type`	Confidence interval type passed to `boot::boot.ci`.
`probs`	Probability cuts for the confidence intervals.
`...`	Further arguments passed to `boot::boot.ci`.

Value

A numeric vector of length two.

---

ci_chisq_ncp — *Confidence Interval for the Non-Centrality Parameter of the Chi-Squared Distribution*

Description

This function calculates confidence intervals for the non-centrality parameter of the chi-squared distribution based on chi-squared test inversion or the bootstrap. A positive lower (1-alpha)*100%-confidence limit for the ncp goes hand-in-hand with a significant association test at level alpha.

Usage

```
ci_chisq_ncp(
  x,
  probs = c(0.025, 0.975),
  correct = TRUE,
  type = c("chi-squared", "bootstrap"),
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

`x`	The result of `stats::chisq.test`, a `table/matrix` of frequencies, or a `data.frame` with exactly two columns.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
`correct`	Should Yates continuity correction be applied to the 2x2 case? The default is `TRUE` (also used in the bootstrap).
`type`	Type of confidence interval. One of "chi-squared" (default) or "bootstrap".
`boot_type`	Type of bootstrap confidence interval ("bca", "perc", "norm", "basic"). Only used for `type = "bootstrap"`.
`R`	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
`seed`	An integer random seed. Only used for `type = "bootstrap"`.
`...`	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188). Note that large chi-squared test statistics might provide unreliable results with method "chi-squared" (see `?pchisq`).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Smithson, M. (2003). Confidence intervals. Series: Quantitative Applications in the Social Sciences. New York, NY: Sage Publications.
2. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
3. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

See Also

`ci_cramersv`.

Examples

```
ci_chisq_ncp(mtcars[c("am", "vs")])
ci_chisq_ncp(mtcars[c("am", "vs")], type = "bootstrap", R = 999)
ir <- iris
ir$PL <- ir$Petal.Width > 1
ci_chisq_ncp(ir[, c("Species", "PL")])
ci_chisq_ncp(ir[, c("Species", "PL")], probs = c(0.05, 1))
```

`ci_cor` *Confidence Interval for Correlation Coefficients*

Description

This function calculates confidence intervals for a population correlation coefficient. For Pearson correlation, "normal" confidence intervals are available (by `stats::cor.test`). Also bootstrap confidence intervals are supported and are the only option for rank correlations.

Usage

```
ci_cor(
  x,
  y = NULL,
  probs = c(0.025, 0.975),
  method = c("pearson", "kendall", "spearman"),
  type = c("normal", "bootstrap"),
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

`x`	A numeric vector or a `matrix/data.frame` with exactly two numeric columns.
`y`	A numeric vector (only used if `x` is a vector).
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
`method`	Type of correlation coefficient, one of "pearson" (default), "kendall", or "spearman". For the latter two, only bootstrap confidence intervals are supported. The names can be abbreviated.
`type`	Type of confidence interval. One of "normal" (the default) or "bootstrap" (the only option for rank-correlations).
`boot_type`	Type of bootstrap confidence interval ("bca", "perc", "norm", "basic"). Only used for `type = "bootstrap"`.
`R`	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
`seed`	An integer random seed. Only used for `type = "bootstrap"`.
`...`	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
2. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

Examples

```
ci_cor(iris[1:2])
ci_cor(iris[1:2], type = "bootstrap", R = 999, seed = 1)
ci_cor(iris[1:2], method = "spearman", type = "bootstrap", R = 999, seed = 1)
ci_cor(iris[1:2], method = "k", type = "bootstrap", R = 999, seed = 1)
```

---

ci_cramersv — *Confidence Interval for the Population Cramer's V*

Description

This function calculates confidence intervals for the population Cramer's V. By default, a parametric approach based on the non-centrality parameter (ncp) of the chi-squared distribution is utilized. Alternatively, bootstrap confidence intervals are available, also by boostrapping confidence intervals for the ncp.

Usage

```
ci_cramersv(
  x,
  probs = c(0.025, 0.975),
  type = c("chi-squared", "bootstrap"),
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  test_adjustment = TRUE,
  ...
)
```

Arguments

x	The result of `stats::chisq.test`, a matrix/table of counts or a `data.frame` with exactly two columns representing the two variables.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. One of "chi-squared" (default) or "bootstrap".
boot_type	Type of bootstrap confidence interval ("bca", "perc", "norm", "basic"). Only used for `type = "bootstrap"`.
R	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
seed	An integer random seed. Only used for `type = "bootstrap"`.

test_adjustment
: Adjustment to allow for test of association, see Details. The default is TRUE.

...
: Further arguments passed to resample::CI.boot_type.

Details

A positive lower (1-alpha)*100%-confidence limit for the ncp goes hand-in-hand with a significant association test at level alpha. In order to allow such test approach also with Cramer's V, if the lower bound for the ncp is 0, we round down to 0 also the lower bound for Cramer's V. Without this slightly conservative adjustment, the lower limit for V would always be positive since ci for V = sqrt((ci for ncp + df)/(n (k - 1))), where k is the smaller number of levels in the two variables (see Smithson for this formula). Use `test_adjustment = FALSE` to switch off this behaviour. Note that this is also a reason to bootstrap V via ncp instead of directly bootstrapping V. Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188). Note that no continuity correction is applied for 2x2 tables. Further note that large chi-squared test statistics might provide unreliable results with method "chi-squared" (see `?pchisq`).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Smithson, M. (2003). Confidence intervals. Series: Quantitative Applications in the Social Sciences. New York, NY: Sage Publications.
2. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
3. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

See Also

`ci_chisq_ncp`.

Examples

```
ir <- iris
ir$PL <- ir$Petal.Width > 1
ci_cramersv(ir[, c("Species", "PL")])
ci_cramersv(ir[, c("Species", "PL")], type = "bootstrap", R = 999)
ci_cramersv(ir[, c("Species", "PL")], probs = c(0.05, 1))
```

```
ci_cramersv(mtcars[c("am", "vs")])
ci_cramersv(mtcars[c("am", "vs")], test_adjustment = FALSE)
```

---

ci_f_ncp *Confidence Interval for the Non-Centrality Parameter of the F Distribution*

Description

Based on the inversion principle, parametric confidence intervals for the non-centrality parameter Delta of the F distribution are calculated. Note that we do not provide bootstrap confidence intervals here to keep the input interface simple. A positive lower (1-alpha)*100%-confidence limit for the ncp goes hand-in-hand with a significant F test at level alpha.

Usage

```
ci_f_ncp(x, df1 = NULL, df2 = NULL, probs = c(0.025, 0.975))
```

Arguments

`x`	The result of `lm` or the F test statistic.
`df1`	The numerator degree of freedom, e.g. the number of parameters (including the intercept) of a linear regression. Only used if `x` is a test statistic.
`df2`	The denominator degree of freedom, e.g. n - df1 - 1 in a linear regression. Only used if `x` is a test statistic.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.

Details

Note that, according to `?pf`, the results might be unreliable for very large F values.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

Smithson, M. (2003). Confidence intervals. Series: Quantitative Applications in the Social Sciences. New York, NY: Sage Publications.

See Also

ci_rsquared.

Examples

```
fit <- lm(Sepal.Length ~ ., data = iris)
ci_f_ncp(fit)
ci_f_ncp(fit, probs = c(0.05, 1))
ci_f_ncp(fit, probs = c(0, 0.95))
ci_f_ncp(x = 188.251, df1 = 5, df2 = 144)
```

---

ci_IQR *Confidence Interval for the Interquartile Range*

Description

This function calculates bootstrap confidence intervals for the population interquartile range (IQR), i.e. the difference between first and third quartile.

Usage

```
ci_IQR(
  x,
  probs = c(0.025, 0.975),
  type = "bootstrap",
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. Currently not used as the only type is "bootstrap".
boot_type	Type of bootstrap confidence interval c("bca", "perc", "norm", "basic").
R	The number of bootstrap resamples.
seed	An integer random seed.
...	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
2. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

Examples

```
set.seed(1)
x <- rnorm(100)
ci_IQR(x, R = 999)
```

---

ci_kurtosis *Confidence Interval for the Kurtosis*

Description

This function calculates bootstrap confidence intervals for the population kurtosis, see Details. Note that we use the version of the kurtosis that equals 3 for a theoretical normal distribution.

Usage

```
ci_kurtosis(
  x,
  probs = c(0.025, 0.975),
  type = "bootstrap",
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

`x`	A numeric vector.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
`type`	Type of confidence interval. Currently not used as the only type is "bootstrap".
`boot_type`	Type of bootstrap confidence interval c("bca", "perc", "norm", "basic").
`R`	The number of bootstrap resamples.
`seed`	An integer random seed.
`...`	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
2. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

See Also

`moments`, `ci_skewness`.

Examples

```
set.seed(1)
x <- rnorm(100)
ci_kurtosis(x, R = 999)
```

ci_mad *Confidence Interval for the Median Absolute Deviation*

Description

This function calculates bootstrap confidence intervals for the population median absolute deviation, see `stats::mad` for more information on this measure of scale.

Usage

```
ci_mad(
  x,
  probs = c(0.025, 0.975),
  constant = 1.4826,
  type = "bootstrap",
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

`x`	A numeric vector.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
`constant`	Scaling factor applied. The default (1.4826) ensures that the MAD equals the standard deviation for a theoretical normal distribution.
`type`	Type of confidence interval. Currently not used as the only type is "bootstrap".
`boot_type`	Type of bootstrap confidence interval c("bca", "perc", "norm", "basic").
`R`	The number of bootstrap resamples.
`seed`	An integer random seed.
`...`	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.

- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
2. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

Examples

```
set.seed(1)
x <- rnorm(100)
ci_mad(x, R = 999)
```

---

ci_mean — *Confidence Interval for the Population Mean*

Description

This function calculates confidence intervals for the population mean. By default, Student's t method is used. Alternatively, Wald and bootstrap confidence intervals are available.

Usage

```
ci_mean(
  x,
  probs = c(0.025, 0.975),
  type = c("t", "Wald", "bootstrap"),
  boot_type = c("stud", "bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

`x`	A numeric vector.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
`type`	Type of confidence interval. One of "t" (default), "Wald", or "bootstrap".
`boot_type`	Type of bootstrap confidence interval ("stud", "bca", "perc", "norm", "basic"). Only used for `type = "bootstrap"`.

R	The number of bootstrap resamples. Only used for type = "bootstrap".
seed	An integer random seed. Only used for type = "bootstrap".
...	Further arguments passed to boot::boot.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type for the mean is "stud" (bootstrap t) as it enjoys the property of being second order accurate and has a stable variance estimator (see Efron, p. 188).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Smithson, M. (2003). Confidence intervals. Series: Quantitative Applications in the Social Sciences. New York, NY: Sage Publications.
2. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
3. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

Examples

```
x <- 1:100
ci_mean(x)
ci_mean(x, type = "bootstrap", R = 999, seed = 1)
ci_mean(x, type = "bootstrap", R = 999, probs = c(0.025, 1), seed = 1)
ci_mean(x, type = "bootstrap", R = 999, probs = c(0, 0.975), seed = 1)
```

---

ci_mean_diff *Confidence Interval for the Population Mean Difference*

Description

This function calculates confidence intervals for the population value of mean(x) - mean(y). The default is Student's method with Welch's correction for unequal variances, but also bootstrap confidence intervals are available.

Usage

```
ci_mean_diff(
  x,
  y,
  probs = c(0.025, 0.975),
  var.equal = FALSE,
  type = c("t", "bootstrap"),
  boot_type = c("stud", "bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector.
y	A numeric vector.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
var.equal	Should the two variances be treated as being equal? The default is `FALSE`. If `TRUE`, the pooled variance is used to estimate the variance of the mean difference. Otherweise, Welch's approach is used. This also applies to the "stud" boostrap.
type	Type of confidence interval. One of "t" (default), or "bootstrap".
boot_type	Type of bootstrap confidence interval ("stud", "bca", "perc", "norm", "basic"). Only used for `type = "bootstrap"`.
R	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
seed	An integer random seed. Only used for `type = "bootstrap"`.
...	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot". The default bootstrap type for the mean difference is "stud" (bootstrap t) as it enjoys the property of being second order accurate and has a stable variance estimator (see Efron, p. 188). The resampling is done within sample. If `boot_type = "stud"`, the standard error is estimated by Welch's method if `var.equal = FALSE` (the default) and by pooling otherwise. Thus, `var.equal` has not only an effect for the classic Student approach (`type = "t"`) but also for `boot_type = "stud"`.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.

- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
2. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

Examples

```
x <- 10:30
y <- 1:30
ci_mean_diff(x, y)
t.test(x, y)$conf.int
ci_mean_diff(x, y, type = "bootstrap", R = 999)
```

---

ci_median — *Confidence Interval for the Population Median*

Description

This function calculates confidence intervals for the population median by calling `ci_quantile(...,q = 0.5)`. See `ci_quantile` for details.

Usage

```
ci_median(
  x,
  probs = c(0.025, 0.975),
  type = c("binomial", "bootstrap"),
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

`x`	A numeric vector.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
`type`	Type of confidence interval. One of "binomial" (default), or "bootstrap".
`boot_type`	Type of bootstrap confidence interval ("bca", "perc", "norm", "basic"). Only used for `type = "bootstrap"`.
`R`	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
`seed`	An integer random seed. Only used for `type = "bootstrap"`.
`...`	Further arguments passed to `boot::boot`.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

See Also

`ci_quantile`.

Examples

```
ci_median(1:100)
ci_quantile(1:100)
```

---

ci_median_diff   *Confidence Interval for the Population Median Difference of two Samples*

Description

This function calculates bootstrap confidence intervals for the population value of median(x) - median(y) by calling ci_quantile_diff(, q = 0.5). See `ci_quantile_diff` for details.

Usage

```
ci_median_diff(
  x,
  y,
  probs = c(0.025, 0.975),
  type = "bootstrap",
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector.
y	A numeric vector.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. Currently, "bootstrap" is the only option.
boot_type	Type of bootstrap confidence interval ("bca", "perc", "norm", "basic").
R	The number of bootstrap resamples.
seed	An integer random seed.
...	Further arguments passed to `boot::boot`.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

See Also

`ci_quantile_diff`.

Examples

```
x <- 10:30
y <- 1:30
ci_median_diff(x, y, R = 999)
```

---

ci_oddsratio *Confidence Interval for the Odds Ratio*

Description

This function calculates a confidence interval for the odds ratio in a 2x2 table/matrix or a data frame with two columns. The confidence interval is obtained through `stats::fisher.test`. Bootstrap confidence intervals are not available.

Usage

```
ci_oddsratio(x, probs = c(0.025, 0.975))
```

Arguments

`x`	A 2x2 `table/matrix` of frequencies, or a `data.frame` with exactly two columns.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

See Also

`oddsratio`.

Examples

```
x <- cbind(c(10, 5), c(4, 4))
ci_oddsratio(x)
```

`ci_proportion` *Confidence Interval for a Population Proportion*

Description

This function calculates confidence intervals for a population proportion. By default, "Clopper-Pearson" confidence intervals are calculated (via `stats::binom.test`). Further possibilities are "Wilson", "Agresti-Coull", and "bootstrap" (mainly added for consistency and didactic purposes).

Usage

```
ci_proportion(
  x,
  n = NULL,
  probs = c(0.025, 0.975),
  type = c("Clopper-Pearson", "Agresti-Coull", "Wilson", "bootstrap"),
  boot_type = c("bca", "perc", "stud", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector of 0 and 1 or the number of successes.
n	The sample size. Only needed if x is a vector of length 1.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. One of "Clopper-Pearson" (the default), "Agresti–Coull", "Wilson", "bootstrap".
boot_type	Type of bootstrap confidence interval ("bca", "perc", "stud", "norm", "basic"). Only used for type = "bootstrap".
R	The number of bootstrap resamples. Only used for type = "bootstrap".
seed	An integer random seed. Only used for type = "bootstrap".
...	Further arguments passed to boot::boot.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type for the proportion is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188). Note that we use the formula in https://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval which does not simplify the 0.975 quantile of the normal by 2 as sometimes in other references.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Clopper, C. and Pearson, E. S. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. Biometrika. 26 (4).
2. Wilson, E. B. (1927). Probable inference, the law of succession, and statistical inference. Journal of the American Statistical Association, 22 (158).
3. Agresti, A. and Coull, B. A. (1998). Approximate is better than 'exact' for interval estimation of binomial proportions. The American Statistician, 52 (2).
4. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
5. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

Examples

```
x <- rep(0:1, times = c(50, 100))
ci_proportion(x)
ci_proportion(x, type = "Wilson")
ci_proportion(x, type = "Agresti-Coull")
ci_proportion(x, type = "bootstrap", R = 999)
```

`ci_quantile` *Confidence Interval for a Population Quantile*

Description

This function calculates confidence intervals for a population quantile. By default, distribution-free confidence intervals based on the binomial distribution are formed, see Hahn and Meeker. Alternatively, bootstrap confidence intervals are available.

Usage

```
ci_quantile(
  x,
  q = 0.5,
  probs = c(0.025, 0.975),
  type = c("binomial", "bootstrap"),
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

`x`	A numeric vector.
`q`	A single probability value determining the quantile. Set to 0.5 for the median (the default).
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
`type`	Type of confidence interval. One of "binomial" (default), or "bootstrap".
`boot_type`	Type of bootstrap confidence interval ("bca", "perc", "norm", "basic"). Only used for `type = "bootstrap"`.
`R`	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
`seed`	An integer random seed. Only used for `type = "bootstrap"`.
`...`	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Hahn, G. and Meeker, W. (1991). Statistical Intervals. Wiley 1991.
2. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
3. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

See Also

`ci_quantile`.

Examples

```
x <- 1:100
ci_quantile(x)
ci_quantile(x, q = 0.25)
ci_quantile(x, q = 0.25, type = "bootstrap", R = 999)
```

`ci_quantile_diff` *Confidence Interval for the Population Quantile Difference of two Samples*

Description

This function calculates bootstrap confidence intervals for the population value of q quantile(x) - q quantile(y).

Usage

```
ci_quantile_diff(
  x,
  y,
  q = 0.5,
  probs = c(0.025, 0.975),
  type = "bootstrap",
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector.
y	A numeric vector.
q	A single probability value determining the quantile. Set to 0.5 for the median (default).
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. Currently, "bootstrap" is the only option.
boot_type	Type of bootstrap confidence interval ("bca", "perc", "norm", "basic").
R	The number of bootstrap resamples.
seed	An integer random seed.
...	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot". The default bootstrap type is "bca" (bias-corrected & accelerated) as it enjoys the property of being second order accurate and is transformation respecting (see Efron, p. 188). The sampling is done within sample.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
2. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

See Also

`ci_median_diff`.

Examples

```
x <- 10:30
y <- 1:30
ci_quantile_diff(x, y, R = 999)
```

`ci_rsquared` *Confidence Interval for the Population R-Squared*

Description

This function calculates parametric confidence intervals for the population R-squared. It is based on confidence intervals for the non-centrality parameter Delta of the F distribution, found by test inversion. Delta values are mapped to R-squared by R-squared = Delta / (Delta + df1 + df2 + 1), where df1 and df2 are the degrees of freedom of the F test statistic. A positive lower (1-alpha)*100%-confidence limit for the R-squared goes hand-in-hand with a significant F test at level alpha.

Usage

```
ci_rsquared(x, df1 = NULL, df2 = NULL, probs = c(0.025, 0.975))
```

Arguments

`x`	The result of `stats::lm` or the F test statistic.
`df1`	The numerator degree of freedom. Only used if `x` is a test statistic.
`df2`	The denominator degree of freedom. Only used if `x` is a test statistic.
`probs`	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.

Details

According to `?pf`, the results might be unreliable for very large F values. Note that we do not provide bootstrap confidence intervals here to keep the input interface simple.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

Smithson, M. (2003). Confidence intervals. Series: Quantitative Applications in the Social Sciences. New York, NY: Sage Publications.

See Also

`ci_f_ncp`.

Examples

```
fit <- lm(Sepal.Length ~ ., data = iris)
summary(fit)$r.squared
ci_rsquared(fit)
ci_rsquared(fit, probs = c(0.05, 1))
ci_rsquared(fit, probs = c(0, 0.95))
ci_rsquared(188.251, 5, 144)
```

---

ci_sd — *Confidence Interval for the Population Standard Deviation*

Description

This function calculates confidence intervals for the population standard deviation. They are derived by calculating confidence intervals for the variance and then taking the square-root. For details, see `ci_var`.

Usage

```
ci_sd(
  x,
  probs = c(0.025, 0.975),
  type = c("chi-squared", "bootstrap"),
  boot_type = c("bca", "perc", "stud", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. One of "chi-squared" (default) or "bootstrap".
boot_type	Type of bootstrap confidence interval ("bca", "perc", "stud", "norm", "basic"). Only used for `type = "bootstrap"`.
R	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
seed	An integer random seed. Only used for `type = "bootstrap"`.
...	Further arguments passed to `boot::boot`.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

See Also

`ci_var`.

Examples

```
x <- 1:100
sd(x)
ci_sd(x)
sqrt(ci_var(x)$interval)
ci_sd(x, type = "bootstrap", R = 999)
```

---

ci_skewness *Confidence Interval for the Skewness*

Description

This function calculates bootstrap confidence intervals for the population skewness, see Details.

Usage

```
ci_skewness(
  x,
  probs = c(0.025, 0.975),
  type = "bootstrap",
  boot_type = c("bca", "perc", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. Currently not used as the only type is "bootstrap".
boot_type	Type of bootstrap confidence interval c("bca", "perc", "norm", "basic").
R	The number of bootstrap resamples.
seed	An integer random seed.
...	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188).

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
2. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

See Also

`moments`, `ci_kurtosis`.

Examples

```
set.seed(1)
x <- rnorm(100)
ci_skewness(x, R = 999)
```

---

ci_var *Confidence Interval for the Population Variance*

Description

This function calculates confidence intervals for the population variance. By default, classic confidence intervals are calculated based on the chi-squared distribution, assuming normal distribution (see Smithson). Bootstrap confidence intervals are also available and are recommended for the non-normal case as the chi-squared confidence intervals are sensitive to deviations from normality.

Usage

```
ci_var(
  x,
  probs = c(0.025, 0.975),
  type = c("chi-squared", "bootstrap"),
  boot_type = c("bca", "perc", "stud", "norm", "basic"),
  R = 9999,
  seed = NULL,
  ...
)
```

Arguments

x	A numeric vector.
probs	Error probabilites. The default c(0.025, 0.975) gives a symmetric 95% confidence interval.
type	Type of confidence interval. One of "chi-squared" (default) or "bootstrap".
boot_type	Type of bootstrap confidence interval ("bca", "perc", "stud", "norm", "basic"). Only used for `type = "bootstrap"`.
R	The number of bootstrap resamples. Only used for `type = "bootstrap"`.
seed	An integer random seed. Only used for `type = "bootstrap"`.
...	Further arguments passed to `boot::boot`.

Details

Bootstrap confidence intervals are calculated by the package "boot", see references. The default bootstrap type is "bca" (bias-corrected accelerated) as it enjoys the property of being second order accurate as well as transformation respecting (see Efron, p. 188). The "stud" (bootstrap t) bootstrap uses a general formula for the standard error of the sample variance given in Wilks.

Value

A list with class `cint` containing these components:

- `parameter`: The parameter in question.
- `interval`: The confidence interval for the parameter.
- `estimate`: The estimate for the parameter.
- `probs`: A vector of error probabilities.
- `type`: The type of the interval.
- `info`: An additional description text for the interval.

References

1. Smithson, M. (2003). Confidence intervals. Series: Quantitative Applications in the Social Sciences. New York, NY: Sage Publications.
2. S.S. Wilks (1962), Mathematical Statistics, Wiley & Sons.
3. Efron, B. and Tibshirani R. J. (1994). An Introduction to the Bootstrap. Chapman & Hall/CRC.
4. Canty, A and Ripley B. (2019). boot: Bootstrap R (S-Plus) Functions.

See Also

`ci_sd`.

Examples

```
x <- 1:100
ci_var(x)
ci_var(x, type = "bootstrap", R = 999)
```

---

cramersv *Cramer's V*

Description

This function calculates Cramer's V, a measure of association between two categorical variables. It is a scaled version of the chi-squared test statistic and lies between 0 and 1. Cramer's V is calculated as sqrt(chi-squared / (n * (k - 1))), where n is the number of observations and k is the smaller of the number of levels of the two variables.

Usage

```
cramersv(x)
```

Arguments

`x`	The result of `stats::chisq.test`, a matrix/table of counts or a `data.frame` with exactly two columns representing the two variables.

Details

Yates continuity correction is never applied. So in the 2x2 case, if `x` is the result of `stats::chisq.test`, make sure no continuity correction was applied. Otherwise, results can be inconsistent.

Value

A numeric vector of length one.

References

Cramer, Harald. 1946. Mathematical Methods of Statistics. Princeton: Princeton University Press, page 282 (Chapter 21. The two-dimensional case).

Examples

```
tab <- table(mtcars[c("am", "vs")])
chi <- chisq.test(tab, correct = FALSE)
cramersv(mtcars[c("am", "vs")])
cramersv(chi)
cramersv(tab)
```

`is.cint` *Type Check*

Description

Checks if an object inherits class 'cint'.

Usage

```
is.cint(x)
```

Arguments

`x`	Any object.

Value

A logical vector of length one.

Examples

```
is.cint(ci_proportion(5, 20))
```

moments *Moments*

Description

Functions to calculate moments, skewness, and Pearson's measure of kurtosis. The latter is defined as the ratio of the 4th central moment and the squared second central moment. For a theoretical normal distribution, the kurtosis equals 3.

Usage

```
moment(z, p = 1, central = TRUE, na.rm = TRUE)

skewness(z, na.rm = TRUE)

kurtosis(z, na.rm = TRUE)
```

Arguments

z	Numeric vector.
p	Order of moment.
central	Should central moment be calculated? Default is TRUE.
na.rm	Should missing values be removed? Default is TRUE. Otherwise, the result will be NA if missing values are present.

Value

A numeric vector of length one.

Examples

```
x <- 1:100
moment(x, 4)
skewness(x)
kurtosis(x)
```

oddsratio *Odds Ratio*

Description

This function calculates the odds ratio in a 2x2 table/matrix or a data frame with two columns. The numerator equals the ratio of the top left entry and the bottom left entry, while the denonimator equals the ratio of the top right entry and the bottom right entry. This is usually slightly different from the calculation by `stats::fisher.test` which is based on the ML estimate of the odds ratio.

Usage

```
oddsratio(x)
```

Arguments

x	A 2x2 matrix/table of counts or a `data.frame` with exactly two columns representing the two binary variables.

Value

A numeric vector of length one.

Examples

```
tab <- cbind(c(10, 5), c(4, 4))
oddsratio(tab)
```

`print.cint` *Print a cint Object*

Description

Print method for an object of class `cint`.

Usage

```
## S3 method for class 'cint'
print(x, digits = getOption("digits"), ...)
```

Arguments

x	A on object of class `cint`.
digits	Number of digits used to format sample estimate and confidence limits.
...	Further arguments passed from other methods.

Value

Invisibly, the input is returned.

Examples

```
ci_mean(1:100)
```

`se` *Standard errors*

Description

Functions to calculate standard errors of different statistics. The availability of a standard error (or statistic proportional to it) allows to apply "stud" (bootstrap t) bootstrap.

Usage

```
se_mean(z, na.rm = TRUE, ...)

se_mean_diff(z, y, na.rm = TRUE, var.equal = FALSE, ...)

se_var(z, na.rm = TRUE, ...)

se_proportion(z, na.rm = TRUE, ...)
```

Arguments

`z`	Numeric vector.
`na.rm`	Should missing values be removed before calculation? The default is `TRUE` for convenience.
`...`	Further arguments to be passed from other methods.
`y`	Numeric vector.
`var.equal`	Should the two variances be treated as being equal? The default is `FALSE`. If `TRUE`, the pooled variance is used to estimate the variance of the mean difference. Otherweise, Welch's approach is used (see `stats::t.test`). This also applies to the "stud" bootstrap.

Value

A numeric vector of length one.

Examples

```
se_mean(1:100)
```

Index