Package ‘QQreflimits’

September 1, 2025

Version 1.0.3

Title Reference Limits using QQ Methodology

Author Douglas M. Hawkins [aut],
Jessica J. Kraker [aut, cre]

Maintainer Jessica J. Kraker <krakerjj@uwec.edu>

Depends R (>= 3.6.0)

Suggests testthat (>= 3.0.0)

Imports stats

Description A collection of routines for finding reference limits using, where appropriate, QQ methodology. All use a data vector X of cases from the reference population. The default is to get the central 95% reference range of the population, namely the 2.5 and 97.5 percentile, with optional adjustment of the range. Along with the reference limits, we want confidence intervals which, for historical reasons, are typically at 90% confidence. A full analysis provides six numbers:
– the upper and the lower reference limits, and
- each of their confidence intervals.
For application details, see Hawkins and Esquivel (2024) <doi:10.1093/jalm/jfad109>.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2025-09-01 16:50:07 UTC

Contents

`BCr_Pval` *P value testing for normality of (transformed) data*

Description

Computes the P value of the quantile-quantile correlation coefficient (QQCC).

Usage

```
BCr_Pval(correl, n, censor=0, winsor=0, isBC=FALSE, is2pBC=FALSE)
```

Arguments

`correl`	the QQ correlation coefficient
`n`	the sample size
`censor`	*optional* (default of 0) - the number of readings censored on the left
`winsor`	*optional* (default of 0) - the number of readings winsorized in each tail
`isBC`	*optional* (default of FALSE) - if TRUE, the QQCC is after Box-Cox transformation
`is2pBC`	*optional* (default of FALSE) - if TRUE, the QQCC is after a shifted Box-Cox transformation.

Details

Lower-level function, called by other functions in package. It takes information from a quantile-quantile regression, along with the circumstances leading up to it, to produce a P value testing for normality.

Value

`Pval`	the P value

Author(s)

Douglas M. Hawkins, Jessica J. Kraker krakerjj@uwec.edu

References

Hawkins DM, Esquivel RN (2024). A Quantile-Quantile Toolbox for Reference Intervals. *The Journal of Applied Laboratory Medicine*, **9:2**, 357-370.

Examples

```
# compute the Pvalue for two QQCC's
BCr_Pval(c(0.993, 0.99), 120)
# if censored
BCr_Pval(c(0.993, 0.99), 120, censor=3)
# if winsorized
BCr_Pval(c(0.993, 0.99), 120, winsor=3)
# on Box-Cox transformed data
BCr_Pval(c(0.993, 0.99), 120, isBC=TRUE)
# on Box-Cox transformed data, and winsorized
BCr_Pval(c(0.993, 0.99), 120, isBC=TRUE, winsor=3)
```

`BC_limits` *Limits for Box-Cox Transformed Data*

Description

Box Cox transformation of data to normality; reference limits are found using golden section search.

Usage

```
BC_limits(X, perc=0.95, cover=0.9, censor = 0, winsor=0,
          bottom=-3, top=3, epsilon=0.0001, neff=NA, CI_corrfac=NA,
          printem=FALSE)
```

Arguments

`X` the numeric data vector to be transformed.

`perc` *optional* (default of 0.95) - the two-sided coverage of the reference range computed.

`cover` *optional* (default of 0.9) - the confidence level of the CI computed for the reference limits.

`censor` *optional* (default of 0) - the number of left-censored readings

`winsor` *optional* (default of 0) - the number winsorised in each tail

`bottom` *optional* (default of -3) - the smallest Box-Cox power to be considered.

`top` *optional* (default of 3) - the largest Box-Cox power to be considered.

`epsilon` *optional* (default of 0.0001) - a tolerance limit for convergence.

`neff` *optional* (default of NA) - effective sample size, computed by the code but can be overridden.

`CI_corrfac` *optional* (default of NA) - correction factor for CIs, computed by code but can be overridden.

`printem` *optional* - if TRUE, routine will print out results as a `message`.

Details

The function fits the Box-Cox transformation by finding the exponent that maximizes the QQ correlation coefficient. Having done so, it

- calculates the reference limits and confidence intervals of the transformed data and
- then transforms them back to the original scale. The QQ analysis may incorporate censoring or winsorizing if appropriate.

Value

A list containing the following components:

`bestr`	the maximized QQ correlation coefficient.
`bestpow`	the fitted Box Cox power.
`bestxform`	the fitted Box Cox transform of the data.
`lower`	the lower reference limit and CI on the original scale.
`upper`	the upper reference limit and CI on the original scale.
`BClower`	the lower reference limits and CI on the transformed scale.
`BCupper`	the upper reference limits and CI on the transformed scale.
`meanof`	the mean of the Box-Cox transform.
`sdof`	the sd of the Box-Cox transform.
`intercept`	the intercept of the fitted QQ regression.
`slope`	the slope of the fitted QQ regression.
`Pval`	the P value of the QQ correlation.

Author(s)

Douglas M. Hawkins, Jessica J. Kraker krakerjj@uwec.edu

References

Hawkins DM, Esquivel RN (2024). A Quantile-Quantile Toolbox for Reference Intervals. *The Journal of Applied Laboratory Medicine*, **9:2**, 357-370.

Examples

```
# parameters
mul    <- 3.6
sigmal <- 0.75

# replicable randomization
set.seed(1069)
X      <- exp(mul + sigmal*rnorm(120))

# evaluate and review
BC_results <- BC_limits(X, printem=TRUE)
BC_results$bestpow
```

```
BC_results$bestr
# original-scale limits
BC_results$lower[1]; BC_results$upper[1]
cat("\nWith 90% [default] confidence, the lower limit is between",
    signif(BC_results$lower[2],5), "and", signif(BC_results$lower[3],5),
    ";\n while the upper limit is between",
    signif(BC_results$upper[2],5),"and",signif(BC_results$upper[3],5),".\n\n")

# adjust to have heavy tails
HT     <- X
HT[c(1,2,3,4)] <- HT[c(1,2,3,4)] * c(0.5, 0.5, 2, 2)

# evaluate and review
BC_HT_results <- BC_limits(HT)
BC_HT_results$lower; BC_HT_results$upper
# winsorized
BC_HT_wins_results <- BC_limits(HT, winsor=3)
BC_HT_wins_results$lower; BC_HT_wins_results$upper
```

nonp_limits *Nonparametric Limits*

Description

Calculate shortest nonparametric reference limits, at given confidence level.

Usage

```
nonp_limits(X, RR=TRUE, perc=0.95, cover=0.9)
```

Arguments

X the data set whose reference limit is sought.

RR *optional* (default of TRUE) - the way percentiles are defined:

- FALSE – use the Weibull limit i/(n+1),
- TRUE – use the Hazen limit (i-0.5)/n.

perc *optional* (default of 0.95) - the two-sided probability.

cover *optional* (default of 0.9) - the confidence level of the CI for the reference limit.

Details

The reference limits are estimated as lower and upper percentiles of the sample. There are many ways of defining sample percentiles of which the code has two options – the Hazen limit (preferred) and the Weibull limit (which has historical precedents).

The confidence limits on these percentiles come from standard binomial methodology.

- Each interval is a pair of order statistics from the sample.

- The codes find the order statistics with the required 90% coverage but use order statistics with indices as close together as possible.
- An implication of this is that the 90% confidence interval on each reference limit has a total tail area less than 10% but does not necessarily have less than 5% in each tail.

If the sample size is too small (<91 for the default settings), confidence limits can not be computed and will be returned as NA. If it is even smaller (<21 for the default settings) the estimated reference limits will also be returned as NA.

Value

A list containing the following components:

lower	the lower limit and confidence interval
upper	the upper limit and confidence interval
a	the index of the order statistic defining the lower limit of the CI
b	the index of the order statistic defining the upper limit of the CI
coverage	the actual confidence level the interval achieves

Author(s)

Douglas M. Hawkins, Jessica J. Kraker krakerjj@uwec.edu

References

Horn PS, Peske AJ (2005). Reference intervals: a user's guide. Washington (DC): AACC Press.

Examples

```
# parameters
mu    <- 40
sigma <- 10
n     <- 120

# replicable randomization
set.seed(1069)
X     <- mu + sigma*rnorm(n)

# evaluate and review
perc_used = 0.95
nonp_results <- nonp_limits(X, perc=perc_used)
cat("\nThe reference limits are the", 100*(1-perc_used)/2, "th percentile",
    signif(nonp_results$lower[1],5), "and the", 100*(1+perc_used)/2, "th percentile",
    signif(nonp_results$upper[1],5), ".\n")

cat("\nAnd with ", round(100*nonp_results$coverage,2),"% confidence, the lower limit is between",
    signif(nonp_results$lower[2],5), "and", signif(nonp_results$lower[3],5),
    ";\n while the upper limit is between",
    signif(nonp_results$upper[2],5),"and",signif(nonp_results$upper[3],5),".\n\n")
```

para_limits *Parametric Limits, based on Normal*

Description

Parametric confidence limit for a normal or normal-in-the-middle sample.

Usage

```
para_limits(mean, sd, N, censor=0, winsor=0, perc=0.95, cover=0.9)
```

Arguments

mean	the mean of a complete normal sample, or the intercept of the QQ regression of a censored or winsorized sample.
sd	the sd of a complete normal sample, or the slope of the QQ regresion of a censorised or winsorized sample.
N	the sample size
censor	*optional* (default of 0) - the number of left-censored readings
winsor	*optional* (default of 0) - the number winsorized in each tail
perc	*optional* (default of 0.95) - the two-sided coverage sought
cover	*optional* (default of 0.9) - the confidence level of the CI on the reference limit

Details

This function computes two-sided reference limits and their confidence intervals for data that are normal; normal across the reference interval; or censored normal. The reference limits are conventional mean + z*se, and their confidence intervals come from the delta method.

Value

A list containing the following components:

lower	the lower reference limit and its CI
upper	the upper reference limit and its CI
effn	the effective sample size with censoring or winsorization

Author(s)

Douglas M. Hawkins, Jessica J. Kraker krakerjj@uwec.edu

References

Horn PS, Peske AJ (2005). Reference intervals: a user's guide. Washington (DC): AACC Press.

Examples

```
# parameters
mu    <- 40
sigma <- 10
n     <- 120
# identifying winsoring
wins  <- trunc(n/40)
# replicable randomization
set.seed(1069)
X     <- mu + sigma*rnorm(n)
# replicable randomization with heavy tails
set.seed(1069)
HT    <- mu + sigma * rt(n, 5)

# retain original visual settings
oldsettings <- par(mfrow=par()$mfrow, oma=par()$oma)
# visual settings
par(mfrow=c(2,2))
par(oma=c(0,0,2,0))
# plot to compare
base <- QQnorm(X, main="Base normal", showsum=TRUE)
title("Illustrating QQnorm with para_limits", outer=TRUE)
basew <- QQnorm(X, main="Winsorized", winsor=wins, showsum=TRUE)
ht   <- QQnorm(HT, main="Heavy tail", showsum=TRUE)
htw  <- QQnorm(HT, main="Winsorized", winsor=wins, showsum=TRUE)
# restore
par(oldsettings)

# evaluate and review
norm_results <- para_limits(mean(X), sd(X), n)
norm_results
# evaluate and review with tails
tailed_results <- para_limits(htw$intercept, htw$slope, n, winsor=wins)
tailed_results
```

QQnorm *QQ Plot with Correlation*

Description

Creates QQ plot of complete or censored data, with summaries and test.

Usage

```
QQnorm(X, main="", ylab="", censor=0, winsor=0,
       joinem=FALSE, ylim=c(NA,NA), isBC=FALSE, is2pBC=FALSE,
       doplot=TRUE, showP=TRUE, fitline=TRUE, showsum=FALSE)
```

Arguments

`X` the numeric data vector to be plotted. Censored values should be reported as the censoring value.

`main` the header text.

`ylab` the name of the variable.

`censor` *optional* (default of 0) - the number of data censored on the left.

`winsor` *optional* (default of 0) - the number of data winsored in each tail.

`joinem` *optional* (default of FALSE) - if TRUE the plot is drawn as a segmented line, if FALSE the individual points are plotted as x if winsorized, else *.

`ylim` *optional* - y limits on the plot.

`isBC` *optional* (default of FALSE) - if true, the data set is a Box-Cox transform of the original data.

`is2pBC` *optional* (default of FALSE) - if true, the data set is a shifted Box-Cox tranform.

`doplot` *optional* (default of TRUE) - if true, the QQ plot is drawn.

`showP` *optional* (default of TRUE) - if true, the QQCC P value is shown on the plot.

`fitline` *optional* (default of TRUE) - if true, the QQ regression line is plotted.

`showsum` *optional* (default of FALSE) - if true, the intercept and slope of the QQ regression are shown.

Details

Makes QQ plot of complete or censored data, as input by user:

- fits regression line to complete uncensored portion of data,
- calculates the QQ correlation coefficient of the fitted line, and
- reports the P value of the QQ correlation coefficient as calculated by `BCR_pval`. Parameters `isBC` and `is2pBC` are relevant to the P value calculation.

Value

A list containing the following components:

`correl` the QQ correlation coefficient.

`Pval` the P value of the QQ correlation coefficient.

`mean` the mean of the data.

`sd` the sd of the data.

`intercept` the intercept of the QQ regression line, used in place of the mean when there is censoring or winsorization.

`slope` the slope of the QQ regression line, used in place of the sd when there is censorin or winsorization.

Author(s)

Douglas M. Hawkins, Jessica J. Kraker krakerjj@uwec.edu

References

Hawkins DM, Esquivel RN (2024). A Quantile-Quantile Toolbox for Reference Intervals. *The Journal of Applied Laboratory Medicine*, **9:2**, 357-370.

Examples

```
# parameters
mu    <- 40
sigma <- 10
n     <- 120
# identifying winsoring
wins  <- trunc(n/40)
# replicable randomization
set.seed(1069)
X     <- mu + sigma*rnorm(n)
# replicable randomization with heavy tails
set.seed(1069)
HT    <- mu + sigma * rt(n, 5)

# retain original visual settings
oldsettings <- par(mfrow=par()$mfrow, oma=par()$oma)
# visual settings
par(mfrow=c(2,2))
par(oma=c(0,0,2,0))
# plot to compare
base <- QQnorm(X, main="Base normal", showsum=TRUE)
title("Illustrating QQnorm with para_limits", outer=TRUE)
basew <- QQnorm(X, main="Winsorized", winsor=wins, showsum=TRUE)
ht   <- QQnorm(HT, main="Heavy tail", showsum=TRUE)
htw  <- QQnorm(HT, main="Winsorized", winsor=wins, showsum=TRUE)
# restore
par(oldsettings)
```

Index