Package: referenceIntervals (via r-universe)

October 31, 2026

Type Package

Title Reference Intervals

Version 1.3.1

Date 2024-03-30

Author Daniel Finnegan

Maintainer Daniel Finnegan <dan.finnegan@gmail.com>

Imports boot, extremevalues, MASS, outliers

LazyData no

Description This is a collection of tools to allow the medical professional to calculate appropriate reference ranges (intervals) with confidence intervals around the limits for diagnostic purposes.

License GPL-3

NeedsCompilation no

Repository https://digital-vet.r-universe.dev

Date/Publication 2024-03-31 02:33:33 UTC

RemoteUrl https://github.com/cran/referenceIntervals

RemoteRef HEAD

RemoteSha 2e98d04abf80b066cbf397a4ccf9a6bcc98bb92b

Contents

referenceIntervals-package 2
cook.outliers 3
dixon.outliers 4
horn.outliers 5
nonparRanks 6
nonparRI 7
plot.interval 8
print.interval 8
print.interval.sub 9

refLimit 10
robust 12
set120 13
set20 13
set200 14
set50 14
singleRefLimit 15
vanderLoo.outliers 16

Index **18**

`referenceIntervals-package`

This package calculates reference intervals from a dataset using either parametric, non-parametric, or robust methods.

Description

This package also calculates the confidence intervals around the calculated reference intervals in order to provide a metric for how precise the calculations are. This package also contains four outlier detection functions.

Details

Package:	referenceIntervals
Type:	Package
Version:	1.3.1
Date:	2024-03-30
License:	GPL-3

Author(s)

Daniel Finnegan

Maintainer: Daniel Finnegan <dan.finnegan@gmail.com>

References

Clinical and Laboratory Standards Institute. Defining, Establishing, and Verifying Reference Intervals in the Clinical Laboratory; Approved Guideline - Third Edition. C28-A3c. 28(30).

Friedman JH. A recursive partitioning decision rule for nonparametric classification. *IEEE Transactions on Computers.* 1977. April: 404-408.

Friedrichs KR, Harr KE, Freeman KP, et al. ASVCP reference interval guidelines: determination of de novo reference intervals in veterinary species and other related topics. *Vet Clin Pathol.* 2012;441-453.

H.E. Solberg Department of Clinical Chemistry. Approved recommendation (1987) on the theory of reference values. Part 5. Statistical treatment of collected reference values. Determination of reference limits. *Clinica Chimica Acta.* 1987:S13-S32.

Horn PS, Pesce AJ. Reference intervals: an update. *Clinica Chimica Acta.* 2003. 334:5-23.

Horn PS, Pesce AJ, Copeland BE. A robust approach to reference interval estimation and evaluation. *Clinical Chemistry.* 1998. 44(3):622-631.

Horn PS, Pesce AJ, Copeland BE. Reference interval computation using robust vs parametric and nonparametric analyses. *Clinical Chemistry.* 1999. 45(12):2284-2285.

Linnet K. Nonparametric estimation of reference intervals by simple and bootstrap-based procedures. *Clinical Chemistry.* 2000. 46(6):867-869.

Verma SP, Quiroz-Ruiz A. Critical values for six Dixon tests for outliers in normal samples up to sizes 100, and applications in science and engineering. *Revista Mexicana de Ciencias Geologicas.* 2006. 23(2):133-161.

Verma SP, Quiroz-Ruiz A, Diaz-Gonzalez L. Critical values for 33 discordancy test variants for outliers in normal samples up to sizes 1000, and applications in quality control in Earth Sciences. *Revista Mexicana de Ciencias Geologicas.* 2008. 25(1):82-96.

Virtanen A, Kairisto V, Uusipaikka E. Regression-based reference limits: determination of sufficient sample size. *Clinical Chemistry.* 1998. 44(11):2353-2358.

Examples

```
refLimit(set50, out.rm = TRUE, out.method = "cook")
refLimit(set200, out.method = "horn", RI = "n", refConf = 0.90, limitConf = 0.80)
horn.outliers(set120)
dixon.outliers(set20)

refLimit(set50, out.method = "vanderLoo", out.rm = TRUE, RI = "r", bootStat = "perc")

frame = data.frame(one = rnorm(30, m = 5, sd = 2), two = rnorm(30, m = 7, sd = 1),
 three = rnorm(30, m = 2, sd = 0.5))
result = refLimit(frame)
plot(result)
```

`cook.outliers` *Determines outliers using Cook's Distance*

Description

A linear regression model is calculated for the data (which is the mean for one-dimensional data. From that, using the Cook Distances of each data point, outliers are determined and returned.

Usage

```
cook.outliers(data)
```

Arguments

data	A vector of data points.

Value

Returns a list containing a vector of outliers and a vector of the cleaned data (subset).

outliers	A vector of outliers from the data set
subset	A vector containing the remaining data, cleaned of outliers

Author(s)

Daniel Finnegan

Examples

```
cook.outliers(set50)
plot(cook.outliers(set50)$subset)
```

---

dixon.outliers *Determines outliers using Dixon's Q Test method*

Description

This determines outliers of the dataset by calculating Dixon's Q statistic and comparing it to a standardized table of statistics. This method can only determine outliers for datasets of size 3 <= n <= 30. This function requires the outliers package.

Usage

```
dixon.outliers(data)
```

Arguments

data	A vector of data points.

Value

Returns a list containing a vector of outliers and a vector of the cleaned data (subset).

outliers	A vector of outliers from the data set
subset	A vector containing the remaining data, cleaned of outliers

Author(s)

Daniel Finnegan

References

Statistical treatment for rejection of deviant values: critical values of Dixon's "Q" parameter and related subrange ratios at the 95 (2), pp 139-146 DOI: 10.1021/ac00002a010. Publication Date: January 1991

One-sided and Two-sided Critical Values for Dixon's Outlier Test for Sample Sizes up to n = 30. Economic Quality Control, Vol 23(2008), No. 1, 5-13.

Examples

```
dixon.outliers(set20)
summary(dixon.outliers(set20)$subset)
```

---

horn.outliers *Determines outliers using Horn's method and Tukey's interquartile fences on a Box-Cox transformation of the data.*

Description

This function determines outliers in a Box-Cox transformed dataset using Horn's method of outlier detection using Tukey's interquartile fences. If a data point lies outside 1.5 * IQR from the 1st or 3rd quartile point, it is an outlier.

Usage

```
horn.outliers(data)
```

Arguments

`data`	A vector of data points.

Value

Returns a list containing a vector of outliers and a vector of the cleaned data (subset).

`outliers`	A vector of outliers from the data set
`subset`	A vector containing the remaining data, cleaned of outliers

Author(s)

Daniel Finnegan

References

ASVCP reference interval guidelines: determination of de novo reference intervals in veterinary species and other related topics. Vet Clin Pathol 41/4 (2012) 441-453, 2012 American Society for Veterinary Clinical Pathology

Horn, P. S., Feng, L., Li, Y., & Pesce, A. J. (2001). Effect of outliers and nonhealthy individuals on reference interval estimation. Clinical Chemistry, 47(12), 2137-2145.

Horn, P.S., Pesce, A.J. (2003). Reference Intervals: an update. Clin Chim Acta. 334(1-2):5-23. DOI: doi: 10.1016/s0009-8981(03)00133-5.

Examples

```
horn.outliers(set200)
```

`nonparRanks` *Table that dictate the ranks for the confidence intervals around the calculated reference interval.*

Description

This is a table that dictate the ranks for the confidence intervals around the calculated reference interval. This method is available when 120 <= n <= 1000.

Usage

```
nonparRanks
```

Format

A data frame with 882 observations on the following 3 variables.

`SampleSize` a numeric vector

`Lower` a numeric vector

`Upper` a numeric vector

References

Defining, Establishing, and Verifying Reference Intervals in the Clinical Laboratory; Approved Guideline - 3rd Edition (C28-A3)

Examples

```
data(nonparRanks)
```

nonparRI *Determines the reference interval using non-parametric means*

Description

This function uses the appropriate percentiles as determined by refConf to return the non-parametric reference interval. This is written as a boot function to use within the function refLimit.

Usage

```
nonparRI(data, indices = 1:length(data), refConf = 0.95)
```

Arguments

data	data is a vector of sample values.
indices	The indices of data to be used in the calculations. The default is to use the whole set.
refConf	refConf is a measure of the range covered by the calculation. Most often, as is the default, 95

Value

Returns a vector of two values, the lower and upper limits of the reference interval.

Author(s)

Daniel Finnegan

References

Defining, Establishing, and Verifying Reference Intervals in the Clinical Laboratory; Approved Guideline - 3rd Edition (C28-A3)

Examples

```
nonparRI(set50)
nonparRI(set50, refConf = 0.9)
```

plot.interval — *Overload of plot function to include the ability to plot the results of refLimit*

Description

Plots the reference interval and confidence intervals around the limits of the reference interval.

Usage

```
## S3 method for class 'interval'
plot(x, main, ...)
```

Arguments

x	Object x is of type "interval".
main	Title of plot.
...	Arguments to be passed to methods, such as graphical parameters (see 'par').

Value

No return value.

Author(s)

Daniel Finnegan

Examples

```
result = refLimit(set200)
plot(result)
```

print.interval — *Overload of print in order to concisely print the results of refLimit*

Description

This function allows for the pretty-printing of a large list object created by calling the refLimit function.

Usage

```
## S3 method for class 'interval'
print(x, digits = 4L, quote = TRUE, prefix = "", ...)
```

Arguments

x	x is an object of type "interval"
digits	minimal number of _significant_ digits. See 'print.default'.
quote	logical, indicating whether or not strings should be printed with surrounding quotes.
prefix	Option to specify a formatting prefix.
...	further arguments passed to or from other methods.

Value

No return value.

Author(s)

Daniel Finnegan

Examples

```
result = refLimit(set120)
result
```

---

print.interval.sub *Overload of print in order to concisely print the results of refLimit*

Description

This function allows for the pretty-printing of a large list object created by calling the refLimit function.

Usage

```
## S3 method for class 'interval.sub'
print(x, digits = 4L, quote = TRUE, prefix = "", ...)
```

Arguments

x	x is an object of type "interval"
digits	minimal number of _significant_ digits. See 'print.default'.
quote	logical, indicating whether or not strings should be printed with surrounding quotes.
prefix	Option to specify a formatting prefix.
...	further arguments passed to or from other methods.

Value

No return value.

Author(s)

Daniel Finnegan

`refLimit` *Calculates and returns reference and confidence intervals for a dataset*

Description

This function calculates a reference interval from a dataset using parametric, non-parametric, or robust methods.

Usage

```
refLimit(data, out.method = "horn", out.rm = FALSE, RI = "p", CI = "p",
refConf = 0.95, limitConf = 0.9, bootStat = "basic")
```

Arguments

`data`	A vector of data points.
`out.method`	The outlier detection method. Valid options include "horn", "cook", "dixon", and "vanderLoo".
`out.rm`	Remove outliers. If value is TRUE, outliers will be automatically removed prior to calculations. If FALSE (default), outliers will be detected but not removed.
`RI`	Method for reference interval calculations. Valid options include "p" (default) for parametric, "n" for non-parametric, and "r" for robust method.
`CI`	Method for confidence interval calculations. Valid options include "p" for parametric (default), "n" for non-parametric, and "boot" for bootstrapping method. The minimum samplesize for non-parametric confidence interval calculations is 120. With smaller samples, bootstrapping methods are used.
`refConf`	Desired coverage for the calculated reference interval. The default is a 95 interval.
`limitConf`	Desired confidence interval level. The default is a 90 reference interval limits.
`bootStat`	Method for calculating confidence intervals from package *boot*. Valid arguments include "basic" (basic bootstrap method), "perc" (bootstrap percentile method), "norm" (normal approximation method), "stud" (studentized bootstrap method), and "bca" (adjusted bootstrap percentile method).

Details

A confidence interval around each limit of the reference interval is calculated as a metric for determining the validity of the result. Outliers can be detected in one of four different methods and automatically eliminated.

To determine the most appropriate calculation for confidence intervals using the bootstrapping method, please consult chapter 5 of Davison and Hinkley's "Bootstrap Methods and their Applications."

Value

Returns a list of necessary information.

`size`	Size of dataset
`dname`	Name of dataset
`out.method`	Outlier detection method
`out.rm`	Boolean indicating whether outliers are automatically removed
`outliers`	Vector of detected outliers
`methodRI`	Method for reference interval calculations (p, n, or r)
`methodCI`	Method for confidence interval calculations (p, n, boot)
`norm`	Results of running Shapiro-Wilk and Kolmorgorov-Smirnov normacy tests
`refConf`	Desired coverage of reference interval
`limitConf`	Desired confidence interval level
`Ref_Int`	List containing the reference interval and confidence interval values

Author(s)

Daniel Finnegan

References

ASVCP reference interval guidelines: determination of de novo reference intervals in veterinary species and other related topics. *Vet Clin Pathol* 41/4 (2012) 441-453, 2012. American Society for Veterinary Clinical Pathology

Davison, A.C. and Hinkley, D.V. (1997) *Bootstrap Methods and Their Application*, Chapter 5. Cambridge University Press.

Examples

```
refLimit(set20, out.method = "dixon")
refLimit(set200, out.method = "cook", out.rm = TRUE, RI = "n", refConf = 0.9)
refLimit(set50, out.method = "vanderLoo", out.rm = TRUE, RI = "r", bootStat = "perc")
```

`robust` *Algorithm that implements the robust method for reference interval calculations*

Description

The robust method is an iterative method that determines the most appropriate weighted mean of the data and then calculates the desired reference interval.

Usage

```
robust(data, indices = c(1:length(data)), refConf = 0.95)
```

Arguments

`data`	Vector of data.
`indices`	Indices of data to use for calculations.
`refConf`	Desired coverage of the reference interval. Default is 95 interval.

Value

Returns a vector containing the lower and upper limits of the reference interval.

Author(s)

Daniel Finnegan

References

Defining, Establishing, and Verifying Reference Intervals in the Clinical Laboratory; Approved Guideline - 3rd Edition (C28-A3)

Examples

```
robust(set50)
robust(horn.outliers(set20)$subset)
```

set120 *Dataset containing 120 values*

Description

Small dataset containing 120 samples. The mean is centered on 27 with a standard deviation of 7.

Usage

```
set120
```

Format

The format is: num [1:120] 38.1 12.6 31.3 35.5 22.6 ...

Source

```
rnorm(120, m = 27, sd = 7)
```

Examples

```
data(set120)
```

set20 *Small dataset containing 20 samples*

Description

Small dataset containing 20 samples. The mean is centered on 42 with a standard deviation of 5.

Usage

```
set20
```

Format

The format is: num [1:20] 35 32.9 43.6 44.6 35.9 ...

Source

```
rnorm(20, m = 42, sd = 6)
```

Examples

```
data(set20)
```

set200 *Dataset containing 200 values*

Description

Small dataset containing 200 samples. The mean is centered on 5 with a standard deviation of 1.

Usage

```
set200
```

Format

The format is: num [1:200] 3.95 5.16 5.32 3.86 3.54 ...

Source

```
rnorm(200, m = 5, sd = 1)
```

Examples

```
data(set200)
```

set50 *Dataset containing 50 values*

Description

Small dataset containing 50 samples. The mean is centered on 14 with a standard deviation of 3.

Usage

```
set50
```

Format

The format is: num [1:50] 16.61 20.43 7.91 15.19 14.77 ...

Source

```
rnorm(50, m = 14, sd = 3)
```

Examples

```
data(set50)
```

singleRefLimit *This is the workhorse of the refLimit function*

Description

This is the function called to work on each individual vector of data.

Usage

```
singleRefLimit(data, dname = "default", out.method = "horn", out.rm = FALSE,
RI = "p", CI = "p", refConf = 0.95, limitConf = 0.9, bootStat = "basic")
```

Arguments

data	A vector of data points.
dname	Name of dataset.
out.method	The outlier detection method. Valid options include "horn", "cook", "dixon", and "vanderLoo".
out.rm	Remove outliers. If value is TRUE, outliers will be automatically removed prior to calculations. If FALSE (default), outliers will be detected but not removed.
RI	Method for reference interval calculations. Valid options include "p" (default) for parametric, "n" for non-parametric, and "r" for robust method.
CI	Method for confidence interval calculations. Valid options include "p" for parametric (default), "n" for non-parametric, and "boot" for bootstrapping method. The minimum sample size for non-parametric confidence interval calculations is 120. With smaller samples, bootstrapping methods are used.
refConf	Desired coverage for the calculated reference interval. The default is a 95
limitConf	Desired confidence interval level. The default is a 90 confidence interval around the reference interval limits.
bootStat	Method for calculating confidence intervals from package boot. Valid arguments include "basic" (basic bootstrap method), "perc" (bootstrap percentile method), "norm" (normal approximation method), "stud" (studentized bootstrap method), and "bca" (adjusted bootstrap percentile method).

Value

Returns a list of necessary information.

size	Size of dataset
dname	Name of dataset
out.method	Method of outlier detection
out.rm	Boolean indicating whether outliers are automatically removed
outliers	Vector of detected outliers

methodRI	Method for reference interval calculations (p, n, or r)
methodCI	Method for confidence interval calculations (p, n, boot)
norm	Results of running Shapiro-Wilk and Kolmorgorov-Smirnov normacy tests
refConf	Desired coverage of reference interval
limitConf	Desired confidence interval level
Ref_Int	List containing the reference interval and confidence interval values

Author(s)

Daniel Finnegan

Examples

```
singleRefLimit(set200, out.method = "horn", out.rm = TRUE)
```

---

vanderLoo.outliers *Mark van der Loo's outlier detection method in the extremevalues package*

Description

Separates data into vectors of outliers and a cleaned subset of the data.

Usage

```
vanderLoo.outliers(data)
```

Arguments

data	Vector of data values.

Value

Returns a list containing a vector of outliers and a vector of the cleaned data (subset).

outliers	A vector of outliers from the data set
subset	A vector containing the remaining data, cleaned of outliers

Note

Requires extremevalues package.

Author(s)

Daniel Finnegan

References

http://cran.r-project.org/web/packages/extremevalues/extremevalues.pdf

Examples

```
vanderLoo.outliers(set50)
vanderLoo.outliers(set200)
```

Index

∗ **~Cook**
    cook.outliers, 3
∗ **~Dixon**
    dixon.outliers, 4
∗ **~Horn**
    horn.outliers, 5
∗ **~interval**
    nonparRI, 7
    print.interval, 8
    refLimit, 10
    robust, 12
    singleRefLimit, 15
∗ **~nonparametric**
    nonparRI, 7
∗ **~outliers**
    vanderLoo.outliers, 16
∗ **~outlier**
    cook.outliers, 3
    dixon.outliers, 4
    horn.outliers, 5
∗ **~print**
    print.interval, 8
∗ **~range**
    nonparRI, 7
∗ **~reference**
    refLimit, 10
    singleRefLimit, 15
∗ **~robust**
    robust, 12
∗ **~vanderLoo**
    vanderLoo.outliers, 16
∗ **datasets**
    nonparRanks, 6
    set120, 13
    set20, 13
    set200, 14
    set50, 14
∗ **methods**
    plot.interval, 8
∗ **package**
    referenceIntervals-package, 2
∗ **print**
    print.interval.sub, 9

cook.outliers, 3

dixon.outliers, 4

horn.outliers, 5

nonparRanks, 6
nonparRI, 7

plot.interval, 8
print.interval, 8
print.interval.sub, 9

referenceIntervals
        *(*referenceIntervals-package*)*, 2
referenceIntervals-package, 2
refLimit, 10
robust, 12

set120, 13
set20, 13
set200, 14
set50, 14
singleRefLimit, 15

vanderLoo.outliers, 16