Package ‘silviculture’

May 29, 2025

Type Package

Title Utility Functions for Forest Inventory and Silviculture

Version 0.1.0

Description Perform common dendrometry operations such as inventory preparing, and inventory data analysis.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Imports cli, dplyr, S7

Depends R (>= 4.1.0)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

URL `https://cidree.github.io/silviculture/`

Collate 'biomass.R' 'data.R' 'utils-not-exported.R'
'dendrometry-summary.R' 'dendrometry.R' 'globals.R'
'inventory.R' 'lidar.R' 'volume.R' 'zzz.R'

NeedsCompilation no

Author Adrián Cidre González [aut, cre]

Maintainer Adrián Cidre González <`adrian.cidre@gmail.com`>

Repository CRAN

Date/Publication 2025-05-29 08:50:02 UTC

Contents

`biomass_models` *Biomass models*

Description

Biomass models available in `silviculture` package. If you would like to suggest new models, please open a new issue.

Usage

```
biomass_models
```

Format

A `tibble`

`Inventory` *Inventory class*

Description

A list containing forest inventory data summaries. It includes data by diametric class (`dclass_metrics`) and by plot and species (`group_metrics`).

Usage

```
Inventory(dclass_metrics = data.frame(), group_metrics = data.frame())
```

Arguments

`dclass_metrics`	A data.frame summarised by diametric class with variables such as plot_id, species, dclass, height, ntrees, ntrees_ha, h0, dg, and g_ha.
`group_metrics`	A data.frame summarised by plot and species with variables such as plot_id, species, d_mean, d_median, d_sd, dg, h_mean, h_median, h_sd, h_lorey, h0, ntrees, ntrees_ha, g_ha, and spacing.

Value

An S7 `Inventory` object, which contains the inventory data.

`inventory_samples` *Forest inventory samples*

Description

Inventory data from Spanish National Forest Inventory

Usage

```
inventory_samples
```

Format

A `tibble`

`lid_fcov` *Calculate Forest Fraction Cover from LiDAR Data*

Description

This function calculates the forest fraction cover (Fcov) from LiDAR data. The Fcov in LiDAR is defined as the proportion of first returns above a specified height threshold (default: 5 meters) relative to the total number of first returns.

Usage

```
lid_fcov(z, rn, th = 5)
```

Arguments

`z`	A numeric vector representing the heights of LiDAR returns
`rn`	An integer vector indicating the return number for each LiDAR point. First returns are identified by a value of `1`
`th`	a numeric vector of length one specifying the height threshold

Value

A numeric value representing the forest fraction cover, which is the proportion of first returns with heights greater than 5 meters.

Examples

```
# Example data
z <- c(2, 6, 10, 4, 15)
rn <- c(1, 1, 2, 1, 1)

# Calculate forest fraction cover
lid_fcov(z, rn)
```

`lid_lhdi` *LiDAR-derived Height Diversity Index (LHDI)*

Description

LiDAR metric that calculates the LiDAR Height Diversity Index, which can be used in `lidR` `*_metrics` functions

Usage

```
lid_lhdi(z, interval = 0.5)
```

Arguments

`z`	coordinate Z (height) of the point
`interval`	height of the intervals to calculate the metric

Value

numeric

References

Listopad, C. M. C. S., Masters, R. E., Drake, J., Weishampel, J., & Branquinho, C. (2015). Structural diversity indices based on airborne LiDAR as ecological indicators for managing highly dynamic landscapes. Ecological Indicators, 57, 268–279. doi:10.1016/j.ecolind.2015.04.017

Examples

```
1 + 1 ## TODO
```

`plot` *Plot an object*

Description

Generic for plotting objects.

Usage

```
plot(x, ...)
```

Arguments

`x`	Object to plot.
`...`	Other arguments passed to methods.

Value

Usually called for side-effects (producing a plot).

`SampleSize` *SampleSize class*

Description

A list containing the summary data of silv_sample_size. It includes input data (sampling_opts) and output data (sampling_res)

Usage

```
SampleSize(sampling_res = list(), sampling_opts = list())
```

Arguments

`sampling_res`	A list containing the results of sample size calculations.
`sampling_opts`	A list containing the input options of sample size calculations.

Value

An S7 `SampleSize` object

`silv_basal_area` *Calculates Basal Area*

Description

Calculates Basal Area in square meters.

Usage

```
silv_basal_area(diameter, ntrees = NULL, units = "cm")
```

Arguments

`diameter`	Numeric vector of diameters or diameter classes
`ntrees`	Numeric vector with number of trees of the diameter class per hectare. If `ntrees = NULL`, the function will assume that each diameter corresponds to only one tree. Therefore, basal area will be calculated for each individual tree
`units`	The units of the diameter (one of `cm`, `mm`, or `m`)

Details

The function uses the next formula:

$G = \frac{\pi}{40000} \cdot D^2$

where G is the basal area in m^2, and D is the diameter in the `units` specified in the function. It is recommended to use the squared mean diameter calculated with silv_sqrmean_diameter

Value

A numeric vector

Examples

```
## calculate G for inventory data grouped by plot_id and species
library(dplyr)
inventory_samples |>
mutate(dclass = silv_diametric_class(diameter)) |>
  summarise(
    height = mean(height, na.rm = TRUE),
    ntrees = n(),
    .by    = c(plot_id, species, dclass)
  ) |>
  mutate(
    ntrees_ha = silv_ntrees_ha(ntrees, plot_size = 10),
    dg        = silv_sqrmean_diameter(dclass, ntrees_ha),
    g         = silv_basal_area(dclass, ntrees_ha),
    .by       = c(plot_id, species)
  )
```

```
## calculate individual basal area
silv_basal_area(c(23, 11, 43.5, 94))
```

`silv_biomass` *Calculate Tree Biomass*

Description

Computes the biomass of a tree species using species-specific allometric equations (in kg).

Usage

```
silv_biomass(
  diameter = NULL,
  height = NULL,
  ntrees = NULL,
  species = NULL,
  component = "stem",
  model = "ruiz-peinado-2012",
  return_rmse = FALSE,
  quiet = FALSE
)
```

Arguments

`diameter`	A numeric vector of tree diameters (in cm).
`height`	A numeric vector of tree heights (in m).
`ntrees`	An optional numeric value indicating the number of trees in this diameter-height class. Defaults to 1 if NULL.
`species`	A character string specifying the scientific name of the tree species. See Details for available species.
`component`	A character string specifying the tree component for biomass calculation (e.g., "tree", "stem", "branches"). See Details.
`model`	A character string indicating the ID of the publication in which the model was developed. Currently supported models: "ruiz-peinado-2012" (hardwood species in Spain) and "ruiz-peinado-2011" (softwood species in Spain). See Details.
`return_rmse`	A logical value. If TRUE, the function returns the root mean squared error (RMSE) of the selected model instead of the biomass value.
`quiet`	A logical value. If TRUE, suppresses any informational messages.

Details

The function estimates biomass using validated allometric models available in the dataset biomass_models. The available models include:

- **ruiz-peinado-2011**: Developed for softwood species in Spain.
- **ruiz-peinado-2012**: Developed for hardwood species in Spain.

Users can check the list of supported species and their corresponding components in biomass_models.

If you would like to suggest additional models, please open a new issue on GitHub.

Value

A numeric vector of biomass values (in kg). If `return_rmse = TRUE`, returns the RMSE instead.

Examples

```
# Calculate biomass for a single tree
silv_biomass(
  diameter = 45,
  height   = 22,
  species  = "Pinus pinaster",
  model    = "ruiz-peinado-2011"
)
```

`silv_diametric_class` *Classify diameters in classes*

Description

Classifies the measured diameters into classes of a specified length

Usage

```
silv_diametric_class(
  diameter,
  dmin = 7.5,
  dmax = NULL,
  class_length = 5,
  include_lowest = TRUE,
  return_intervals = FALSE
)
```

Arguments

`diameter`	A numeric vector of diameters
`dmin`	The minimum inventory diameter in centimeters
`dmax`	The maximum inventory diameter in centimeters. Values that are greater than `dmax` are included in the greatest class
`class_length`	The length of the class in centimeters
`include_lowest`	Logical. If TRUE (the default), the intervals are `[dim1, dim2)`. If FALSE, the intervals are `(dim1, dim2]` [dim1, dim2)`. If FALSE, the intervals are `(dim1, dim2]: R:dim1,%20dim2)%60.%20If%20FALSE
`return_intervals`	If FALSE, it returns the intervals. Otherwise (the default), it returns the class center

Value

A numeric vector

Examples

```
library(dplyr)
inventory_samples |>
  mutate(dclass = silv_diametric_class(diameter))
```

`silv_dominant_height` *Calculates the dominant height*

Description

Calculates the dominant height using the Assman equation or the Hart equation

Usage

```
silv_dominant_height(diameter, height, ntrees = NULL, which = "assman")
```

Arguments

`diameter`	Numeric vector with diameter classes
`height`	Numeric vector with averaged heights by diameter class
`ntrees`	Optional. Numeric vector with number of trees per hectare. Use this argument when you have aggregated data by diametric classes (see details).
`which`	The method to calculate the dominant height (see details)

Details

The dominant height H_0 is the mean height of dominant trees, which is less affected than overall mean height by thinning or other treatments.

- **Assman**: calculates the H_0 as the mean height of the 100 thickest trees per hectare
- **Hart**: calculates the H_0 as the mean height of the 100 tallest trees per hectare

When `ntrees = NULL`, the function will assume that each diameter and height belongs to only one trees. If you have data aggregated by hectare, you'll use the number of trees per hectare in this argument.

Value

A numeric vector

References

Assmann, E. (1970) The principles of forest yield study: Studies in the organic production, structure, increment, and yield of forest stands. Pergamon Press, Oxford.

Examples

```
## calculate h0 for inventory data grouped by plot_id and species
library(dplyr)
inventory_samples |>
mutate(dclass = silv_diametric_class(diameter)) |>
  summarise(
    height = mean(height, na.rm = TRUE),
    ntrees = n(),
    .by    = c(plot_id, species, dclass)
  ) |>
  mutate(
    ntrees_ha = silv_ntrees_ha(ntrees, plot_size = 10),
    h0        = silv_dominant_height(dclass, height, ntrees_ha),
    .by       = c(plot_id, species)
  )
```

`silv_lorey_height` *Calculates Lorey's Height*

Description

Tree's mean height weighted by basal area

Usage

```
silv_lorey_height(height, g, ntrees = NULL)
```

Arguments

`height`	Numeric vector of heights
`g`	Numeric vector of basal areas
`ntrees`	Optional. Numeric vector of number of trees per hectare. Use this argument when you have aggregated data by diametric classes (see details).

Details

The function calculates Lorey's mean height according to:

$$h_L = \frac{\sum n_i g_i h_i}{\sum n_i g_i}$$

When ntrees is not provided (i.e. `ntrees = NULL`) the formula is simply the weighted mean of the provided heights and basal areas:

$$h_L = \frac{\sum g_i h_i}{\sum g_i}$$

Value

A numeric vector

Examples

```
## Calculate Lorey's Height by plot and species
library(dplyr)
inventory_samples |>
  mutate(g = silv_basal_area(diameter)) |>
  summarise(
    lh  = silv_lorey_height(height, g),
    .by = c(plot_id, species)
  )
```

`silv_ntrees_ha` *Calculates number of trees per hectare*

Description

Calculates number of trees per hectare for a given plot size and shape

Usage

```
silv_ntrees_ha(ntrees, plot_size, plot_shape = "circular")
```

Arguments

`ntrees`	A numeric vector representing the number of trees in a sampling plot
`plot_size`	A numeric vector of length one for circular radius in meters; or a numeric vector of length two for each side of a rectangular plot shape
`plot_shape`	The shape of the sampling plot. Either `circular` or `rectangular`

Value

A numeric vector

Examples

```
library(dplyr)
## Circular plot of 10 meters radius
inventory_samples |>
  count(plot_id, species) |>
  mutate(
    ntrees_ha = silv_ntrees_ha(n, plot_size = 10)
  )

## Rectangular plot of 10x15 meters
inventory_samples |>
  count(plot_id, species) |>
  mutate(
    ntrees_ha = silv_ntrees_ha(
      n,
      plot_size = c(10, 15),
      plot_shape = "rectangular"
     )
  )
```

`silv_sample_size` *Calculates sample size for a random sampling inventory*

Description

Calculates sample size for a random sampling inventory

Usage

```
silv_sample_size(
  x,
  plot_size = 100,
  total_area = 150000,
  max_error = 0.05,
  conf_level = 0.95,
  max_iter = 1000,
  quiet = FALSE
)
```

Arguments

`x`	vector of field survey
`plot_size`	a numeric vector of length one with plot size in squared meters
`total_area`	total area of the study area in squared meters
`max_error`	maximum allowed error
`conf_level`	confidence level
`max_iter`	maximum number of iteration to find the plot size
`quiet`	if `TRUE`, messages will be supressed

Value

SampleSize object

Examples

```
## pilot inventory measuring 4 plots of 25x25 meters
## total forest area 15 ha
## measured variable (x): basal area per hectare
silv_sample_size(
  x          = c(33, 37.5, 42, 35.2),
  plot_size  = 25 * 25,  # squared plot of 25x25
  total_area = 15 * 1e4, # 15 ha
  max_error  = 0.05,
  conf_level = 0.95,
  max_iter   = 100
)
```

`silv_spacing_index` *Hart or Hart-Becking spacing index*

Description

Calculates the Hart Index or the Hart-Becking Index for even-aged stands

Usage

```
silv_spacing_index(h0, ntrees, which = "hart")
```

Arguments

`h0`	Numeric vector with dominant height
`ntrees`	Numeric vector with number of trees of the dominant height per hectare
`which`	A character with the name of the index (either `hart` or `hart-brecking`). See details

Details

The spacing index can be used to determine whether a thinning is needed or not, and also to determine how intense it should be.

- **Hart Index**: it assumes even-aged stands with square planting pattern.
- **Hart-Brecking Index**: it assumes triangular planting pattern.

Value

A numeric vector

References

Assmann, E. (1970) The principles of forest yield study: Studies in the organic production, structure, increment, and yield of forest stands. Pergamon Press, Oxford.

Examples

```
library(dplyr)
## Calculate spacing index for each plot
inventory_samples |>
  summarise(
    h0     = silv_dominant_height(diameter, height),
    ntrees = n(),
    .by    = plot_id
  ) |>
  ## calculate number of trees per hectare
  mutate(ntrees_ha = silv_ntrees_ha(ntrees, plot_size = 14.1)) |>
  mutate(spacing = silv_spacing_index(h0, ntrees_ha))
```

`silv_sqrmean_diameter` *Calculates the squared mean diameter*

Description

Calculates the squared mean diameter

Usage

```
silv_sqrmean_diameter(diameter, ntrees = NULL)
```

Arguments

`diameter`	Numeric vector of diameters or diameter classes
`ntrees`	Numeric vector with number of trees of the diameter class per hectare. If `ntrees = NULL`, the function will assume that each diameter corresponds to only one tree. Therefore, basal area will be calculated for each individual tree

Value

A numeric vector

Examples

```
## calculate dg for inventory data grouped by plot_id and species
library(dplyr)
inventory_samples |>
mutate(dclass = silv_diametric_class(diameter)) |>
  summarise(
    height = mean(height, na.rm = TRUE),
    ntrees = n(),
    .by    = c(plot_id, species, dclass)
  ) |>
  mutate(
    ntrees_ha = silv_ntrees_ha(ntrees, plot_size = 10),
    h0        = silv_dominant_height(dclass, height, ntrees_ha),
    dg        = silv_sqrmean_diameter(dclass, ntrees_ha),
    .by       = c(plot_id, species)
  )

## calculate dg for a vector of diameters
silv_sqrmean_diameter(c(12.5, 23.5, 14, 16, 18.5))
```

`silv_summary` *Calculates a bunch of forest metrics*

Description

Summarize forest inventory data calculating most typical variables

Usage

```
silv_summary(
  data,
  diameter,
  height,
  plot_size,
  .groups = NULL,
  plot_shape = "circular",
  dmin = 7.5,
  dmax = NULL,
  class_length = 5,
  include_lowest = TRUE,
  which_h0 = "assman",
  which_spacing = "hart"
)
```

Arguments

`data`	A tibble of inventory data
`diameter`	A column with inventory diameters
`height`	A column with inventory heights
`plot_size`	The size of the plot. See silv_ntrees_ha
`.groups`	A character vector with variables to group by (e.g. plot id, tree species, etc)
`plot_shape`	The shape of the sampling plot. Either `circular` or `rectangular`
`dmin`	The minimum inventory diameter in centimeters
`dmax`	The maximum inventory diameter in centimeters. Values that are greater than `dmax` are included in the greatest class
`class_length`	The length of the class in centimeters
`include_lowest`	Logical. If TRUE (the default), the intervals are `[dim1, dim2)`. If FALSE, the intervals are `(dim1, dim2]` [dim1, dim2)`. If FALSE, the intervals are `(dim1, dim2]: R:dim1,%20dim2)%60.%20If%20FALSE
`which_h0`	The method to calculate the dominant height. See silv_dominant_height
`which_spacing`	A character with the name of the index (either `hart` or `hart-brecking`). See silv_spacing_index

Details

The function calculates many inventory parameters and returns two tibbles:

- **dclass_metrics**: metrics summarized by .groups and diametric classes
- **group_metrics**: metrics summarized by .groups

Value

an S7 `Inventory` list with 2 `tibbles`

Examples

```
silv_summary(
  data      = inventory_samples,
  diameter  = diameter,
  height    = height,
  plot_size = 10,
  .groups   = c("plot_id", "species")
 )
```

`silv_volume` *Calculate Tree Volume*

Description

This function calculates the volume of a tree or logs using different formulas: Pressler, Huber, Smalian, and Newton. The appropriate diameter and height parameters must be provided depending on the selected formula.

Usage

```
silv_volume(
  diameter_base = NULL,
  diameter_top = NULL,
  diameter_center = NULL,
  diameter = NULL,
  height = NULL,
  formula = "pressler",
  ntrees = NULL
)
```

Arguments

`diameter_base`	A numeric vector. The diameter at the base of the tree (required for Pressler, Smalian, and Newton formulas).
`diameter_top`	A numeric vector. The diameter at the top of the tree (required for Smalian and Newton formulas).
`diameter_center`	A numeric vector. The diameter at the center of the tree (required for Huber and Newton formulas).
`diameter`	A numeric vector. The diameter at breast height (used in Pressler formula if provided instead of `diameter_base`).
`height`	A numeric vector. The tree or log height (required for all formulas).
`formula`	Character. The volume formula to use. Options: `"pressler"`, `"huber"`, `"smalian"`, `"newton"`. Default is `"pressler"`.
`ntrees`	A numeric vector with number of trees of the same dimensions. Default is 1.

Value

A numeric value representing the tree volume.

Examples

```
silv_volume(diameter_base = 30, height = 20, formula = "pressler")
silv_volume(diameter_center = 25, height = 15, formula = "huber")
silv_volume(diameter_base = 30, diameter_top = 20, height = 20, formula = "smalian")
```

Index