Package: simpleboot (via r-universe)

October 20, 2026

Version 1.1-8

Depends R (>= 3.0.0)

Imports boot, stats, graphics

Title Simple Bootstrap Routines

Author Roger D. Peng <roger.peng@austin.utexas.edu>

Maintainer Roger D. Peng <roger.peng@austin.utexas.edu>

Description Simple bootstrap routines.

License GPL (>= 2)

URL https://github.com/rdpeng/simpleboot

Repository https://rdpeng.r-universe.dev

Date/Publication 2024-06-27 16:11:40 UTC

RemoteUrl https://github.com/rdpeng/simpleboot

RemoteRef HEAD

RemoteSha 9088ce77864d6ba9721e26d1af13e89aba021299

Contents

hist.simpleboot 2
lm.boot 3
lm.simpleboot.methods 4
loess.boot 5
loess.simpleboot.methods 7
one.boot 8
pairs_boot 9
perc 10
plot.lm.simpleboot 11
plot.loess.simpleboot 12
samples 13
two.boot 14

Index **16**

`hist.simpleboot` *Histograms for bootstrap sampling distributions.*

Description

Construct a histogram of the bootstrap distribution of univariate statistic.

Usage

```
## S3 method for class 'simpleboot'
hist(x, do.rug = FALSE, xlab = "Bootstrap samples", main = "", ...)
```

Arguments

`x`	An object of class `"simpleboot"` returned from either `one.boot`, `two.boot`, or `pairs.boot`.
`do.rug`	Should a rug of the bootstrap distribution be plotted under the histogram?
`xlab`	The label for the x-axis.
`main`	The title for the histogram.
`...`	Other arguments passed to `hist`.

Details

`hist` constructs a histogram for the bootstrap distribution of a univariate statistic. It cannot be used with linear model or loess bootstraps. In the histogram a red dotted line is plotted denoting the observed value of the statistic.

Value

Nothing is returned.

Author(s)

Roger D. Peng

Examples

```
x <- rnorm(100)

## Bootstrap the 75th percentile
b <- one.boot(x, quantile, R = 1000, probs = 0.75)
hist(b)
```

`lm.boot` *Linear model bootstrap.*

Description

Bootstrapping of linear model fits (using `lm`). Bootstrapping can be done by either resampling rows of the original data frame or resampling residuals from the original model fit.

Usage

```
lm.boot(lm.object, R, rows = TRUE, new.xpts = NULL, ngrid = 100,
        weights = NULL)
```

Arguments

`lm.object`	A linear model fit, produced by `lm`.
`R`	The number of bootstrap replicates to use.
`rows`	Should we resample rows? Setting `rows` to `FALSE` indicates resampling of residuals.
`new.xpts`	Values at which you wish to make new predictions. If specified, fitted values from each bootstrap sample will be stored.
`ngrid`	If `new.xpts` is `NULL` and the regression is 2 dimensional, then predictions are made on an evenly spaced grid (containing `ngrid` points) spanning the range of the predictor values.
`weights`	Reseampling weights; a vector of length equal to the number of observations.

Details

Currently, `"lm.simpleboot"` objects have a simple `print` method (which shows the original fit), a `summary` method and a `plot` method.

Value

An object of class `"lm.simpleboot"` (which is a list) containing the elements:

`method`	Which method of bootstrapping was used (rows or residuals).
`boot.list`	A list containing values from each of the bootstrap samples. Currently, bootstrapped values are model coefficients, residual sum of squares, R-square, and fitted values for predictions.
`orig.lm`	The original model fit.
`new.xpts`	The locations where predictions were made.
`weights`	The resampling weights. If none were used, this component is `NULL`

Author(s)

Roger D. Peng

See Also

The `plot.lm.simpleboot` method.

Examples

```
data(airquality)
attach(airquality)
set.seed(30)
lmodel <- lm(Ozone ~ Wind)
lboot <- lm.boot(lmodel, R = 1000)
summary(lboot)

## With weighting
w <- runif(nrow(model.frame(lmodel)))
lbootw <- lm.boot(lmodel, R = 1000, weights = w)
summary(lbootw)

## Resample residuals
lboot2 <- lm.boot(lmodel, R = 1000, rows = FALSE)
summary(lboot2)
```

`lm.simpleboot.methods` *Methods for linear model bootstrap.*

Description

Methods for "`lm.simpleboot`" class objects.

Usage

```
## S3 method for class 'lm.simpleboot'
summary(object, ...)
## S3 method for class 'summary.lm.simpleboot'
print(x, ...)
## S3 method for class 'lm.simpleboot'
fitted(object, ...)
```

Arguments

`object`	An object of class "`lm.simpleboot`", as returned by `lm.boot`.
`x`	An object of class "`summary.lm.simpleboot`", a result of a call to `summary.lm.simpleboot`.
`...`	Other arguments passed to and from other methods.

Details

print is essentially the same as the usual printing of a linear model fit, except the bootstrap standard errors are printed for each model coefficient.

fitted returns the fitted values from each bootstrap sample for the predictor values specified by the new.xpts in lm.boot (or from the grid if new.xpts was not specified). This is a p x R matrix where p is the number of points where prediction was desired and R is the number of bootstrap samples specified. Using fitted is the equivalent of using samples(object, name = "fitted").

Value

summary returns a list containing the original estimated coefficients and their bootstrap standard errors.

Author(s)

Roger D. Peng

See Also

lm.boot.

Examples

```
data(airquality)
attach(airquality)
lmodel <- lm(Ozone ~ Wind + Solar.R)
lboot <- lm.boot(lmodel, R = 300)
summary(lboot)
```

---

loess.boot *2-D Loess bootstrap.*

Description

Bootstrapping of loess fits produced by the loess function in the modreg package. Bootstrapping can be done by resampling rows from the original data frame or resampling residuals from the original model fit.

Usage

```
loess.boot(lo.object, R, rows = TRUE, new.xpts = NULL, ngrid = 100,
           weights = NULL)
```

Arguments

lo.object	A loess fit, produced by loess.
R	The number of bootstrap replicates.
rows	Should we resample rows? Setting rows to FALSE indicates resampling of residuals.
new.xpts	Locations where new predictions are to be made. If new.xpts is NULL, then an evenly spaced grid spanning the range of X (containing ngrid points) is used. In either case
ngrid	Number of grid points to use if new.xpts is NULL.
weights	Resampling weights; a vector with length equal to the number of observations.

Details

The user can specify locations for new predictions through new.xpts or an evenly spaced grid will be used. In either case, fitted values at each new location will be stored from each bootstrap sample. These fitted values can be retrieved using either the fitted method or the samples function.

Note that the loess function has many parameters for the user to set that can be difficult to reproduce in the bootstrap setting. Right now, the user can only specify the span argument to loess in the original fit.

Value

An object of class "loess.simpleboot" (which is a list) containing the elements:

method	Which method of bootstrapping was used (rows or residuals).
boot.list	A list containing values from each of the bootstrap samples. Currently, only residual sum of squares and fitted values are stored.
orig.loess	The original loess fit.
new.xpts	The locations where predictions were made (specified in the original call to loess.boot).

Author(s)

Roger D. Peng

Examples

```
set.seed(1234)

x <- runif(100)

## Simple sine function simulation
y <- sin(2*pi*x) + .2 * rnorm(100)
plot(x, y)  ## Sine function with noise
lo <- loess(y ~ x, span = .4)

## Bootstrap with resampling of rows
```

```
lo.b <- loess.boot(lo, R = 500)

## Plot original fit with +/- 2 std. errors
plot(lo.b)

## Plot all loess bootstrap fits
plot(lo.b, all.lines = TRUE)

## Bootstrap with resampling residuals
lo.b2 <- loess.boot(lo, R = 500, rows = FALSE)
plot(lo.b2)
```

`loess.simpleboot.methods` *Methods for loess bootstrap.*

Description

Methods for "`loess.simpleboot`" class objects.

Usage

```
## S3 method for class 'loess.simpleboot'
fitted(object, ...)
```

Arguments

`object`	An object of class "`loess.simpleboot`" as returned by the function `loess.boot`.
`...`	Other arguments passed to and from other methods.

Details

`fitted` returns a n x R matrix of fitted values where n is the number of new locations at which predictions were made and R is the number of bootstrap replications used in the original loess bootstrap. This is the equivalent of calling `samples(object, "fitted")`.

Value

Nothing is returned.

Author(s)

Roger D. Peng

one.boot *One sample bootstrap of a univariate statistic.*

Description

one.boot is used for bootstrapping a univariate statistic for one sample problems. Examples include the mean, median, etc.

Usage

```
one.boot(data, FUN, R, student = FALSE, M, weights = NULL, ...)
```

Arguments

data	The data. This should be a vector of numbers.
FUN	The statistic to be bootstrapped. This can be either a quoted string containing the name of a function or simply the function name.
R	The number of bootstrap replicates to use.
student	Should we do a studentized bootstrap? This requires a double bootstrap so it might take longer.
M	If student is set to TRUE, then M is the number of internal bootstrap replications to do.
weights	Resampling weights; a vector of length equal to the number of observations.
...	Other (named) arguments that should be passed to FUN.

Value

An object of class "simpleboot", which is almost identical to the regular "boot" object. For example, the boot.ci function can be used on this object.

Author(s)

Roger D. Peng

Examples

```
library(boot)
set.seed(20)
x <- rgamma(100, 1)
b.mean <- one.boot(x, mean, 500)
print(b.mean)
boot.ci(b.mean)  ## No studentized interval here
hist(b.mean)

## Bootstrap with weights
set.seed(10)
w <- runif(100)
```

```
bw <- one.boot(x, median, 100, weights = w)
print(bw)

## Studentized
bw.stud <- one.boot(x, median, R = 100, student = TRUE, M = 50,
                    weights = w)
boot.ci(bw.stud, type = "stud")
```

---

pairs_boot *Two sample bootstrap.*

Description

pairs.boot is used to bootstrap a statistic which operates on two samples and returns a single value. An example of such a statistic is the correlation coefficient (i.e. cor). Resampling is done pairwise, so x and y must have the same length (and be ordered correctly). One can alternatively pass a two-column matrix to x.

Usage

```
pairs_boot(x, y = NULL, FUN, R, student = FALSE, M, weights = NULL, ...)
```

Arguments

x	Either a vector of numbers representing the first sample or a two column matrix containing both samples.
y	If NULL it is assumed that x is a two-column matrix. Otherwise, y is the second sample.
FUN	The statistic to bootstrap. If x and y are separate vectors then FUN should operate on separate vectors. Similarly, if x is a matrix, then FUN should operate on two-column matrices. FUN can be either a quoted string or a function name.
R	The number of bootstrap replicates.
student	Should we do a studentized bootstrap? This requires a double bootstrap so it might take longer.
M	If student is set to TRUE, then M is the number of internal bootstrap replications to do.
weights	Resampling weights.
...	Other (named) arguments that should be passed to FUN.

Value

An object of class "simpleboot", which is almost identical to the regular "boot" object. For example, the boot.ci function can be used on this object.

Author(s)

Roger D. Peng

Examples

```
library(boot)
set.seed(1)
x <- rnorm(100)
y <- 2 * x + rnorm(100)
boot.cor <- pairs_boot(x, y, FUN = cor, R = 100)
boot.ci(boot.cor)
```

---

perc *Extract percentiles from a bootstrap sampling distribution.*

Description

perc can be used to extract percentiles from the sampling distribution of a statistic.

Usage

```
perc(boot.out, p = c(0.025, 0.975))
perc.lm(lm.boot.obj, p)
```

Arguments

boot.out	Output from either one.boot, two.boot, or pairs.boot.
p	numeric vector with values in [0, 1].
lm.boot.obj	An object of class "lm.simpleboot", returned from lm.boot.

Details

perc automatically calls perc.lm if boot.out is of the class "lm.simpleboot" so there is no need to use perc.lm separately.

Value

For bootstraps which are not linear model bootstraps, perc returns a vector of percentiles of length length(p). Linear interpolation of percentiles is done if necessary. perc.lm returns a matrix of percentiles of each of the model coefficients. For example, if there are k model coefficients, the perc.lm returns a length(p) by k matrix.

Author(s)

Roger D. Peng

Examples

```
x <- rnorm(100)
b <- one.boot(x, median, R = 1000)
perc(b, c(.90, .95, .99))
```

`plot.lm.simpleboot` *Plot method for linear model bootstraps.*

Description

Plot regression lines with bootstrap standard errors. This method only works for 2-D regression fits.

Usage

```
## S3 method for class 'lm.simpleboot'
plot(x, add = FALSE, ...)
```

Arguments

`x`	An object of class `"lm.simpleboot"` returned by `lm.boot`.
`add`	Switch indicating whether the regression line should be added to the current plot.
`...`	Additional arguments passed down to `plot`. Ignored if `add = TRUE`.

Details

This function plots the data and the original regression line fit along with +/- 2 bootstrap standard errors at locations specified by the `new.xpts` argument to `lm.boot` (or on an evenly spaced grid).

Value

Nothing is returned.

Author(s)

Roger D. Peng

Examples

```
## None right now
```

`plot.loess.simpleboot` *Plot method for loess bootstraps.*

Description

Plot loess lines with bootstrap standard errors.

Usage

```
## S3 method for class 'loess.simpleboot'
plot(x, add = FALSE, all.lines = FALSE, ...)
```

Arguments

`x`	An object of class "`loess.simpleboot`" as returned by the function `loess.boot`.
`add`	Should the loess line be plotted over the current plot?
`all.lines`	Should we plot each of the individual loess lines from the bootstrap samples?
`...`	Other arguments passed to `plot`.

Details

`plot` constructs (and plots) the original loess fit and +/- 2 bootstrap standard errors at locations specified in the `new.xpts` in `loess.boot` (or on an evenly spaced grid).

Value

Nothing is returned.

Author(s)

Roger D. Peng

Examples

```
## See the help page for `loess.boot' for an example.
```

`samples` *Extract sampling distributions from bootstrapped linear/loess models.*

Description

Extract sampling distributions of various entities from either a linear model or a loess bootstrap. Entities for linear models are currently, model coefficients, residual sum of squares, R-square, and fitted values (given a set of X values in the original bootstrap). For loess, one can extract residual sum of squares and fitted values.

Usage

```
samples(object, name = c("fitted", "coef", "rsquare", "rss"))
```

Arguments

`object`	The output from either `lm.boot` or `loess.boot`.
`name`	The name of the entity to extract. The default is fitted values.

Value

Either a vector or matrix depending on the entity extracted. For example, when extracting the sampling distributions for linear model coefficents, the return value is p x R matrix where p is the number of coefficients and R is the number of bootstrap replicates.

Author(s)

Roger D. Peng

Examples

```
data(airquality)
attach(airquality)
lmodel <- lm(Ozone ~ Solar.R + Wind)
lboot <- lm.boot(lmodel, R = 100)

## Get sampling distributions for coefficients
s <- samples(lboot, "coef")

## Histogram for the intercept
hist(s[1,])
```

two.boot *Two sample bootstrap of differences between univariate statistics.*

Description

two.boot is used to bootstrap the difference between various univariate statistics. An example is the difference of means. Bootstrapping is done by independently resampling from sample1 and sample2.

Usage

```
two.boot(sample1, sample2, FUN, R, student = FALSE, M, weights = NULL, ...)
```

Arguments

sample1	First sample; a vector of numbers.
sample2	Second sample; a vector of numbers.
FUN	The statistic which is applied to each sample. This can be a quoted string or a function name.
R	Number of bootstrap replicates.
student	Should we do a studentized bootstrap? This requires a double bootstrap so it might take longer.
M	If student is set to TRUE, then M is the number of internal bootstrap replications to do.
weights	Resampling weights; a list with two components. The first component of the list is a vector of weights for sample1 and the second component of the list is a vector of weights for sample2.
...	Other (named) arguments that should be passed to FUN.

Details

The differences are always taken as FUN(sample1) - FUN(sample2). If you want the difference to be reversed you need to reverse the order of the arguments sample1 and sample2.

Value

An object of class "simpleboot", which is almost identical to the regular "boot" object. For example, the boot.ci function can be used on this object.

Author(s)

Roger D. Peng

Examples

```
library(boot)
set.seed(50)
x <- rnorm(100, 1)  ## Mean 1 normals
y <- rnorm(100, 0)  ## Mean 0 normals
b <- two.boot(x, y, median, R = 100)
hist(b)  ## Histogram of the bootstrap replicates
b <- two.boot(x, y, quantile, R = 100, probs = .75)
```

Index

∗ **hplot**
    hist.simpleboot, 2
∗ **loess**
    loess.boot, 5
    loess.simpleboot.methods, 7
    plot.loess.simpleboot, 12
∗ **regression**
    lm.boot, 3
    lm.simpleboot.methods, 4
    plot.lm.simpleboot, 11
∗ **univar**
    one.boot, 8
    pairs_boot, 9
    perc, 10
    two.boot, 14
∗ **utilities**
    samples, 13

fitted.lm.simpleboot
        *(*lm.simpleboot.methods*)*, 4
fitted.loess.simpleboot
        *(*loess.simpleboot.methods*)*, 7

hist.simpleboot, 2

lm.boot, 3
lm.simpleboot.methods, 4
loess.boot, 5
loess.simpleboot.methods, 7

one.boot, 8

pairs_boot, 9
perc, 10
plot.lm.simpleboot, 11
plot.loess.simpleboot, 12
print.lm.simpleboot *(*lm.boot*)*, 3
print.loess.simpleboot *(*loess.boot*)*, 5
print.summary.lm.simpleboot
        *(*lm.simpleboot.methods*)*, 4

samples, 13
summary.lm.simpleboot
        *(*lm.simpleboot.methods*)*, 4

two.boot, 14