Package ‘cvcqv’

August 6, 2019

Type Package
Title Coefficient of Variation (CV) with Confidence Intervals (CI)
Version 1.0.0
Date 2019-08-01
Maintainer Maani Beigy <manibeygi@gmail.com>
Depends R (>= 3.1.2), dplyr (>= 0.8.0.1)
Imports R6, SciViews, boot, MBESS
Suggests testthat, knitr, rmarkdown, covr
VignetteBuilder knitr
URL https://github.com/MaaniBeigy/cvcqv
BugReports https://github.com/MaaniBeigy/cvcqv/issues
License GPL-3
Encoding UTF-8
LazyData true
RoxygenNote 6.1.1
NeedsCompilation no
Author Maani Beigy [aut, cre]
Repository CRAN
Date/Publication 2019-08-06 09:30:06 UTC
**R topics documented:**

BootCoefQuartVar .................................................. 2
BootCoefVar ............................................................ 3
CoefQuartVar ............................................................ 4
CoefQuartVarCI ........................................................... 4
CoefVar ................................................................. 6
CoefVarCI ................................................................. 7
cqv_versatile ............................................................. 9
cv_versatile .............................................................. 11
SampleQuantiles ......................................................... 13

**Index**

14

---

**Description**

The R6 class `BootCoefQuartVar` produces the bootstrap resampling for the coefficient of quartile variation (cqv) of the given numeric vectors. It uses `boot` from the package `boot`. Also, it produces the bootstrap confidence intervals for the cqv based on the `boot.ci` from the package `boot`.

**Arguments**

- `x`: An R object. Currently there are methods for numeric vectors
- `na.rm`: a logical value indicating whether NA values should be stripped before the computation proceeds.
- `alpha`: The allowed type I error probability
- `R`: integer indicating the number of bootstrap replicates.

**References**


**Examples**

```r
x <- c(0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4,
       4.6, 5.4, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9)
cqv_x <- BootCoefQuartVar$new(x)
cqv_x$boot_cqv()
```

BootCoefVar

R6 Bootstrap Resampling for Coefficient of Variation

Description

The R6 class `BootCoefVar` produces the bootstrap resampling for the coefficient of variation (cv) of the given numeric vectors. It uses `boot` and `boot.ci` from the package `boot`.

Arguments

- `x` An R object. Currently there are methods for numeric vectors
- `na.rm` a logical value indicating whether NA values should be stripped before the computation proceeds.
- `alpha` The allowed type I error probability
- `R` integer indicating the number of bootstrap replicates.

References


Examples

```r
x <- c(
  0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4,
  4.6, 5.4, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9
)
cv_x <- BootCoefVar$new(x)
cv_x$boot_cv()
cv_x$boot_cv_corr()
cv_x$boot_basic_ci_cv()
cv_x$boot_norm_ci_cv()
cv_x$boot_perc_ci_cv()
cv_x$boot_bca_ci_cv()
cv_x$boot_basic_ci_cv_corr()
cv_x$boot_norm_ci_cv_corr()
cv_x$boot_perc_ci_cv_corr()
cv_x$boot_bca_ci_cv_corr()
R6::is.R6(cv_x)
```
Description

The R6 class CoefQuartVar for the coefficient of quartile variation (cqv)

Arguments

- `x`: An R object. Currently there are methods for numeric vectors
- `na.rm`: a logical value indicating whether NA values should be stripped before the computation proceeds.
- `digits`: integer indicating the number of decimal places to be used.

Details

**Coefficient of Quartile Variation**  
cqv is a measure of relative dispersion that is based on interquartile range (iqr). Since cqv is unitless, it is useful for comparison of variables with different units. It is also a measure of homogeneity [1].

References


Examples

```r
x <- c(0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4,
       4.6, 5.4, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9)
CoefQuartVar$new(x)$est()
cqv_x <- CoefQuartVar$new(x, digits = 2)
cqv_x$est()
R6::is.R6(cqv_x)
```

Description

The R6 class CoefQuartVarCI for the confidence intervals of coefficient of quartile variation (cqv)
Arguments

- **x**: An R object. Currently there are methods for numeric vectors.
- **na.rm**: A logical value indicating whether NA values should be stripped before the computation proceeds.
- **digits**: An integer indicating the number of decimal places to be used.
- **methods**: The available computation methods of confidence intervals are: "bonett_ci", "norm_ci", "basic_ci", "perc_ci", "bca_ci" or "all_ci".
- **R**: An integer indicating the number of bootstrap replicates.

Details

**Coefficient of Quartile Variation** The `cqv` is a measure of relative dispersion that is based on interquartile range (iqr). Since `cqv` is unitless, it is useful for comparison of variables with different units. It is also a measure of homogeneity [1, 2].

Value

An object of type "list" which contains the estimate, the intervals, and the computation method. It has two components:

- **$method**: A description of statistical method used for the computations.
- **$statistics**: A data frame representing three vectors: est, lower and upper limits of 95% confidence interval (CI):
  
  - **est**: `cqv*100`
  - **Bonett 95% CI**: It uses a centering adjustment which helps to equalize the tail error probabilities [1, 2].
  - **Normal approximation 95% CI**: The intervals calculated by the normal approximation [3, 4], using `boot.ci`.
  - **Basic bootstrap 95% CI**: The intervals calculated by the basic bootstrap method [3, 4], using `boot.ci`.
  - **Bootstrap percentile 95% CI**: The intervals calculated by the bootstrap percentile method [3, 4], using `boot.ci`.
  - **Adjusted bootstrap percentile (BCa) 95% CI**: The intervals calculated by the adjusted bootstrap percentile (BCa) method [3, 4], using `boot.ci`.

References

Examples

```r
y <- c(0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4, 4.6, 5.4, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9)
CoefQuartVarCI$new(x = y)$bonett_ci()
cqv_y <- CoefQuartVarCI$new(x = y, alpha = 0.05, R = 1000, digits = 2)
cqv_y$bonett_ci()
R6::is.R6(cqv_y)
```

---

**Description**

The R6 class `CoefVar` for the coefficient of variation (cv)

**Arguments**

- **x**: An R object. Currently there are methods for numeric vectors
- **na.rm**: a logical value indicating whether NA values should be stripped before the computation proceeds.
- **digits**: integer indicating the number of decimal places to be used.

**Details**

**Coefficient of Variation** The `cv` is a measure of relative dispersion representing the degree of variability relative to the mean [1]. Since `cv` is unitless, it is useful for comparison of variables with different units. It is also a measure of homogeneity [1].

**References**

Examples

```r
x <- c(0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4, 4.6, 5.4, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9)
CoefVar$new(x)$est()
cv_x <- CoefVar$new(x, digits = 2)
cv_x$est()
cv_x$est_corr()
R6::is.R6(cv_x)
```

---

**Description**

The R6 class `CoefVarCI` for the confidence intervals of coefficient of variation (cv)

**Arguments**

- **x**: An R object. Currently there are methods for numeric vectors
- **na.rm**: a logical value indicating whether NA values should be stripped before the computation proceeds.
- **digits**: integer indicating the number of decimal places to be used.
- **method**: a scalar representing the type of confidence intervals required. The value should be any of the values "kelley_ci", "mckay_ci", "miller_ci", "vangel_ci", "mahmoudvand_hassani_ci", "equal_tailed_ci", "shortest_length_ci", "normal_approximation_ci", "norm_ci", "basic_ci", or "all_ci".
- **alpha**: The allowed type I error probability
- **R**: integer indicating the number of bootstrap replicates.
- **correction**: returns the unbiased estimate of the coefficient of variation if TRUE is determined.

**Details**

**Coefficient of Variation** The cv is a measure of relative dispersion representing the degree of variability relative to the mean [1]. Since cv is unitless, it is useful for comparison of variables with different units. It is also a measure of homogeneity [1].

**Value**

An object of type "list" which contains the estimate, the intervals, and the computation method. It has two main components:

- **$method**: A description of statistical method used for the computations.
$statistics$ A data frame representing three vectors: est, lower and upper limits of confidence interval (CI); additional description vector is provided when "all" is selected:

**est:** \(cv*100\)

**Kelley Confidence Interval:** Thanks to package MBESS [2] for the computation of confidence limits for the noncentrality parameter from a \(t\) distribution conf.limits.nct [3].

**McKay Confidence Interval:** The intervals calculated by the method introduced by McKay [4], using chi-square distribution.

**Miller Confidence Interval:** The intervals calculated by the method introduced by Miller [5], using the standard normal distribution.

**Vangel Confidence Interval:** Vangel [6] proposed a method for the calculation of CI for \(cv\); which is a modification on McKay’s CI.

**Mahmoudvand-Hassani Confidence Interval:** Mahmoudvand and Hassani [7] proposed a new CI for \(cv\); which is obtained using ranked set sampling (RSS)

**Normal Approximation Confidence Interval:** Wararit Panichkitkosolkul [8] proposed another CI for \(cv\); which is a normal approximation.

**Shortest-Length Confidence Interval:** Wararit Panichkitkosolkul [8] proposed another CI for \(cv\); which is obtained through minimizing the length of CI.

**Equal-Tailed Confidence Interval:** Wararit Panichkitkosolkul [8] proposed another CI for \(cv\); which is obtained using chi-square distribution.

**Bootstrap Confidence Intervals:** Thanks to package boot by Canty & Ripley [9] we can obtain bootstrap CI around \(cv\) using boot.ci.

References


Examples

```r
y <- c(
  0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4,
  4.6, 5.4, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9
)
CoefVarCI$new(x = y)$kelley_ci()
cv_y <- CoefVarCI$new(
  x = y,
  alpha = 0.05,
  R = 1000,
  digits = 2,
  correction = TRUE
)
cv_y$kelley_ci()
cv_y$mckay_ci()
R6::is.R6(cv_y)
```

---

**cqv_versatile Coefficient of Quartile Variation (cqv)**

**Description**

Versatile function for the coefficient of quartile variation (cqv)

**Arguments**

- **x**  
  An R object. Currently there are methods for numeric vectors
- **na.rm**  
  a logical value indicating whether NA values should be stripped before the computation proceeds.
- **digits**  
  integer indicating the number of decimal places to be used.
- **method**  
  a scalar representing the type of confidence intervals required. The value should be any of the values "bonett", "norm", "basic", "perc", "bca" or "all".
- **R**  
  integer indicating the number of bootstrap replicates.
Details

Coefficient of Quartile Variation The cqv is a measure of relative dispersion that is based on interquartile range (iqr). Since cqv is unitless, it is useful for comparison of variables with different units. It is also a measure of homogeneity [1, 2].

Value

An object of type "list" which contains the estimate, the intervals, and the computation method. It has two components:

$method A description of statistical method used for the computations.

$statistics A data frame representing three vectors: est, lower and upper limits of 95% confidence interval (CI):

est: cqv*100

Bonett 95% CI: It uses a centering adjustment which helps to equalize the tail error probabilities [1, 2].

Normal approximation 95% CI: The intervals calculated by the normal approximation [3, 4], using boot.ci.

Basic bootstrap 95% CI: The intervals calculated by the basic bootstrap method [3, 4], using boot.ci.

Bootstrap percentile 95% CI: The intervals calculated by the bootstrap percentile method [3, 4], using boot.ci.

Adjusted bootstrap percentile (BCa) 95% CI: The intervals calculated by the adjusted bootstrap percentile (BCa) method [3, 4], using boot.ci.

References


Examples

```r
x <- c(0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4,
       4.6, 5.4, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9)
cqv_versatile(x)
```
cv_versatile

cqv_versatile(x, na.rm = TRUE, digits = 2)
cqv_versatile(x, na.rm = TRUE, digits = 2, method = "bonett")

---

**cv_versatile**  
*Coefficient of Variation (cv)*

### Description
Versatile function for the coefficient of variation (cv)

### Arguments
- **x**: An R object. Currently there are methods for numeric vectors
- **na.rm**: a logical value indicating whether NA values should be stripped before the computation proceeds.
- **digits**: integer indicating the number of decimal places to be used.
- **method**: a scalar representing the type of confidence intervals required. The value should be any of the values "kelley", "mckay", "miller", "vangel", "mahmoudvand_hassani", "equal_tailed", "shortest_length", "normal_approximation", "norm", "basic", or "all".
- **correction**: returns the unbiased estimate of the coefficient of variation
- **alpha**: The allowed type I error probability
- **R**: integer indicating the number of bootstrap replicates.

### Details

**Coefficient of Variation**  The *cv* is a measure of relative dispersion representing the degree of variability relative to the mean [1]. Since *cv* is unitless, it is useful for comparison of variables with different units. It is also a measure of homogeneity [1].

### Value
An object of type "list" which contains the estimate, the intervals, and the computation method. It has two main components:

- **$method**: A description of statistical method used for the computations.
- **$statistics**: A data frame representing three vectors: est, lower and upper limits of confidence interval (CI); additional description vector is provided when "all" is selected:

  **est**: cv*100

  **Kelley Confidence Interval**: Thanks to package MBESS [2] for the computation of confidence limits for the noncentrality parameter from a *t* distribution conf.limits.nct [3].

  **McKay Confidence Interval**: The intervals calculated by the method introduced by McKay
[4], using chi-square distribution.

**Miller Confidence Interval:** The intervals calculated by the method introduced by Miller [5], using the standard normal distribution.

**Vangel Confidence Interval:** Vangel [6] proposed a method for the calculation of CI for $cv$; which is a modification on McKay’s CI.

**Mahmoudvand-Hassani Confidence Interval:** Mahmoudvand and Hassani [7] proposed a new CI for $cv$; which is obtained using ranked set sampling (RSS)

**Normal Approximation Confidence Interval:** Wararit Panichkitkosolkul [8] proposed another CI for $cv$; which is a normal approximation.

**Shortest-Length Confidence Interval:** Wararit Panichkitkosolkul [8] proposed another CI for $cv$; which is obtained through minimizing the length of CI.

**Equal-Tailed Confidence Interval:** Wararit Panichkitkosolkul [8] proposed another CI for $cv$; which is obtained using chi-square distribution.

**Bootstrap Confidence Intervals:** Thanks to package **boot** by Canty & Ripley [9] we can obtain bootstrap CI around $cv$ using **boot.ci**.

**References**


**Examples**

```r
x <- c(0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4,
       4.6, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9)

cv_versatile(x)
cv_versatile(x, correction = TRUE)
cv_versatile(x, na.rm = TRUE, digits = 3, method = "kelley", correction = TRUE)
cv_versatile(x, na.rm = TRUE, method = "mahmoudvand_hassani", correction = TRUE)
```

---

**Description**

The R6 class `SampleQuantiles` produces the sample quantiles corresponding to the given probabilities. It uses `quantile` from the package `stats`.

**Arguments**

- **x**: An R object. Currently there are methods for numeric vectors
- **na.rm**: a logical value indicating whether NA values should be stripped before the computation proceeds.
- **digits**: integer indicating the number of decimal places to be used.
- **probs**: numeric vector of probabilities with values in \([0, 1]\).
- **names**: logical; if TRUE, the result has a `names` attribute regarding the percentiles.
- **type**: an integer between 1 and 9 selecting one of the nine quantile algorithms explained in `quantile` to be used.

**Examples**

```r
x <- c(0.2, 0.5, 1.1, 1.4, 1.8, 2.3, 2.5, 2.7, 3.5, 4.4,
       4.6, 5.4, 5.7, 5.8, 5.9, 6.0, 6.6, 7.1, 7.9)

SampleQuantiles$new(x)$qx()

percentile_95 <- SampleQuantiles$new(x, na.rm = TRUE, digits = 2, probs = 0.95)
percentile_95$qx()

percentile_75 <- SampleQuantiles$new(x, na.rm = TRUE, digits = 3, probs = 0.75)
percentile_75$qx()

R6::is.R6(percentile_95)
```
Index

boot, 2, 3
boot.ci, 2, 3, 5, 8, 10, 12
BootCoefQuartVar, 2
BootCoefVar, 3
CoefQuartVar, 4
CoefQuartVarCI, 4
CoefVar, 6
CoefVarCI, 7
conf.limits.nct, 8, 11
cqv_versatile, 9
cv_versatile, 11

MBESS, 8, 11

names, 13

quantile, 13

SampleQuantiles, 13