Package ‘tidyboot’

May 8, 2026

Type Package

Title Tidyverse-Compatible Bootstrapping

Version 0.1.3

Description Compute arbitrary non-parametric bootstrap statistics on data in tidy data frames.

Depends R (>= 3.4.0)

License GPL-3

URL https://github.com/langcog/tidyboot

BugReports https://github.com/langcog/tidyboot/issues

Encoding UTF-8

RoxygenNote 7.3.3

Imports dplyr (>= 0.7.4), modelr (>= 0.1.1), purrr (>= 0.2.4), rlang (>= 0.1.6), tidyr (>= 0.7.2)

NeedsCompilation no

Author Mika Braginsky [aut, cre], Daniel Yurovsky [aut]

Maintainer Mika Braginsky <mika.br@gmail.com>

Repository CRAN

Date/Publication 2026-02-17 20:10:02 UTC

Contents

ci_lower 2
ci_upper 2
tidyboot 3
tidyboot.data.frame 3
tidyboot.logical 4
tidyboot.numeric 5
tidyboot_mean 6

Index **8**

ci_lower — *Confidence interval (lower 2.5%)*

Description

Confidence interval (lower 2.5%)

Usage

```
ci_lower(x, na.rm = FALSE)
```

Arguments

x	A numeric vector
na.rm	A logical value indicating whether NA values should be stripped before the computation proceeds.

Value

2.5

Examples

```
x <- rnorm(1000, mean = 0, sd = 1)
ci_lower(x)
```

ci_upper — *Confidence interval (upper 97.5%)*

Description

Confidence interval (upper 97.5%)

Usage

```
ci_upper(x, na.rm = FALSE)
```

Arguments

x	A numeric vector
na.rm	A logical value indicating whether NA values should be stripped before the computation proceeds.

Value

97.5

Examples

```
x <- rnorm(1000, mean = 0, sd = 1)
ci_upper(x)
```

---

tidyboot — *Non-parametric bootstrap with multiple sample statistics*

Description

`tidyboot` is a generic function for bootstrapping on various data structures. The function invokes particular methods which depend on the class of the first argument.

Usage

```
tidyboot(data, ...)
```

Arguments

`data`	A data structure containing the data to bootstrap.
`...`	Additional arguments passed to particular methods.

Examples

```
## List of available methods
methods(tidyboot)
```

---

tidyboot.data.frame — *Non-parametric bootstrap for data frames*

Description

Computes arbitrary bootstrap statistics on univariate data.

Usage

```
## S3 method for class 'data.frame'
tidyboot(
  data,
  column = NULL,
  summary_function = mean,
  statistics_functions,
  nboot = 1000,
  ...
)
```

Arguments

`data`	A data frame.
`column`	A column of `data` to bootstrap over (if not supplied, summary_function and statistic_function must operate over the appropriate data frame).
`summary_function`	A function to be computed over each set of samples as a data frame, or a function to be computed over each set of samples as a single column of a data frame indicated by `column` (defaults to `mean`).
`statistics_functions`	A function to be computed over each set of samples as a data frame, or a named list of functions to be computed over each set of samples as a single column of a data frame indicated by `column`.
`nboot`	The number of bootstrap samples to take (defaults to `1000`).
`...`	Other arguments passed from generic.

Examples

```
## Mean and 95% confidence interval for 500 samples from two different normal distributions
require(dplyr)
gauss1 <- tibble(value = rnorm(500, mean = 0, sd = 1), condition = 1)
gauss2 <- tibble(value = rnorm(500, mean = 2, sd = 3), condition = 2)
df <- bind_rows(gauss1, gauss2)

mean_ci_funs <- list("ci_lower" = ci_lower, "mean" = mean, "ci_upper" = ci_upper)
df %>% group_by(condition) %>%
  tidyboot(column = value, summary_function = mean, statistics_functions = mean_ci_funs)

df %>% group_by(condition) %>%
  tidyboot(summary_function = function(x) x %>% summarise(stat = mean(value)),
           statistics_functions = function(x) x %>%
             summarise(across(stat, mean_ci_funs, .names = "{.fn}")))
```

`tidyboot.logical` *Non-parametric bootstrap for logical vector data*

Description

Computes arbitrary bootstrap statistics on univariate data.

Usage

```
## S3 method for class 'logical'
tidyboot(
  data,
  summary_function = mean,
  statistics_functions,
```

```
  nboot = 1000,
  size = 1,
  replace = TRUE,
  ...
)
```

Arguments

`data`	A logical vector of data to bootstrap over.
`summary_function`	A function to be computed over each set of samples. This function needs to take a vector and return a single number (defaults to `mean`).
`statistics_functions`	A named list of functions to be computed over the set of summary values from all samples.
`nboot`	The number of bootstrap samples to take (defaults to `1000`).
`size`	The fraction of items to sample (defaults to 1).
`replace`	Logical indicating whether to sample with replacement (defaults to `TRUE`).
`...`	Other arguments passed from generic.

Examples

```
## Mean and 95% confidence interval for 500 samples from a binomial distribution
x <- as.logical(rbinom(500, 1, 0.5))
tidyboot(x, statistics_functions = c(ci_lower, mean, ci_upper))
```

`tidyboot.numeric` *Non-parametric bootstrap for numeric vector data*

Description

Computes arbitrary bootstrap statistics on univariate data.

Usage

```
## S3 method for class 'numeric'
tidyboot(
  data,
  summary_function = mean,
  statistics_functions,
  nboot = 1000,
  size = 1,
  replace = TRUE,
  ...
)
```

Arguments

`data`	A numeric vector of data to bootstrap over.
`summary_function`	A function to be computed over each set of samples. This function needs to take a vector and return a single number (defaults to `mean`).
`statistics_functions`	A named list of functions to be computed over the set of summary values from all samples.
`nboot`	The number of bootstrap samples to take (defaults to `1000`).
`size`	The fraction of items to sample (defaults to 1).
`replace`	Logical indicating whether to sample with replacement (defaults to `TRUE`).
`...`	Other arguments passed from generic.

Examples

```
## Mean and 95% confidence interval for 500 samples from a normal distribution
x <- rnorm(500, mean = 0, sd = 1)
tidyboot(x, statistics_functions = list("ci_lower" = ci_lower,
                                        "mean" = mean,
                                        "ci_upper" = ci_upper))
```

`tidyboot_mean` *Non-parametric bootstrap and empirical central tendency for data frames Designed to make standard use of* `tidyboot.data.frame` *easier*

Description

Computes arbitrary bootstrap statistics on univariate data. NOTE: Both empirical functions and bootstrapping functions will be computed over the grouping variables currently specified for the data frame.

Usage

```
tidyboot_mean(data, column, nboot = 1000, na.rm = FALSE)
```

Arguments

`data`	A data frame.
`column`	A column of `data` to bootstrap over.
`nboot`	The number of bootstrap samples to take (defaults to `1000`).
`na.rm`	A logical value indicating whether NA values should be stripped before the computation proceeds.

Examples

```
## Mean and 95% confidence interval for 500 samples from two different normal distributions
require(dplyr)
gauss1 <- tibble(value = rnorm(500, mean = 0, sd = 1), condition = 1)
gauss2 <- tibble(value = rnorm(500, mean = 2, sd = 3), condition = 2)
df <- bind_rows(gauss1, gauss2)

df %>%
 group_by(condition) %>%
 tidyboot_mean(column = value)
```

Index

ci_lower, 2
ci_upper, 2

tidyboot, 3
tidyboot.data.frame, 3
tidyboot.logical, 4
tidyboot.numeric, 5
tidyboot_mean, 6