

Package: silviculture (via r-universe)

October 4, 2026

Type Package

Title Utility Functions for Forest Inventory and Silviculture

Version 0.2.0

Author Adrián Cidre González [aut, cre], Aitor Vázquez Veloso [aut]

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

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

License GPL (>= 3)

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

Imports cli, dplyr, lifecycle, rlang, S7, tibble, tidyr

Depends R (>= 4.1.0)

Suggests knitr, quarto, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

URL <https://cidree.github.io/silviculture/>

Collate 'data.R' 'utils-not-exported.R' 'deprecated-funs.R'
'inventory-summary.R' 'metrics-lidar.R'
'metrics-stand-density.R' 'metrics-stand-level.R'
'metrics-tree-level.R' 'predict-biomass.R' 'predict-carbon.R'
'predict-height.R' 'predict-snfi-volume.R' 'sample-size.R'
'silviculture-package.R' 'treatment-thinning.R'
'tree-summary.R' 'utils-assert.R' 'zzz.R'

Config/roxygen2/version 8.1.0

VignetteBuilder quarto

Config/pak/sysreqs libicu-dev

Repository <https://cidree.r-universe.dev>

Date/Publication 2026-10-04 08:42:58 UTC

RemoteUrl <https://github.com/cidree/silviculture>

RemoteRef HEAD

RemoteSha 23cca7a2293a6646e7a914216088993c959f5213

Contents

biomass_models	3
carbon_models	4
eq_biomass_cudjoe_2024	5
eq_biomass_dieguez_aranda_2009	6
eq_biomass_manrique_2017	8
eq_biomass_menendez_2022	9
eq_biomass_montero_2005	11
eq_biomass_ruiz_peinado_2011	12
eq_biomass_ruiz_peinado_2012	14
eq_hd_vazquez_veloso_2025	15
inventory_samples	17
lid_fcov	17
lid_lhdi	18
plot	18
predict_carbon	19
sdi_coefficients	21
sdimax_models	22
silv_basal_area	23
silv_biomass	24
silv_density_hart	25
silv_density_ntrees_ha	26
silv_density_sdi	27
silv_density_sdi_auto	28
silv_density_sdi_class	30
silv_density_sdimax	31
silv_diametric_class	32
silv_dominant_height	33
silv_lorey_height	34
silv_ntrees_ha	35
silv_predict_biomass	36
silv_predict_biomass_auto	38
silv_predict_biomass_components	40
silv_predict_height	41
silv_predict_snfi_volume	43
silv_sample_size	45
silv_sample_size_simple	47
silv_sample_size_stratified	48
silv_site_factor	51
silv_snfi_provinces	51
silv_snfi_species	52
silv_spacing_index	52
silv_sqrmean_diameter	53
silv_stand_basal_area	54
silv_stand_dominant_diameter	55
silv_stand_dominant_height	56
silv_stand_lorey_height	58

silv_stand_qmean_diameter	59
silv_stand_slenderness	60
silv_summary	60
silv_summary_mortality	63
silv_summary_species	64
silv_treatment_thinning	65
silv_tree_bal	67
silv_tree_bas	68
silv_tree_basal_area	69
silv_tree_basal_area_ha	70
silv_tree_circumference	70
silv_tree_coordinates	71
silv_tree_crown_ratio	72
silv_tree_dclass	72
silv_tree_expansion_factor	73
silv_tree_mean_dbh	74
silv_tree_slenderness	75
silv_tree_summary	75
silv_tree_volume	77
silv_volume	78
site_factor_models	79
snfi3_volume_coefficients	80
snfi4_volume_coefficients	81
Index	82

biomass_models	<i>Biomass models</i>
----------------	-----------------------

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

carbon_models

*Carbon content models***Description**

Carbon content percentages per biomass component available in the silviculture package, compiled from Díeguez-Aranda et al. (2009) and Montero et al. (2005). If you would like to suggest new models, please open a new issue.

Usage

carbon_models

Format

A tibble with 264 rows and 13 variables:

article_id Character. Short identifier of the source article (e.g. "montero-2005", "dieguez-aranda-2009").

title Character. Full title of the source article.

doi_url Character. DOI URL of the source article.

country Character. Country where the study was conducted.

region Character. Region within the country.

species Character. Scientific name of the tree species.

biomass_group Character. Biomass group ("AGB" or "BGB").

tree_group Character. Sub-group within the biomass group.

tree_component Character. Tree component (e.g. "stem", "bark", "thick branches", "small branches", "twigs", "leaves", "needles", "roots").

carbon_percentage Numeric. Carbon content as a percentage (0-100). Multiply by the component biomass and divide by 100 to obtain carbon mass in the same units. NA where not reported in the source.

r2 Numeric. Coefficient of determination of the original model, if reported.

rmse Numeric. Root mean square error of the original model, if reported.

obs Character. Additional observations from the source article.

References

Díeguez-Aranda, U., Rojo Alboreca, A., Castedo-Dorado, F., Álvarez González, J.G., Barrio-Anta, M., Crecente-Campo, F., González González, J.M., Pérez-Cruzado, C., Rodríguez Soalleiro, R., López-Sánchez, C.A., Balboa-Murias, M.A., Gorgoso Varela, J.J., Sánchez Rodríguez, F. (2009). *Herramientas selvícolas para la gestión forestal sostenible en Galicia*. Xunta de Galicia.

Montero, G., Ruiz-Peinado, R., Muñoz, M. (2005). *Producción de biomasa y fijación de CO2 por los bosques españoles*. Monografías INIA: Serie Forestal n.º 13. Instituto Nacional de Investigación y Tecnología Agraria y Alimentaria, Madrid.

eq_biomass_cudjoe_2024
<i>Biomass equations for 2 species in Castille and León (Spain)</i>

Description

Allometric equations adjusted for *Quercus petraea*, and *Pinus sylvestris* in Castille and León (Spain)

Usage

eq_biomass_cudjoe_2024(species, component = "AGB", return_rmse = FALSE)

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species.
component	A character string specifying the tree component for biomass calculation (e.g., "stem", "branches"). 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.

Details

Supported species (3):

- *Pinus sylvestris*
- *Quercus petraea*
- "mixed" — mixed stand of *Pinus sylvestris* × *Quercus petraea*

Available components:

Aboveground group (summed automatically):

- "AGB" — total aboveground biomass (sum of all components below)

Individual tree components (species availability varies):

- "stem" — stem wood (all species)
- "thick branches" — branches > 7 cm (all species)
- "medium branches and small branches" — branches < 7 cm (all species)
- "leaves" — foliage/needles (*Pinus sylvestris* only)

Note that no belowground biomass (BGB / roots) or total-tree equations are available in the source paper for this model.

Users can check all available species and components in the [biomass_models](#) dataset provided by the library.

Value

A ModelBiomass object containing the configured model parameters and expressions.

See Also

[silv_predict_biomass\(\)](#), [biomass_models](#), [eq_biomass_montero_2005\(\)](#), [eq_biomass_dieguez_aranda_2009\(\)](#), [eq_biomass_ruiz_peinado_2011\(\)](#), [eq_biomass_ruiz_peinado_2012\(\)](#), [eq_biomass_manrique_2017\(\)](#), [eq_biomass_menendez_2022\(\)](#)

Examples

```
## Aboveground biomass for a mixed stand
eq_biomass_cudjoe_2024("mixed", "AGB")
```

eq_biomass_diegue z _aranda_2009
<i>Biomass equations for Galician species</i>

Description

Allometric equations adjusted for Galician (Spain) species

Usage

```
eq_biomass_dieguez_aranda_2009(  
  species,  
  component = "stem",  
  return_r2 = FALSE,  
  return_rmse = FALSE  
)
```

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species.
component	A character string specifying the tree component for biomass calculation (e.g., "tree", "stem", "branches"). See Details.
return_r2	A logical value. If TRUE, the function returns the root mean squared error (RMSE) of the selected model instead of the biomass value.
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.

Details

Supported species (7):

Betula alba, *Eucalyptus globulus*, *Eucalyptus nitens*, *Pinus pinaster*, *Pinus radiata*, *Pinus sylvestris*, *Quercus robur*

Available components:

Aboveground / belowground groups (summed automatically):

- "AGB" — total aboveground biomass
- "BGB" — total belowground biomass (roots)

Tree structural groups:

- "stem" — stem fraction(s)
- "branches" — all branch fractions combined
- "roots" — roots (equivalent to BGB)

Individual tree components (species availability varies):

- "stem and thick branches" — stem together with thickest branches
- "thick branches" — branches > 7 cm
- "medium branches" — branches 2–7 cm
- "small branches" — branches 0.5–2 cm
- "twigs" — branches < 0.5 cm
- "dry branches" — dead attached branches
- "leaves" — foliage (including needles)
- "roots" — coarse roots

Note that total-tree equations ("tree" / "all") are not available for this model. Also, *Eucalyptus globulus*, *Eucalyptus nitens*, and *Pinus pinaster* lack BGB equations (requesting "BGB" or "roots" will fail).

Users can check all available species and components in the [biomass_models](#) dataset provided by the library.

Value

A ModelBiomass object containing the configured model parameters and expressions.

See Also

[silv_predict_biomass\(\)](#), [biomass_models](#), [eq_biomass_montero_2005\(\)](#), [eq_biomass_ruiz_peinado_2011\(\)](#), [eq_biomass_ruiz_peinado_2012\(\)](#), [eq_biomass_manrique_2017\(\)](#), [eq_biomass_menendez_2022\(\)](#), [eq_biomass_cudjoe_2024\(\)](#)

Examples

```
## Aboveground biomass for Pinus pinaster
eq_biomass_diegues_aranda_2009("Pinus pinaster", "AGB")
```

eq_biomass_manrique_2017

Biomass equations two Quercus species

Description

Allometric equations adjusted for *Quercus petraea* and *Quercus pyrenaica* in Palencia, Spain

Usage

```
eq_biomass_manrique_2017(
  species,
  component = "AGB",
  return_r2 = FALSE,
  return_rmse = FALSE
)
```

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species.
component	A character string specifying the tree component for biomass calculation (e.g., "stem", "branches"). See Details.
return_r2	A logical value. If TRUE, the function returns the root mean squared error (RMSE) of the selected model instead of the biomass value.
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.

Details

Supported species (2):

- *Quercus petraea*
- *Quercus pyrenaica*

Available components:

Aboveground group (summed automatically):

- "AGB" — total aboveground biomass (sum of all components below)

Individual tree components:

- "stem and thick branches" — stem together with branches > 7 cm
- "medium branches" — branches 2–7 cm
- "small branches" — branches < 2 cm

Note that no belowground biomass (BGB / roots) or total-tree equations are available in the source paper.

Users can check all available species and components in the [biomass_models](#) dataset provided by the library.

Value

A ModelBiomass object containing the configured model parameters and expressions.

See Also

[silv_predict_biomass\(\)](#), [biomass_models](#), [eq_biomass_montero_2005\(\)](#), [eq_biomass_diegueza_aranda_2009\(\)](#), [eq_biomass_ruiz_peinado_2011\(\)](#), [eq_biomass_ruiz_peinado_2012\(\)](#), [eq_biomass_menendez_2022\(\)](#), [eq_biomass_cudjoe_2024\(\)](#)

Examples

```
## Aboveground biomass for Quercus petraea
eq_biomass_manrique_2017("Quercus petraea", "AGB")
```

eq_biomass_menendez_2022
<i>Biomass equations for young Spanish plantations</i>

Description

Allometric equations for young (<30) plantations of 18 Spanish species including broadleaf and conifer species. Only aboveground biomass.

Usage

```
eq_biomass_menendez_2022(species, return_r2 = FALSE, return_rmse = FALSE)
```

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species.
return_r2	A logical value. If TRUE, the function returns the root mean squared error (RMSE) of the selected model instead of the biomass value.
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.

Details

Supported species (18):

Betula sp., *Fagus sylvatica*, *Juniperus thurifera*, *Pinus halepensis*, *Pinus nigra*, *Pinus pinaster*, *Pinus pinea*, *Pinus radiata*, *Pinus sylvestris*, *Quercus faginea*, *Quercus ilex*, *Quercus petraea*, *Quercus pyrenaica*, *Quercus robur*, *Quercus suber*

Generic equations are also available for functional groups:

- "Conifers" — generic equation for conifer species
- "Deciduous broadleaves" — generic equation for deciduous broadleaf species
- "Evergreen broadleaves" — generic equation for evergreen broadleaf species

Available components:

- "AGB" — aboveground biomass (only component available; no component argument needed)

Important — non-standard input variables:

Unlike other biomass models, these equations were fitted on **young plantations (< 30 years)** and use different predictor variables:

- Most species use `rcd` = root collar diameter (cm), **not** diameter at breast height. Pass it via the `rcd` argument of `silv_predict_biomass()`.
- Some species (*Pinus halepensis*, *Pinus nigra*, *Quercus suber*, *Evergreen broadleaves*) use `bp` = biomass packing (m^3). Pass it via the `bp` argument of `silv_predict_biomass()`.
- *Betula* sp. uses only `h` (total height).

Note that no belowground biomass (BGB / roots) or total-tree equations are available in the source paper for this model.

Users can check all available species and components in the `biomass_models` dataset provided by the library.

Value

A `ModelBiomass` object containing the configured model parameters and expressions.

See Also

`silv_predict_biomass()`, `biomass_models`, `eq_biomass_montero_2005()`, `eq_biomass_diegueza_aranda_2009()`, `eq_biomass_ruiz_peinado_2011()`, `eq_biomass_ruiz_peinado_2012()`, `eq_biomass_manrique_2017()`, `eq_biomass_cudjoe_2024()`

Examples

```
## AGB for Fagus sylvatica using root collar diameter
eq_biomass_menendez_2022("Fagus sylvatica")
```

eq_biomass_montero_2005

Biomass equations for Spanish species

Description

Allometric equations adjusted for Spanish species

Usage

```
eq_biomass_montero_2005(species, component = "stem", return_r2 = FALSE)
```

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species.
component	A character string specifying the tree component for biomass calculation (e.g., "tree", "stem", "branches"). See Details.
return_r2	A logical value. If TRUE, the function returns the root mean squared error (RMSE) of the selected model instead of the biomass value.

Details

Supported species (35):

Abies alba, *Abies pinsapo*, *Alnus glutinosa*, *Betula* spp., *Castanea sativa*, *Ceratonia siliqua*, *Erica arborea*, *Eucalyptus* spp., *Fagus sylvatica*, *Fraxinus* spp., *Ilex canariensis*, *Juniperus oxycedrus*, *Juniperus phoenicea*, *Juniperus thurifera*, *Laurus azorica*, *Myrica faya*, *Olea europaea* var. *sylvestris*, *Other broadleaves*, *Other conifers*, *Other laurel species*, *Pinus canariensis*, *Pinus halepensis*, *Pinus nigra*, *Pinus pinaster*, *Pinus pinea*, *Pinus radiata*, *Pinus sylvestris*, *Pinus uncinata*, *Populus x euramericana*, *Quercus canariensis*, *Quercus faginea*, *Quercus ilex*, *Quercus pyrenaica*, *Quercus robur*, *Quercus suber*

Available components:

Aboveground / belowground groups (summed automatically):

- "AGB" — total aboveground biomass
- "BGB" — total belowground biomass (roots)
- "all" or "tree" — total tree biomass (AGB + BGB)

Tree structural groups:

- "stem" — stem fraction(s)
- "branches" — all branch fractions combined
- "roots" — roots (equivalent to BGB)

Individual tree components (species availability varies):

- "stem" — stem wood
- "stem and thick branches" — stem together with branches > 7 cm
- "thick branches" — branches > 7 cm
- "medium branches" — branches 2–7 cm
- "small branches" — branches < 2 cm
- "leaves" — foliage (including needles)
- "roots" — coarse roots

Note that for this model, "tree" (or "all") is an independent regression equation fitted to total-tree data. It was **not** derived by summing the AGB and BGB equations. Consequently, there is a numerical discrepancy between the direct "tree" estimation and the sum of separate "AGB" and "BGB" estimations (e.g. for *Pinus sylvestris* at diameter = 20 cm and height = 10 m, the direct total is 89.1 kg, while AGB + BGB is 115.9 kg, a 24% difference).

Also, the following 6 species have no BGB/roots equations in this model: *Abies pinsapo*, *Erica arborea*, *Eucalyptus* spp., *Ilex canariensis*, *Laurus azorica*, *Myrica faya* (requesting "BGB" or "roots" will fail).

Users can check all available species and components in the [biomass_models](#) dataset provided by the library.

Value

A ModelBiomass object containing the configured model parameters and expressions.

See Also

[silv_predict_biomass\(\)](#), [biomass_models](#), [eq_biomass_diequez_aranda_2009\(\)](#), [eq_biomass_ruiz_peinado_2011\(\)](#), [eq_biomass_ruiz_peinado_2012\(\)](#), [eq_biomass_manrique_2017\(\)](#), [eq_biomass_menendez_2022\(\)](#), [eq_biomass_cudjoe_2024\(\)](#)

Examples

```
## Aboveground biomass for Pinus pinaster
eq_biomass_montero_2005("Pinus pinaster", "AGB")
```

eq_biomass_ruiz_peinado_2011

Biomass equations for Spanish softwood species

Description

Allometric equations adjusted for Spanish softwood species

Usage

```
eq_biomass_ruiz_peinado_2011(species, component = "stem", return_rmse = FALSE)
```

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species.
component	A character string specifying the tree component for biomass calculation (e.g., "tree", "stem", "branches"). 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.

Details

Supported species (10):

Abies alba, *Abies pinsapo*, *Juniperus thurifera*, *Pinus canariensis*, *Pinus halepensis*, *Pinus nigra*, *Pinus pinaster*, *Pinus pinea*, *Pinus sylvestris*, *Pinus uncinata*

Available components:

Aboveground / belowground groups (summed automatically):

- "AGB" — total aboveground biomass
- "BGB" — total belowground biomass (roots)
- "tree" — total tree biomass (AGB + BGB)

Tree structural groups:

- "stem" — stem wood
- "branches" — all branch fractions combined
- "roots" — roots (equivalent to BGB)

Individual tree components (species availability varies):

- "thick branches" — branches > 7 cm
- "thick and medium branches" — branches > 2 cm
- "medium branches" — branches 2–7 cm
- "small branches and leaves" — branches < 2 cm including leaves/needles

Note that *Abies pinsapo* does not have a separate BGB equation (requesting "BGB" or "roots" will fail, though "tree" still works via its pre-summed formula).

Users can check all available species and components in the [biomass_models](#) dataset provided by the library.

Value

A ModelBiomass object containing the configured model parameters and expressions.

See Also

[silv_predict_biomass\(\)](#), [biomass_models](#), [eq_biomass_montero_2005\(\)](#), [eq_biomass_dieguez_aranda_2009\(\)](#), [eq_biomass_ruiz_peinado_2012\(\)](#), [eq_biomass_manrique_2017\(\)](#), [eq_biomass_menendez_2022\(\)](#), [eq_biomass_cudjoe_2024\(\)](#)

Examples

```
## Aboveground biomass for Pinus pinaster
eq_biomass_ruiz_peinado_2011("Pinus pinaster", "AGB")
```

```
eq_biomass_ruiz_peinado_2012
```

Biomass equations for Spanish hardwood species

Description

Allometric equations adjusted for Spanish hardwood species

Usage

```
eq_biomass_ruiz_peinado_2012(species, component = "stem", return_rmse = FALSE)
```

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species.
component	A character string specifying the tree component for biomass calculation (e.g., "tree", "stem", "branches"). 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.

Details**Supported species (13):**

Alnus glutinosa, *Castanea sativa*, *Ceratonia siliqua*, *Eucalyptus globulus*, *Fagus sylvatica*, *Fraxinus angustifolia*, *Olea europaea*, *Populus x euramericana*, *Quercus canariensis*, *Quercus faginea*, *Quercus ilex*, *Quercus pyrenaica*, *Quercus suber*

Available components:

Aboveground / belowground groups (summed automatically):

- "AGB" — total aboveground biomass
- "BGB" — total belowground biomass (roots)
- "tree" — total tree biomass (AGB + BGB)

Tree structural groups:

- "stem" — stem wood
- "branches" — all branch fractions combined
- "roots" — roots (equivalent to BGB)

Individual tree components (species availability varies):

- "stem and thick branches" — stem together with branches > 7 cm
- "thick branches" — branches > 7 cm
- "thick and medium branches" — branches > 2 cm
- "medium branches" — branches 2–7 cm
- "small branches" — branches 0.5–2 cm
- "small branches and leaves" — branches < 2 cm including leaves
- "medium branches, small branches and leaves" — branches < 7 cm including leaves

Note that *Eucalyptus globulus* does not have a separate BGB equation (requesting "BGB" or "roots" will fail, though "tree" still works via its pre-summed formula).

Users can check all available species and components in the [biomass_models](#) dataset provided by the library.

Value

A ModelBiomass object containing the configured model parameters and expressions.

See Also

[silv_predict_biomass\(\)](#), [biomass_models](#), [eq_biomass_montero_2005\(\)](#), [eq_biomass_diegueza_aranda_2009\(\)](#), [eq_biomass_ruiz_peinado_2011\(\)](#), [eq_biomass_manrique_2017\(\)](#), [eq_biomass_menendez_2022\(\)](#), [eq_biomass_cudjoe_2024\(\)](#)

Examples

```
## Aboveground biomass for Quercus suber
eq_biomass_ruiz_peinado_2012("Quercus suber", "AGB")
```

eq_hd_vazquez_veloso_2025

Height-diameter equations from Vázquez-Veloso et al. (2025)

Description

This function is intended to be used in [silv_predict_height\(\)](#). It implements the h-d equations developed in Vázquez-Veloso et al. (2025). These equations have been developed using the Spanish National Forest Inventory, and therefore, they should only be applied within Spain. The model includes parameters for 91 tree species.

Usage

```
eq_hd_vazquez_veloso_2025(
  species,
  bioregion = "mediterranean",
  origin = "natural",
  mixture = "pure"
)
```

Arguments

species	A character string specifying the scientific name of the tree species. It can be a column name if all the species are included in this model. See Details for available species. If not specified, it takes the value "All the species", which corresponds to a generic model applicable to all species.
bioregion	The biogeographic region of the species. Available options are: mediterranean, atlantic, alpine, and macaronesian. If not specified, it takes the value mediterranean, which is the most common region in Spain. You can check the distribution of regions here: https://ars.els-cdn.com/content/image/1-s2.0-S037811272500489X-gr1.jpg
origin	The origin of the stand. Available options are: natural and plantation. If not specified, it takes the value natural, which is the most common origin in Spain.
mixture	The species available in the stand. Available options are: pure and mix. Consider the characteristics of the plot you are evaluating and not the entire forest, as the conditions of each stand are different. In this study, it was considered a stand to be mixed when the combined proportion of at least two species exceeds 90% of the plot's basal area, and the proportion of both species is greater than 15% of the total. It does not matter which species is accompanying or the proportion of mixing. If not specified, it takes the value pure, which is the most common condition in Spain.

Details

The model adjusts the species-specific coefficients using the selected bioregion, stand origin, and mixture type before returning the model object.

Value

A ModelHD object containing the configured model parameters and metadata.

References

Vázquez-Veloso, A., Yang, S.-I., Bullock, B.P., Bravo, F., 2025. One model to rule them all: A nationwide height–diameter model for 91 Spanish forest species. *Forest Ecology and Management* 595, 122981. <https://doi.org/10.1016/j.foreco.2025.122981>

See Also

[silv_predict_height\(\)](#)

Examples

```
model <- eq_hd_vazquez_veloso_2025("All the species")
silv_predict_height(25, model)
```

inventory_samples	<i>Forest inventory samples</i>
-------------------	---------------------------------

Description

Inventory data from Spanish National Forest Inventory

Usage

```
inventory_samples
```

Format

A tibble

lid_fcov	<i>Calculate Forest Fraction Cover from LiDAR Data</i>
----------	--

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	<i>LiDAR-derived Height Diversity Index (LHDI)</i>
----------	--

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

```
z <- c(0.5, 1.2, 1.8, 2.4, 3.1)
lid_lhdi(z)
```

plot	<i>Plot an object</i>
------	-----------------------

Description

Generic for plotting objects.

Usage

```
plot(x, ..., min_error = 0.01, max_error = 0.5)

## S7 method for class <silviculture::SampleSize>
plot(x, ..., min_error = 0.01, max_error = 0.5)
```

Arguments

<code>x</code>	Object to plot.
<code>...</code>	Other arguments passed to methods.
<code>min_error</code>	A numeric value specifying the minimum relative error to consider (default is 0.01).
<code>max_error</code>	A numeric value specifying the maximum relative error to consider (default is 0.5).

Value

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

<code>predict_carbon</code>	<i>Predict carbon content from biomass</i>
-----------------------------	--

Description

`silv_predict_carbon()` calculates the carbon content of a tree component based on its predicted biomass. It uses species- and component-specific carbon percentages from the package's internal `carbon_models` database.

`silv_predict_carbon_auto()` is a vectorized function that automatically selects the best available carbon model for each row based on a provided priority list. If the exact species is not found, it attempts a genus-level fallback (e.g., "Pinus spp.").

Usage

```
silv_predict_carbon(
  biomass,
  species,
  component,
  model = "montero-2005",
  quiet = FALSE
)

silv_predict_carbon_auto(
  biomass,
  species,
  component,
  priority = c("montero-2005", "dieguez-aranda-2009"),
  quiet = FALSE
)
```

Arguments

biomass	A numeric vector of predicted biomass (in kg).
species	A character vector of tree species (e.g., "Pinus pinaster").
component	A character string or vector specifying the tree component (e.g., "stem", "roots", "tree"). Must match the tree_component column in the carbon_models dataset. Note that some models do not provide carbon percentages for aggregate groups like "AGB".
model	A character string specifying the carbon model to use (e.g., "montero-2005").
quiet	Logical. If FALSE, informs the user about genus fallbacks or missing species.
priority	A character vector specifying the order of preference for carbon models. Default is c("montero-2005", "dieguez-aranda-2009").

Value

For `silv_predict_carbon()`: A numeric vector of predicted carbon (in kg). Returns NA if the species/component combination is not supported by the model.

For `silv_predict_carbon_auto()`: A `data.frame` with two columns:

- `carbon`: The predicted carbon (in kg).
- `carbon_model`: The model and species level used (e.g., "montero-2005" or "montero-2005 (genus fallback)").

Examples

```
# 1. Vector-based calculation: calculate stem carbon for Pinus pinaster
biomass_vector <- c(120.5, 230.1, 85.4)
species_vector <- c("Pinus pinaster", "Pinus pinaster", "Pinus pinaster")

carbon_vector <- silv_predict_carbon(
  biomass = biomass_vector,
  species = species_vector,
  component = "stem",
  model = "montero-2005"
)
print(carbon_vector)

# 2. Vector-based calculation with row-wise auto-selection & genus fallback
carbon_df <- silv_predict_carbon_auto(
  biomass = c(120.5, 95.0),
  species = c("Pinus pinaster", "Pinus radiata"), # Pinus radiata triggers genus fallback
  component = c("stem", "roots")
)
print(carbon_df)

# 3. Dataset-based tutorial: apply to a forest inventory data frame
inventory <- data.frame(
  tree_id = 1:3,
  species = c("Pinus pinaster", "Pinus sylvestris", "Quercus robur"),
  biomass_kg = c(150.0, 180.5, 220.0),
```

```

  component = c("stem", "roots", "stem")
)

# Apply row-wise auto-selection to the entire dataset
carbon_results <- silv_predict_carbon_auto(
  biomass  = inventory$biomass_kg,
  species  = inventory$species,
  component = inventory$component
)

# Combine and display results
inventory_with_carbon <- cbind(inventory, carbon_results)
print(inventory_with_carbon)

```

sdi_coefficients

SDI beta coefficients

Description

Specific beta coefficients for Reineke's Stand Density Index (SDI) per species, country, and region.

Usage

```
sdi_coefficients
```

Format

A tibble

article_id Character. Short identifier of the source article.

title Character. Full title of the source article.

doi_url Character. DOI URL of the source article.

country Character. Country where the study was conducted.

region Character. Region within the country.

species Character. Scientific name of the tree species.

beta Numeric. Beta coefficient for SDI calculation.

sdimax_models	<i>Maximum stand density index (SDImax) models</i>
---------------	--

Description

Coefficients for calculating maximum stand density index (SDImax) from Rodríguez de Prado (2020).

Usage

```
sdimax_models
```

Format

A tibble with 88 rows and 13 variables:

article_id Character. Identifier of the article.

title Character. Title of the article.

doi_url Character. DOI URL of the article.

country Character. Country where the study was conducted.

species Character. Tree species scientific name.

model_name Character. Name of the model/equation variant (e.g. "basic", "P1", "MXT3").

a0 Numeric. Coefficient a0.

a1 Numeric. Coefficient a1 (0 if not used/applicable).

b0 Numeric. Coefficient b0.

b1 Numeric. Coefficient b1 (0 if not used/applicable).

aic Numeric. Akaike Information Criterion.

pseudo_r2 Numeric. Pseudo R-squared value.

q_index Numeric. Q index value.

References

Rodríguez-de-Prado, M., et al. (2020). Potential climatic influence on maximum stand carrying capacity for 15 Mediterranean coniferous and broadleaf species. *Forest Ecology and Management*, 458, 117824.

silv_basal_area	<i>Calculates Basal Area</i>
-----------------	------------------------------

Description**[Deprecated]**

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

[Deprecated]

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"
)
```

<code>silv_density_hart</code>	<i>Hart or Hart-Becking spacing index</i>
--------------------------------	---

Description

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

Usage

```
silv_density_hart(h0, ntrees, which = c("hart", "hart-becking"))
```

Arguments

<code>h0</code>	Numeric vector with dominant height
<code>ntrees</code>	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees = NULL</code> , the function will assume that each diameter corresponds to only one tree
<code>which</code>	A character with the name of the index (either <code>hart</code> or <code>hart-becking</code>). 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_stand_dominant_height(diameter, height),
    ntrees  = n(),
    .by     = plot_id
  ) |>
  ## calculate number of trees per hectare
  mutate(ntrees_ha = silv_density_ntrees_ha(ntrees, plot_size = 14.1)) |>
  mutate(spacing = silv_density_hart(h0, ntrees_ha))
```

`silv_density_ntrees_ha`

Calculates number of trees per hectare

Description

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

Usage

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

Arguments

<code>ntrees</code>	A numeric vector representing the number of trees in a sampling plot
<code>plot_size</code>	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
<code>plot_shape</code>	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_density_ntrees_ha(n, plot_size = 10)
  )

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

silv_density_sdi	<i>Calculates the Stand Density Index</i>
------------------	---

Description

The Stand Density Index (SDI) is the relationship between the average tree size and density of trees per hectare.

Usage

```
silv_density_sdi(ntrees, dg, beta = 1.605)
```

Arguments

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
dg	Numeric vector of quadratic mean diameters
beta	The Stand Density Index exponent (default is 1.605).

Details

The SDI has different interpretations depending on the species, location, and also the management type (even-aged, uneven-aged...). The value of maximum SDI must be determined from the literature and used carefully. The beta exponent allows adjustments for different species or mixed stands.

Value

A numeric vector representing the absolute SDI.

References

Reineke, L. H. (1933). Perfecting a stand-density index for even-aged forests. Journal of Agricultural Research, 46(7), 627-638. URL: <https://research.fs.usda.gov/download/treesearch/60134.pdf>

Examples

```
## calculate SDI for a Pinus sylvestris stand (beta = 1.605)
silv_density_sdi(ntrees = 800, dg = 23.4)

## calculate SDI with custom beta
silv_density_sdi(ntrees = 800, dg = 23.4, beta = 1.7)
```

`silv_density_sdi_auto` *Predict Stand Density Index automatically*

Usage

```
silv_density_sdi_auto(
  ntrees,
  dg,
  species,
  country = NULL,
  region = NULL,
  classify = FALSE,
  climatic_model = NULL,
  clim_value = NULL,
  quiet = FALSE
)
```

Arguments

<code>ntrees</code>	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees</code> = <code>NULL</code> , the function will assume that each diameter corresponds to only one tree
<code>dg</code>	Numeric vector of quadratic mean diameters
<code>species</code>	A character string or vector of tree species (e.g., "Pinus sylvestris").
<code>country</code>	A character string or vector of the country (e.g., "Spain"). Defaults to <code>NULL</code> (no country specified).
<code>region</code>	A character string or vector of the region (e.g., "Castilla y León"). Defaults to <code>NULL</code> (no region specified).
<code>classify</code>	A logical value indicating whether to automatically calculate <code>SDImax</code> and classify the values (default is <code>FALSE</code>).

climatic_model	Character. The specific climate-dependent model name (e.g. "P1", "MXT3"). Passed to <code>silv_density_sdimax</code> when <code>classify = TRUE</code> .
clim_value	Numeric vector. Values of the climatic variable corresponding to the selected climate model. Passed to <code>silv_density_sdimax</code> when <code>classify = TRUE</code> .
quiet	Logical. If FALSE, informs the user about fallbacks to genus or default models.

Value

A data.frame with the columns:

- sdi: The computed absolute Stand Density Index.
- beta: The beta exponent used for the calculation.
- sdi_model: The model used for beta exponent.
- sdimax: (If `classify = TRUE`) The maximum SDI for the species.
- sdi_class: (If `classify = TRUE`) The density classification.
- quietLogical. If FALSE, informs the user about fallbacks to genus or default models.

A data.frame with three columns:

- sdi: The computed absolute Stand Density Index.
- beta: The beta exponent used for the calculation.
- sdi_model: The model used (e.g., "del-rio-2006 (Spain, Castilla y León)", "reineke-1933 (-1.605)", etc.).

`silv_density_sdi_auto()` is a vectorized function that automatically selects the best available Stand Density Index exponent (beta) for each row based on a provided species, country, and region from the internal `sdi_coefficients` database.

If an exact species, country, and region match is not found, the function falls back to a country-wide species model (`region = "all"`), then searches other countries, then falls back to a genus-level fallback (e.g., "Pinus spp."), and finally to the default SDI exponent (`beta = -1.605` from Reineke 1933).

```
# Calculate SDI with automatic selection silv_density_sdi_auto( ntrees = 800, dg = 23.4, species =
"Pinus sylvestris", region = "Castilla y León" )
```

```
# With automatic classification silv_density_sdi_auto( ntrees = 800, dg = 23.4, species = "Pinus
sylvestris", classify = TRUE # Fallback to default silv_density_sdi_auto( ntrees = 800, dg = 23.4,
species = "Unknown species" )
```

`silv_density_sdi_class`*Classifies the Stand Density Index*

Description

Classifies the Stand Density Index (SDI) into density classes or calculates the relative SDI percentage based on USDA thresholds.

Usage

```
silv_density_sdi_class(sdi, max_sdi, classify = TRUE)
```

Arguments

<code>sdi</code>	A numeric vector representing the Stand Density Index.
<code>max_sdi</code>	A numeric vector representing the maximum SDI for the species/site.
<code>classify</code>	A logical value indicating whether to classify the values into density classes (default is TRUE). If FALSE, it returns the relative SDI as a percentage.

Details

The option `classify = TRUE` will use the `max_sdi` value to classify the SDI into four competitive and growth conditions: low density (<24%), moderate density (24-35%), high density (34-55%), and extremely high density (>55%).

Value

A character vector with the density classes if `classify = TRUE`, or a numeric vector with the relative SDI percentage if `classify = FALSE`.

References

USDA Forest Service. (n.d.). Stand Density Index. https://www.fs.usda.gov/Internet/FSE_DOCUMENTS/stelprdb5270993.

Examples

```
## calculate SDI for a Pinus sylvestris stand (max 990)
sdi_val <- silv_density_sdi(ntrees = 800, dg = 23.4)

## check base classification
silv_density_sdi_class(sdi = sdi_val, max_sdi = 990)

## get relative SDI percentage
silv_density_sdi_class(sdi = sdi_val, max_sdi = 990, classify = FALSE)
```

`silv_density_sdimax` *Calculates the Maximum Stand Density Index (SDImax)*

Description

The Maximum Stand Density Index (SDImax) represents the maximum stand carrying capacity, calculated using coefficients from Rodríguez de Prado (2020) by default.

Usage

```
silv_density_sdimax(
  species,
  model = "rodriguez-prado-2020",
  climatic_model = NULL,
  clim_value = NULL
)
```

Arguments

<code>species</code>	Character vector. Scientific names of the tree species.
<code>model</code>	Character. The source article or model database (default is "rodriguez-prado-2020").
<code>climatic_model</code>	Character. The specific climate-dependent model name (e.g. "P1", "MXT3"). Required if <code>clim_value</code> is provided, and must not be "basic".
<code>clim_value</code>	Numeric vector. Values of the climatic variable corresponding to the selected climate model. If NULL (default), the reference model ("basic") is calculated.

Details

If `clim_value` is NULL, the function computes the reference SDImax (SDImaxREF) based on the "basic" model parameters:

$$SDImaxREF = \exp(a_0 + b_0 * \log(25.4))$$

If `clim_value` is provided, a climate-dependent model must be specified in `climatic_model`, and the climate-dependent SDImax is calculated as:

$$SDImax(Clim) = \exp((a_0 + a_1 * \log(clim_value)) + (b_0 + b_1 * clim_value) * \log(25.4))$$

Value

A numeric vector representing the SDImax for each species.

References

Rodríguez-de-Prado, M., et al. (2020). Potential climatic influence on maximum stand carrying capacity for 15 Mediterranean coniferous and broadleaf species. *Forest Ecology and Management*, 458, 117824.

Examples

```
## Calculate reference SDImax for Pinus sylvestris
silv_density_sdimax("Pinus sylvestris")

## Calculate climate-dependent SDImax for Pinus canariensis using model P1
silv_density_sdimax("Pinus canariensis", climatic_model = "P1", clim_value = 400)
```

silv_diametric_class	<i>Classify diameters in classes</i>
----------------------	--------------------------------------

Description

[Deprecated]
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]
return_intervals	[dim1,dim2). If FALSE, the intervals are (dim1,dim2]: R:dim1,%20dim2)%60.%20If%20FALSE 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**[Deprecated]**

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

Usage

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

Arguments

<code>diameter</code>	Numeric vector with diameter classes
<code>height</code>	Numeric vector with averaged heights by diameter class
<code>ntrees</code>	Optional. Numeric vector with number of trees per hectare. Use this argument when you have aggregated data by diametric classes (see details).
<code>which</code>	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 tree. 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	<i>Calculates Lorey's Height</i>
-------------------	----------------------------------

Description

```
#' @description [Deprecated]
```

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

Tree's mean height weighted by basal area

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	<i>Calculates number of trees per hectare</i>
----------------	---

Description

[Deprecated] 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_predict_biomass *Calculate Tree Biomass*

Description

Computes the biomass of a tree species using species-specific allometric equations (in kg). Currently, only equations for Spain are available.

Usage

```
silv_predict_biomass(
  diameter = NULL,
  height = NULL,
  model,
  ntrees = NULL,
  rcd = NULL,
  bp = NULL,
  quiet = FALSE
)
```

Arguments

diameter	A numeric vector of tree diameters at breast height (in cm).
height	A numeric vector of tree heights (in m).
model	A function. A function with the structure <code>eq_biomass_*</code> () with additional arguments depending on the model used.
ntrees	An optional numeric value indicating the number of trees in this diameter-height class. Defaults to 1 if NULL.
rcd	An optional numeric vector of root collar diameters (in cm). Required for eq_biomass_menendez_2022 , which uses root collar diameter instead of diameter at breast height. Defaults to diameter if NULL.
bp	An optional numeric vector of biomass packing values (in m ³ \$). Required for a subset of species in eq_biomass_menendez_2022 (e.g. <i>Pinus halepensis</i> , <i>Pinus nigra</i> , <i>Quercus suber</i> , <i>Evergreen broadleaves</i>).
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:

- [eq_biomass_ruiz_peinado_2011\(\)](#): Developed for softwood species in Spain.
- [eq_biomass_ruiz_peinado_2012\(\)](#): Developed for hardwood species in Spain.
- [eq_biomass_montero_2005\(\)](#): Developed for 35 Spanish species.
- [eq_biomass_dieguez_aranda_2009\(\)](#): Developed for 7 Galician species.
- [eq_biomass_manrique_2017\(\)](#): Developed for *Quercus petraea* and *Quercus pyrenaica*.
- [eq_biomass_menendez_2022\(\)](#): Developed for young plantations (< 30 years) of 18 Spanish species. Uses rcd (root collar diameter) instead of diameter; some species require bp (biomass packing) instead of rcd.
- [eq_biomass_cudjoe_2024\(\)](#): Developed for *Pinus sylvestris* and *Quercus petraea* in Castille and León, 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 with predicted tree biomass (kg).

See Also

[biomass_models](#), [eq_biomass_montero_2005\(\)](#), [eq_biomass_dieguez_aranda_2009\(\)](#), [eq_biomass_ruiz_peinado_2011\(\)](#), [eq_biomass_ruiz_peinado_2012\(\)](#), [eq_biomass_manrique_2017\(\)](#), [eq_biomass_menendez_2022\(\)](#), [eq_biomass_cudjoe_2024\(\)](#)

Examples

```
# 1. Vector-based calculation: predict stem/tree biomass for Pinus pinaster
model <- eq_biomass_ruiz_peinado_2011("Pinus pinaster")
predicted_biomass <- silv_predict_biomass(
  diameter = c(20, 25, 30),
  height   = c(15, 17, 18),
  model    = model
)
print(predicted_biomass)

# 2. Dataset-based tutorial: apply to a forest inventory data frame
inventory <- data.frame(
  tree_id = 1:3,
  species = c("Pinus pinaster", "Pinus pinaster", "Pinus pinaster"),
  dbh_cm  = c(18.5, 22.1, 29.4),
  height_m = c(14.0, 16.5, 19.0)
)

# Apply prediction and append a new column to the dataset
inventory$biomass_kg <- silv_predict_biomass(
```

```

    diameter = inventory$dbh_cm,
    height    = inventory$height_m,
    model     = model
  )
  print(inventory)

# 3. Young plantation example (Menendez 2022 model) using rcd and bp
# Menendez 2022 equations use root collar diameter (rcd) and/or biomass packing (bp)
model_menendez <- eq_biomass_menendez_2022("Pinus pinaster")
predicted_young_pinaster <- silv_predict_biomass(
  rcd    = c(5.2, 7.1, 9.4), # Root collar diameter in cm
  height = c(2.1, 3.2, 4.5), # Height in m
  model  = model_menendez
)
print(predicted_young_pinaster)

# For Pinus halepensis, Menendez 2022 requires biomass packing (bp)
model_halepensis <- eq_biomass_menendez_2022("Pinus halepensis")
predicted_young_halepensis <- silv_predict_biomass(
  bp    = c(0.005, 0.012), # Biomass packing in m3
  model = model_halepensis
)
print(predicted_young_halepensis)

```

silv_predict_biomass_auto

Automatically Predict Biomass Using the Best Available Model

Description

Evaluates vectors of tree species, diameters, and heights, and automatically selects the best available allometric model based on a specified priority. If tree height is not provided, is NA, or is 0, the function automatically falls back to the [eq_biomass_montero_2005](#) model.

Usage

```

silv_predict_biomass_auto(
  species,
  diameter,
  height = NULL,
  component = "tree",
  ntrees = NULL,
  rcd = NULL,
  bp = NULL,
  priority = c("ruiz-peinado-2011", "ruiz-peinado-2012", "montero-2005",
    "dieguez-aranda-2009", "manrique-2017", "menendez-2022", "cudjoe-2024"),
  quiet = FALSE
)

```

Arguments

species	A character vector specifying the scientific names of the tree species.
diameter	A numeric vector of tree diameters at breast height (in cm).
height	An optional numeric vector of tree heights (in m). Defaults to NULL.
component	A character string specifying the tree component for biomass calculation (e.g., "tree", "stem", "branches"). Defaults to "tree".
ntrees	An optional numeric vector indicating the number of trees in each class. Defaults to NULL (equivalent to 1 tree per entry).
rcd	An optional numeric vector of root collar diameters (in cm). Required for young plantation equations (eq_biomass_menendez_2022).
bp	An optional numeric vector of biomass packing values (in m ^{\$^3\$}). Required for some species in young plantation equations.
priority	A character vector specifying the priority order for model selection. Defaults to c("ruiz-peinado-2011", "ruiz-peinado-2012", "montero-2005", "dieguez-aranda-2009", "manrique-2017", "menendez-2022", "cudjoe-2024").
quiet	A logical value. If TRUE, suppresses any informational messages and warnings.

Value

A data.frame containing two columns:

- biomass: Numeric vector of predicted biomass values (in kg).
- biomass_model: Character vector specifying the model ID used.

Examples

```
# 1. Vector-based calculation: automatic model selection for mixed species
species_vec <- c("Pinus pinaster", "Quercus petraea")
d_vec <- c(20, 25)
h_vec <- c(12, 14)
auto_results <- silv_predict_biomass_auto(species_vec, d_vec, h_vec)
print(auto_results)

# 2. Vector-based calculation: fallback to Montero 2005 when height is missing
fallback_results <- silv_predict_biomass_auto(species_vec, d_vec, height = NULL)
print(fallback_results)

# 3. Dataset-based tutorial: apply to a mixed-species forest inventory data frame
inventory <- data.frame(
  tree_id = 1:3,
  species = c("Pinus pinaster", "Quercus petraea", "Pinus sylvestris"),
  dbh_cm = c(22.5, 18.0, 31.2),
  height_m = c(15.0, 11.5, 18.0)
)

# Run auto-selection and bind the results directly
biomass_data <- silv_predict_biomass_auto(
  species = inventory$species,
```

```
diameter = inventory$dbh_cm,
height   = inventory$height_m
)

inventory_with_biomass <- cbind(inventory, biomass_data)
print(inventory_with_biomass)
```

```
silv_predict_biomass_components
```

Predict All Individual Biomass Components for Trees

Description

Predicts all available individual biomass components (e.g., stem, bark, branches, roots) for the given trees in a single call, returning a wide data frame.

Usage

```
silv_predict_biomass_components(
  species,
  diameter,
  height = NULL,
  model_fn,
  ntrees = NULL,
  rcd = NULL,
  bp = NULL,
  quiet = TRUE
)
```

Arguments

species	A character vector specifying the scientific names of the tree species.
diameter	A numeric vector of tree diameters at breast height (in cm).
height	An optional numeric vector of tree heights (in m). Defaults to NULL.
model_fn	A function or character string. The constructor function of the model (e.g., <code>eq_biomass_ruiz_peinado_2011</code>) or the model ID string (e.g., <code>"ruiz-peinado-2011"</code>).
ntrees	An optional numeric vector indicating the number of trees in each class. Defaults to NULL.
rcd	An optional numeric vector of root collar diameters (in cm).
bp	An optional numeric vector of biomass packing values (in m ³).
quiet	A logical value. If TRUE, suppresses any informational messages.

Value

A data.frame with the columns species, diameter (cm), height (m, if provided), and one additional numeric column (in kg) for each individual biomass component available for the selected species and model.

Examples

```
# 1. Vector-based calculation: predict all components for Pinus pinaster
comp_results <- silv_predict_biomass_components(
  species = c("Pinus pinaster", "Pinus pinaster"),
  diameter = c(20, 25),
  height = c(12, 15),
  model_fn = eq_biomass_ruiz_peinado_2011
)
print(comp_results)

# 2. Dataset-based tutorial: apply to a forest inventory data frame
inventory <- data.frame(
  tree_id = 1:3,
  species = c("Pinus pinaster", "Pinus pinaster", "Pinus pinaster"),
  dbh_cm = c(18.5, 22.1, 29.4),
  height_m = c(14.0, 16.5, 19.0)
)

# Predict components for the entire dataset
comp_df <- silv_predict_biomass_components(
  species = inventory$species,
  diameter = inventory$dbh_cm,
  height = inventory$height_m,
  model_fn = "ruiz-peinado-2011"
)

# Combine and display results (excluding repeated identifier columns)
inventory_with_components <- cbind(
  inventory,
  comp_df[, -(1:3)]
)
print(inventory_with_components)
```

silv_predict_height	<i>Estimates tree height (m) from DBH (cm)</i>
---------------------	--

Description

Estimates total tree height (m) using height-diameter (h-d) equations. Currently, only models developed for Spain are available.

Usage

```
silv_predict_height(
  diameter,
  model = eq_hd_vazquez_veloso_2025("All the species"),
  quiet = FALSE
)
```

Arguments

diameter	Numeric vector with diameters in cm (DBH).
model	A ModelHD object. An object configured via the eq_hd_*() family of functions. Defaults to eq_hd_vazquez_veloso_2025("All the species").
quiet	A logical value. If TRUE, suppresses any informational messages.

Details

The function estimates total tree height (in meters) using diameter at breast height (in centimeters), and may require additional information depending on the specific model. See each model's documentation for details.

Value

A numeric vector with predicted heights (m).

References

References for the models available:

- `eq_hd_vazquez_veloso_2025()`: Vázquez-Veloso, A., Yang, S.-I., Bullock, B.P., Bravo, F., 2025. One model to rule them all: A nationwide height–diameter model for 91 Spanish forest species. *Forest Ecology and Management* 595, 122981. <https://doi.org/10.1016/j.foreco.2025.122981>

See Also

`eq_hd_vazquez_veloso_2025()`

Examples

```
# 1. Predict height using the default model (Vázquez-Veloso 2025 generic model)
predicted_heights_default <- silv_predict_height(diameter = c(20, 25, 30))
print(predicted_heights_default)

# 2. Load the S7 ModelHD object for Pinus pinaster
model <- eq_hd_vazquez_veloso_2025("Pinus pinaster")

# 3. Vector-based calculation: predict tree heights from diameters
diameters <- c(20, 25, 30)
predicted_heights <- silv_predict_height(diameter = diameters, model = model)
print(predicted_heights)

# 4. Dataset-based tutorial: apply to a forest inventory data frame
inventory <- data.frame(
  tree_id = 1:3,
  species = c("Pinus pinaster", "Pinus pinaster", "Pinus pinaster"),
  dbh_cm = c(18.5, 22.1, 29.4)
)

# Apply prediction and append a new column to the dataset
inventory$height_m <- silv_predict_height(
```

```
diameter = inventory$dbh_cm,
model    = model
)
print(inventory)
```

silv_predict_snfi_volume

Predict tree volume using Spanish National Forest Inventory equations

Description

Computes four volume metrics (in dm^3) for one or more trees using the allometric equations from the 3rd and 4th Spanish National Forest Inventory (SNFI3/SNFI4), as documented by MITECO (Anexo 19).

Usage

```
silv_predict_snfi_volume(
  province,
  species,
  dbh = NULL,
  h = NULL,
  dnm = NULL,
  quality = "default",
  quiet = FALSE
)
```

Arguments

province	A character or numeric vector. Province name (e.g. "Cantabria") or province code (e.g. 39). See silv_snfi_provinces() .
species	A character or numeric vector. Species scientific name (e.g. "Pinus radiata") or SNFI species code (e.g. 28). See silv_snfi_species() .
dbh	A numeric vector of diameter at breast height in cm .
h	A numeric vector of tree heights in m . Optional for some models (e.g. SNFI3 model 14), but required by most VCC equations.
dnm	A numeric vector of mean plot diameter in cm . Required only when the IAVC equation uses model 13.
quality	A character or numeric vector. Quality/shape class (1-6). Use "default" (the default) to select the first available class.
quiet	A logical value. If TRUE, suppresses citation messages. Defaults to FALSE.

Details

Model selection:

For each province, SNFI4 data are preferred. When a province is not covered by SNFI4 (11 provinces), the function falls back to SNFI3 automatically. SNFI3 also provides a supplementary file for species not present in the standard provincial tables.

Equation models:

Component	Models used
VCC	1, 11
VSC	7
IAVC	8, 13, 14, 16, 17, 18, 19, 20, 21, 25
VLE	10, 12

Full model equations are available in Anexo 19 of the MITECO SIG documentation (see References).

Units:

Inputs: dbh and dnm in **cm** (converted to mm internally), h in **m**. All outputs in **dm³**.

Value

A data.frame with one row per input tree and five columns:

vcc Merchantable volume **with** bark (dm³).

vsc Merchantable volume **without** bark (dm³).

iavc Annual volume increment with bark (dm³).

vle Coarse firewood volume (dm³).

snfi_version Dataset used: "SNFI4" or "SNFI3".

Any component for which parameters are missing or a required input is NA returns NA for that column, without stopping execution.

References

MITECO. 4th Spanish National Forest Inventory - SIG database codes. https://www.miteco.gob.es/content/dam/miteco/es/biodiversidad/temas/inventarios-nacionales/documentador_sig_tcm30-536622.pdf

MITECO. 3rd Spanish National Forest Inventory - SIG database codes. https://www.miteco.gob.es/content/dam/miteco/es/biodiversidad/servicios/banco-datos-naturaleza/documentador_bdcampo_ifn3_tcm30-282240.pdf

See Also

[silv_snfi_provinces\(\)](#), [silv_snfi_species\(\)](#), [silv_tree_volume\(\)](#)

Examples

```

# 1. Vector-based calculation: single tree (Pinus radiata in Alava, province 1, code 28)
single_tree <- silv_predict_snfi_volume(
  province = 1,
  species  = 28,
  dbh      = 20,
  h        = 15,
  dnm      = 23
)
print(single_tree)

# 2. Vector-based calculation: multiple trees with province and species by name
multi_trees <- silv_predict_snfi_volume(
  province = c("Araba/Álava", "Cantabria"),
  species  = c("Pinus radiata", "Pinus radiata"),
  dbh      = c(20, 25),
  h        = c(15, 18),
  dnm      = c(23, 26)
)
print(multi_trees)

# 3. Dataset-based tutorial: apply to a forest inventory data frame
inventory <- data.frame(
  tree_id   = 1:3,
  province  = c("Cantabria", "Cantabria", "Araba/Álava"),
  species   = c("Pinus radiata", "Pinus radiata", "Pinus sylvestris"),
  dbh_cm    = c(20.5, 28.0, 18.2),
  height_m  = c(15.2, 18.5, 12.0)
)

# Apply the SNFI volume estimation functions row-wise across the dataset
volume_results <- silv_predict_snfi_volume(
  province = inventory$province,
  species   = inventory$species,
  dbh       = inventory$dbh_cm,
  h         = inventory$height_m
)

# Combine and display results
inventory_with_volume <- cbind(inventory, volume_results)
print(inventory_with_volume)

```

silv_sample_size

*Calculates sample size for a random sampling inventory***Description****[Deprecated]**

Usage

```

silv_sample_size(
  x,
  plot_size,
  total_area,
  method = "random",
  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
method	sampling method. Available options are random
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 suppressed

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_sample_size_simple

Calculates sample size for a simple random sampling (SRS)

Description

Calculates the sample size needed for a SRS inventory, estimated from pilot inventory data.

Usage

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

Arguments

x	vector of the variable measured in the pilot inventory (e.g. basal area, volume)
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 relative error
conf_level	confidence level
max_iter	maximum number of iteration to find the plot size
quiet	if TRUE, messages will be suppressed

Details

Sample size is very important to be optimized, since a small sample size will entail a higher error, while a huge sample size will entail higher costs. The SRS is typically used for random sampling, although it might be used also for regular sampling. The number of samples is calculated using the expression:

$$n \geq \frac{t^2 \cdot CV^2}{\epsilon^2 + \frac{t^2 \cdot CV^2}{N}}$$

Where:

- **t**: the value of student's t for given sample size of the pilot inventory
- **CV**: the coefficient of variation of x
- **ε**: the relative error (max_error)
- **N**: the size of the pilot inventory

x is a variable measured in a pilot inventory. Let's say we measure forest variables in 10 pilot plots, aiming at basal area measurement so we have to measure only the DBH. After some calculations, we will have the basal area per hectare in each of the 10 plots. The sample size is then calculated from the variation of these values and the error that we will allow.

Value

SimpleSampleSize 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_simple(
  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_sample_size_stratified
```

Calculates sample size for a stratified sampling

Description

Calculates the sample size needed for a stratified inventory, estimated from pilot inventory data.

Usage

```
silv_sample_size_stratified(
  data,
  x,
  strata,
  total_area,
  plot_size,
  method = "optimal",
  cost = NA,
  max_error = 0.05,
  conf_level = 0.95,
  max_iter = 1000,
  currency = "EUR",
  quiet = FALSE
)
```

Arguments

data	a data.frame of pilot inventory data
x	name of the variable in data that was measured (e.g. basal area, volume)
strata	name of the variable in data with the name of the stratum
total_area	name of the variable in data with the area of the stratum
plot_size	a numeric vector of length one with plot size in squared meters
method	a character vector of length one with the id of the method. Available options are optimal, cost, and prop. See details
cost	name of the variable in data with the average cost of measuring one plot of the stratum. Used with method = 'cost' for sample size, and for message output in other methods
max_error	maximum allowed relative error
conf_level	confidence level
max_iter	maximum number of iteration to find the plot size
currency	currency to be shown in console output when using method = 'cost'
quiet	if TRUE, messages will be suppressed

Details

Stratified Sampling calculates the number of plots to be inventoried in different strata. For instance, you might have *Pinus sylvestris* and *Pinus pinaster* plots in the same forest, and you might want to get the optimal number of plots for field inventory of each stratum, for a given maximum relative error (e.g. 5%), and with a certain level of confidence (e.g. 95%). Of course, the area of *P. sylvestris* will be different than the area occupied by *P. pinaster*. For instance, the total area of *P. sylvestris* could be 100 ha, while the area of *P. pinaster* could be 200 ha. Therefore, you need to create a pilot inventory and measure a variable such as basal area maybe in 5 pilot plots of *P. sylvestris* and 7 pilot plots of *P. pinaster*. With that data collected, you can use three stratified sample size methods:

- **Optimal Allocation with Constant Cost:** using method = 'optimal'. The sampling units are distributed within the different strata taking into account the size (e.g. 100 ha vs 200 ha) and the heterogeneity (e.g. differences in basal area). It minimizes the number of sampling units.

$$n = \frac{t_{n-m}^2 \cdot (\sum_{j=1}^m P_j \cdot s_j)^2}{\epsilon^2 + \frac{t_{n-m}^2 \cdot \sum_{j=1}^m P_j \cdot s_j^2}{N}}$$

- **Optimal Allocation with Variable Cost:** using method = 'cost'. This method needs to know the cost of a sampling unit in each strata. It will minimize the cost of the inventory, taking into account the size, the heterogeneity, and the cost of the sampling unit of the strata.

$$n = \frac{t_{n-m}^2 \cdot (\sum_{j=1}^m P_j \cdot s_j \cdot \sqrt{c_j}) \cdot (\sum_{j=1}^m \frac{P_j \cdot s_j}{\sqrt{c_j}})}{\epsilon^2 + \frac{t_{n-m}^2 \cdot \sum_{j=1}^m P_j \cdot s_j^2}{N}}$$

- **Proportional Allocation:** using method = 'prop'. The sampling units are distributed proportional to the size of the strata. In the example, 33% of the estimated sampling units will be allocated to *P. sylvestris* and 66% to *P. pinaster*.

$$n = \frac{t_{n-m}^2 \cdot \sum_{j=1}^{j=m} P_j \cdot s_j^2}{\epsilon^2 + \frac{t_{n-m}^2 \cdot \sum_{j=1}^{j=m} P_j \cdot s_j^2}{N}}$$

Where:

- **n**: estimated sample size
- **t**: the value of student's t
- P_j : proportion of pilot plots of j^{th} strata
- s_j : standard deviation of x
- s_j^2 : variance of x
- **N**: population size (number of plots of plot_size that fit in total_area)
- ϵ : maximum allowed absolute error. Calculated from x and max_error
- **N**: the size of the pilot inventory

Value

S7 StratifiedSampleSize object with:

- **results**: data.frame with the main results by stratum
- **strata_error**: data.frame with maximum absolute error $\mp$ C.I (max_abs_error, x_min, x_max), and the estimator of the typical error $\mp$ C.I (sampling error, x_ci_lo, x_ci_hi)
- **sampling_error**: data.frame with the maximum absolute error $\mp$ C.I (max_abs_error, x_min, x_max), and the typical sampling error of the weighted mean $\mp$ C.I (sampling error, x_ci_lo, x_ci_hi)
- **sampling_opts**: list with function options

Examples

```
## read pilot inventory fictitious data
data_path <- system.file("extdata/pilot_inventory.csv", package = "silviculture")
inventory_tbl <- read.csv(data_path)

## calculate sample size
sample_size_list <- silv_sample_size_stratified(
  data = inventory_tbl,
  x = basal_area,
  strata = stratum,
  total_area = area,
  method = "optimal",
  cost = cost,
  plot_size = 100,
  conf_level = .95,
  max_error = .05
)
```

silv_site_factor	<i>Calculates the Site Factor (SF)</i>
------------------	--

Description

The site factor is a variable used in growth equations based on the dominant height and dominant dbh.

Usage

```
silv_site_factor(species, d0, h0)
```

Arguments

species	Character vector. Scientific names of the tree species.
d0	Numeric vector. Dominant diameter.
h0	Numeric vector. Dominant height.

Value

A numeric vector representing the Site Factor. NA for unsupported species.

References

Aguirre, A., et al. (2022).

Examples

```
silv_site_factor(species = "Pinus sylvestris", d0 = 25, h0 = 15)
```

silv_snfi_provinces	<i>List provinces supported by SNFI volume equations</i>
---------------------	--

Description

Returns a data frame with province codes, names, and which SNFI dataset (SNFI4 or SNFI3 fall-back) is used for each.

Usage

```
silv_snfi_provinces()
```

Value

A data.frame with columns province_id, province_name, and snfi_version.

Examples

```
silv_snfi_provinces()
```

silv_snfi_species	<i>List species supported by SNFI volume equations</i>
-------------------	--

Description

Returns a data frame of species codes and names from the SNFI4 species catalogue.

Usage

```
silv_snfi_species(snfi_version = "SNFI4")
```

Arguments

snfi_version Character. Either "SNFI4" (default) or "SNFI3".

Value

A data.frame with columns species_code and species_name.

Examples

```
head(silv_snfi_species())
head(silv_snfi_species("SNFI3"))
```

silv_spacing_index	<i>Hart or Hart-Becking spacing index</i>
--------------------	---

Description**[Deprecated]**

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 quadratic mean diameter (QMD)*

Description

[Deprecated]

Usage

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

Arguments

<code>diameter</code>	Numeric vector of diameters or diameter classes
<code>ntrees</code>	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees = NULL</code> , the function will assume that each diameter corresponds to only one tree. Therefore, the QMD 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_stand_basal_area` *Calculates Basal Area*

Description

Calculates Basal Area in square meters.

Usage

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

Arguments

<code>diameter</code>	Numeric vector of diameters or diameter classes
<code>ntrees</code>	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees</code> = <code>NULL</code> , the function will assume that each diameter corresponds to only one tree
<code>units</code>	The units of the diameter (one of mm, cm, dm, or m)

Details

The function uses the next formula:

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

where G is the basal area in m^2 , and D is the diameter in cm. If $ntrees$ is the number of trees per hectare, then the result will be m^2/ha . It is recommended to use the squared mean diameter calculated with `silv_stand_qmean_diameter()`.

Note that if $ntrees = NULL$, the output of the function will be exactly the same as in `silv_tree_basal_area()`.

Value

A numeric vector

Examples

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

`silv_stand_dominant_diameter`

Calculates the dominant diameter

Description

Calculates the dominant diameter using Assman and Friedrich method, or Weise method

Usage

```
silv_stand_dominant_diameter(
  diameter,
  ntrees = NULL,
  which = c("assman", "weise")
)
```

Arguments

<code>diameter</code>	Numeric vector of diameters or diameter classes
<code>ntrees</code>	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees = NULL</code> , the function will assume that each diameter corresponds to only one tree
<code>which</code>	The method to calculate the dominant diameter (see details)

Details

The dominant diameter D_0 is the mean diameter of the 100 thickest trees per hectare. Therefore, diameter and ntrees should be vectors of the same length.

- **Assman**: calculates the D_0 as the mean diameter of the 100 thickest trees per hectare
- **Weise**: calculates the D_0 as the quadratic mean diameter of the 20% thickest trees per hectare

Value

A numeric vector

Examples

```
## calculate d0 for inventory data grouped by plot_id and species
library(dplyr)
inventory_samples |>
mutate(dclass = silv_tree_dclass(diameter)) |>
  summarise(
    height = mean(height, na.rm = TRUE),
    ntrees = n(),
    .by = c(plot_id, species, dclass)
  ) |>
mutate(
  ntrees_ha = silv_density_ntrees_ha(ntrees, plot_size = 10),
  d0 = silv_stand_dominant_diameter(dclass, ntrees_ha),
  .by = c(plot_id, species)
)
```

`silv_stand_dominant_height`

Calculates the dominant height

Description

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

Usage

```
silv_stand_dominant_height(
  diameter,
  height,
  ntrees = NULL,
  which = c("assman", "hart")
)
```

Arguments

diameter	Numeric vector of diameters or diameter classes
height	Numeric vector of tree heights
ntrees	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees = NULL</code> , the function will assume that each diameter corresponds to only one tree
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 tree. 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_tree_dclass(diameter)) |>
  summarise(
    height = mean(height, na.rm = TRUE),
    ntrees = n(),
    .by = c(plot_id, species, dclass)
  ) |>
mutate(
  ntrees_ha = silv_density_ntrees_ha(ntrees, plot_size = 10),
  h0 = silv_stand_dominant_height(dclass, height, ntrees_ha),
  .by = c(plot_id, species)
)
```

`silv_stand_lorey_height`
Calculates Lorey's Height

Description

Tree's mean height weighted by basal area

Usage

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

Arguments

<code>height</code>	Numeric vector of tree heights
<code>g</code>	Numeric vector of basal areas
<code>ntrees</code>	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees</code> = <code>NULL</code> , the function will assume that each diameter corresponds to only one tree

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_tree_basal_area(diameter)) |>
  summarise(
    lh = silv_stand_lorey_height(height, g),
    .by = c(plot_id, species)
  )
```

`silv_stand_qmean_diameter`*Calculates the quadratic mean diameter (QMD)*

Description

Calculates the quadratic mean diameter (QMD)

Usage

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

Arguments

<code>diameter</code>	Numeric vector of diameters or diameter classes
<code>ntrees</code>	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees</code> = <code>NULL</code> , the function will assume that each diameter corresponds to only one tree

Value

A numeric vector

Examples

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

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

 silv_stand_slenderness

Calculates the Stand Slenderness Index

Description

The Stand Slenderness Index is defined as the ratio of height to diameter, multiplied by 100. It can be applied to mean height and quadratic mean diameter, or dominant height and dominant diameter.

Usage

```
silv_stand_slenderness(height, diameter)
```

Arguments

height	Numeric vector of tree heights
diameter	Numeric vector of diameters or diameter classes

Details

The formula used is:

$$Slenderness = \frac{H}{D} \times 100$$

where H is the height in meters and D is the diameter in centimeters.

Value

A numeric vector representing the slenderness index.

Examples

```
## Calculate slenderness using mean height and quadratic mean diameter
silv_stand_slenderness(height = 18, diameter = 25)
```

 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 = NULL,
  .groups = NULL,
  plot_shape = c("circular", "rectangular", "snfi"),
  dmin = 7.5,
  dmax = NULL,
  class_length = 5,
  include_lowest = TRUE,
  which_h0 = "assman",
  which_spacing = "hart",
  volume = NULL,
  volume_units = "dm3",
  biomass = NULL,
  carbon = NULL,
  predict_volume = FALSE,
  province = NULL,
  species = NULL,
  predict_biomass = FALSE,
  biomass_component = "tree",
  predict_carbon = FALSE
)

```

Arguments

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

volume	Unquoted column name with individual tree volume, optional.
volume_units	Character. Units of the individual tree volume ("dm3" or "m3"). Default is "dm3".
biomass	Unquoted column name with individual tree biomass (kg), optional.
carbon	Unquoted column name with individual tree carbon (kg), optional.
predict_volume	Logical. If TRUE, predicts tree volume using silv_predict_snfi_volume() (default: FALSE).
province	Unquoted column name or scalar string/integer with the province code/name.
species	Unquoted column name with tree species identifier.
predict_biomass	Logical. If TRUE, predicts tree biomass using silv_predict_biomass_auto() (default: FALSE).
biomass_component	Character. Tree component to predict for biomass (default: "tree").
predict_carbon	Logical. If TRUE, predicts tree carbon using silv_predict_carbon_auto() (default: FALSE).

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

Volume is reported at stand level in m^3/ha (v_ha). Biomass and carbon are reported at stand level in t/ha (w_ha, c_ha).

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_summary_mortality

Summarize plot data by mortality status

Description

Calculates the number of trees, basal area, and optional volume, biomass, and carbon per hectare for live and dead trees in a plot.

Usage

```
silv_summary_mortality(
  data,
  plot_id,
  dead,
  expan = NULL,
  g = NULL,
  diameter = NULL,
  volume = NULL,
  volume_units = "dm3",
  biomass = NULL,
  carbon = NULL
)
```

Arguments

data	A data frame or tibble of tree-level data
plot_id	Unquoted column name with the plot identifier
dead	Unquoted column name indicating if the tree is dead (e.g., TRUE/FALSE or 1/0)
expan	Unquoted column name with the expansion factor (trees/ha). If NULL, it is assumed each row represents 1 tree/ha.
g	Unquoted column name with the basal area per tree (m ² /ha). If NULL, diameter must be provided.
diameter	Unquoted column name with the diameter (cm). Used to calculate basal area if g is NULL.
volume	Unquoted column name with individual tree volume, optional.
volume_units	Character. Units of the individual tree volume ("dm3" or "m3"). Default is "dm3".
biomass	Unquoted column name with individual tree biomass (kg), optional.
carbon	Unquoted column name with individual tree carbon (kg), optional.

Value

A tibble with plot-level mortality summaries.

Examples

```
library(dplyr)
inventory_samples |>
  mutate(expan = silv_density_ntrees_ha(1, 10)) |>
  mutate(is_dead = sample(c(TRUE, FALSE), n(), replace = TRUE, prob = c(0.1, 0.9))) |>
  silv_summary_mortality(plot_id, is_dead, expan, diameter = diameter)
```

`silv_summary_species` *Summarize plot data by species*

Description

Calculates the number of trees, basal area, and optional volume, biomass, and carbon per hectare for each species in a plot, and optionally provides the top species formatted into columns.

Usage

```
silv_summary_species(
  data,
  plot_id,
  species,
  expan = NULL,
  g = NULL,
  diameter = NULL,
  volume = NULL,
  volume_units = "dm3",
  biomass = NULL,
  carbon = NULL,
  top_n = 3
)
```

Arguments

<code>data</code>	A data frame or tibble of tree-level data
<code>plot_id</code>	Unquoted column name with the plot identifier
<code>species</code>	Unquoted column name with the species identifier
<code>expan</code>	Unquoted column name with the expansion factor (trees/ha). If NULL, it is assumed each row represents 1 tree/ha.
<code>g</code>	Unquoted column name with the basal area per tree (m ² /ha). If NULL, diameter must be provided.
<code>diameter</code>	Unquoted column name with the diameter (cm). Used to calculate basal area if g is NULL.
<code>volume</code>	Unquoted column name with individual tree volume, optional.
<code>volume_units</code>	Character. Units of the individual tree volume ("dm3" or "m3"). Default is "dm3".

biomass	Unquoted column name with individual tree biomass (kg), optional.
carbon	Unquoted column name with individual tree carbon (kg), optional.
top_n	Number of top species to pivot into columns (default: 3). If 0, no pivoting is performed.

Value

A tibble with species-level summaries.

Examples

```
library(dplyr)
inventory_samples |>
  mutate(expan = silv_density_ntrees_ha(1, 10)) |>
  silv_summary_species(plot_id, species, expan, diameter = diameter, top_n = 3)
```

`silv_treatment_thinning`

Calculate Forestry Thinning Schemes

Description

Calculates thinning schemes for forest management by selecting trees to extract based on specified criteria. Supports both thinning from below (removing smaller trees) and thinning from above (removing larger trees) approaches.

Usage

```
silv_treatment_thinning(
  data,
  var,
  diameter,
  ntrees,
  thinning = c("below", "above"),
  perc = 0.3,
  .groups = NULL
)
```

Arguments

data	A data frame, or <code>silviculture::Inventory</code> object. See details.
var	A variable used for calculating the thinning. Typically used variables basal area, number of trees, or volume
diameter	Numeric vector of diameters or diameter classes
ntrees	Numeric vector with number of trees of the diameter class per hectare. If <code>ntrees = NULL</code> , the function will assume that each diameter corresponds to only one tree

thinning	Character string specifying the thinning type. Available options are below and above
perc	Numeric value between 0 and 1 specifying the percentage of var to extract
.groups	A character vector with variables to group by (e.g. plot id, tree species, etc). Ignored when using a <code>silviculture::Inventory</code> object

Details

This function implements common silvicultural thinning practices:

Thinning from below: Removes trees with the lowest values of the specified variable. This approach typically removes suppressed, damaged, or poor-quality trees, mimicking natural mortality processes.

Thinning from above: Removes trees with the highest values of the specified variable. This approach harvests the most valuable trees while leaving smaller trees to continue growing.

The function calculates which trees to extract based on the ranking of the specified variable and the desired thinning percentage. When grouping variables are provided, thinning is calculated separately for each group.

Using a `silviculture::Inventory` object The result of `silv_summary()` can be used as the data argument. If so, the `.groups` will be taken from this object, and it will keep the previous data in a new S7 object.

Value

A `silviculture::Thinning` object with three items:

- **data:** the input data with two new columns
- **group_metrics:** it will include the data from the `silviculture::Inventory` object
- **thinning_opts:** options used for S7 methods

See Also

[silv_summary\(\)](#)

Examples

```
# Get summary of inventory data
inventory <- inventory_samples |>
  silv_summary(
    diameter = diameter,
    height   = height,
    plot_size = 25,
    .groups  = c('plot_id', 'species')
  )

## Thinning from below removing 30% of trees based on basal area
silv_treatment_thinning(
  data    = inventory,
  var     = g_ha,
```

```

    diameter = dclass,
    ntrees   = ntrees_ha,
    thinning = "below",
    perc     = 0.3
  )

  ## Thinning from above removing 20% of trees based on basal area
  silv_treatment_thinning(
    data      = inventory,
    var       = g_ha,
    diameter  = dclass,
    ntrees    = ntrees_ha,
    thinning  = "above",
    perc      = 0.2
  )

```

silv_tree_bal

Calculate Basal Area Larger (BAL)

Description

Calculates the Basal Area Larger (BAL) for each tree in each plot.

Usage

```

silv_tree_bal(
  data,
  plot_id,
  tree_id,
  diameter = NULL,
  expansion_factor = NULL,
  basal_area_ha = NULL,
  units = "cm"
)

```

Arguments

<code>data</code>	A data frame containing tree measurements.
<code>plot_id</code>	Unquoted column name with the plot identifier.
<code>tree_id</code>	Unquoted column name with the tree identifier.
<code>diameter</code>	Unquoted column name with the DBH (optional).
<code>expansion_factor</code>	Unquoted column name with the expansion factor (optional).
<code>basal_area_ha</code>	Unquoted column name with the basal area per hectare. If provided, <code>diameter</code> and <code>expansion_factor</code> are ignored.
<code>units</code>	The units of the diameter.

Value

A numeric vector with the BAL for each tree.

Examples

```
## Not run:
inventory_samples |>
  dplyr::mutate(bal = silv_tree_bal(
    inventory_samples, plot_id, tree_id, diameter, exp_factor))

## End(Not run)
```

silv_tree_bas	<i>Calculate Basal Area Smaller (BAS)</i>
---------------	---

Description

Calculates the Basal Area Smaller (BAS) for each tree in each plot.

Usage

```
silv_tree_bas(
  data,
  plot_id,
  tree_id,
  diameter = NULL,
  expansion_factor = NULL,
  basal_area_ha = NULL,
  units = "cm"
)
```

Arguments

data	A data frame containing tree measurements.
plot_id	Unquoted column name with the plot identifier.
tree_id	Unquoted column name with the tree identifier.
diameter	Unquoted column name with the DBH (optional).
expansion_factor	Unquoted column name with the expansion factor (optional).
basal_area_ha	Unquoted column name with the basal area per hectare. If provided, diameter and expansion_factor are ignored.
units	The units of the diameter.

Value

A numeric vector with the BAS for each tree.

silv_tree_basal_area	<i>Calculates Basal Area</i>
----------------------	------------------------------

Description

Calculates Basal Area in square meters.

Usage

```
silv_tree_basal_area(diameter, units = "cm")
```

Arguments

diameter	Numeric vector of diameters or diameter classes
units	The units of the diameter (one of mm, cm, dm, 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 of one tree, and D is the diameter in cm.

If you want to calculate the basal area for a group of trees (e.g. per hectares), please use [silv_stand_basal_area\(\)](#)

Value

A numeric vector

See Also

[silv_stand_basal_area\(\)](#)

Examples

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

`silv_tree_basal_area_ha`*Calculate Tree Basal Area per Hectare*

Description

Calculates the tree basal area expanded to hectare.

Usage

```
silv_tree_basal_area_ha(  
  diameter = NULL,  
  basal_area = NULL,  
  expansion_factor,  
  units = "cm"  
)
```

Arguments

diameter	A numeric vector of tree DBH. Optional if basal_area is provided.
basal_area	A numeric vector of tree basal area in m2. Optional if diameter is provided.
expansion_factor	A numeric vector representing the tree expansion factor.
units	The units of the diameter (one of mm, cm, dm, or m).

Value

A numeric vector with the basal area per hectare (m2/ha).

Examples

```
silv_tree_basal_area_ha(diameter = 20, expansion_factor = 127)
```

`silv_tree_circumference`*Calculate Tree Circumference*

Description

Calculate Tree Circumference

Usage

```
silv_tree_circumference(diameter, units = "cm")
```

Arguments

diameter	Numeric vector of diameters or diameter classes
units	The units of the diameter (one of mm, cm, dm, or m).

Value

A numeric vector with the circumference in the same units.

Examples

```
silv_tree_circumference(20)
```

`silv_tree_coordinates` *Calculate Relative and Absolute Tree Coordinates*

Description

Calculate Relative and Absolute Tree Coordinates

Usage

```
silv_tree_coordinates(
  distance,
  bearing,
  x_center = NULL,
  y_center = NULL,
  dist_units = "m",
  bearing_units = "grad"
)
```

Arguments

distance	Distance from plot center.
bearing	Bearing from plot center.
x_center	Plot center X coordinate (optional).
y_center	Plot center Y coordinate (optional).
dist_units	Units of the distance (one of mm, cm, dm, m).
bearing_units	Units of the bearing (degree, grad, radian).

Value

A data frame with relative coordinates (`x_rel`, `y_rel`) and, if plot centers are provided, absolute coordinates (`x_abs`, `y_abs`).

Examples

```
silv_tree_coordinates(distance = 10, bearing = 100)
```

`silv_tree_crown_ratio` *Calculate Tree Crown Ratio*

Description

Calculate Tree Crown Ratio

Usage

```
silv_tree_crown_ratio(height, height_base_crown)
```

Arguments

`height` Total tree height.
`height_base_crown`
 Height to the base of the live crown.

Value

A numeric vector with the crown ratio.

Examples

```
silv_tree_crown_ratio(height = 20, height_base_crown = 8)
```

`silv_tree_dclass` *Classify diameters in classes*

Description

Classifies the measured diameters into classes of a specified length

Usage

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

Arguments

diameter	Numeric vector of diameters or diameter classes
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]
return_intervals	[dim1,dim2). If FALSE, the intervals are (dim1,dim2]: R:dim1,%20dim2)%60.%20If%20FALSE 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_tree_dclass(diameter))
```

`silv_tree_expansion_factor`

Calculate Tree Expansion Factor

Description

Calculates the tree expansion factor according to the plot type (fixed area or SNFI concentric plots).

Usage

```
silv_tree_expansion_factor(
  type = c("fixed_area", "snfi"),
  plot_area = NULL,
  diameter = NULL,
  d_units = "cm",
  a_units = "m2"
)
```

Arguments

type	Plot type, either "fixed_area" or "snfi".
plot_area	Plot area (required for "fixed_area").
diameter	Tree DBH (required for "snfi").
d_units	Units of the diameter.
a_units	Units of the plot area (one of m2 or ha).

Value

A numeric vector with the expansion factor.

Examples

```
silv_tree_expansion_factor(type = "fixed_area", plot_area = 500)
silv_tree_expansion_factor(type = "snfi", diameter = 15)
```

silv_tree_mean_dbh	<i>Calculate Mean DBH or derive it from Perimeter</i>
--------------------	---

Description

Calculates the mean diameter at breast height (DBH) from two perpendicular measurements or derives it from the perimeter.

Usage

```
silv_tree_mean_dbh(dbh1 = NULL, dbh2 = NULL, perimeter = NULL, units = "cm")
```

Arguments

dbh1	A numeric vector with the first DBH measurement.
dbh2	A numeric vector with the second DBH measurement.
perimeter	A numeric vector with the perimeter measurement.
units	The units of the inputs (one of mm, cm, dm, or m).

Value

A numeric vector with the mean DBH.

Examples

```
silv_tree_mean_dbh(dbh1 = 20, dbh2 = 22)
silv_tree_mean_dbh(perimeter = 65)
```

`silv_tree_slenderness` *Calculate Tree Slenderness*

Description

Calculate Tree Slenderness

Usage

```
silv_tree_slenderness(diameter, height, d_units = "cm", h_units = "m")
```

Arguments

<code>diameter</code>	A numeric vector of tree DBH.
<code>height</code>	A numeric vector of tree height.
<code>d_units</code>	Units of the diameter.
<code>h_units</code>	Units of the height.

Value

A numeric vector with the tree slenderness.

Examples

```
silv_tree_slenderness(diameter = 20, height = 15)
```

`silv_tree_summary` *Tree-level inventory summary*

Description

Computes individual tree-level dendrometric metrics, diametric classes, expansion factors, competition indices, and optional volume, biomass, and carbon predictions.

Usage

```
silv_tree_summary(  
  data,  
  diameter,  
  height = NULL,  
  plot_id = NULL,  
  species = NULL,  
  expan = NULL,  
  plot_size = NULL,  
  plot_shape = c("circular", "rectangular", "snfi"),
```

```

dmin = 7.5,
dmax = NULL,
class_length = 5,
include_lowest = TRUE,
compute_bal = FALSE,
predict_volume = FALSE,
province = NULL,
predict_biomass = FALSE,
biomass_component = "tree",
predict_carbon = FALSE
)

```

Arguments

<code>data</code>	A data frame or tibble with tree-level records.
<code>diameter</code>	Unquoted column name with the tree diameter (in cm).
<code>height</code>	Unquoted column name with the tree height (in m), optional.
<code>plot_id</code>	Unquoted column name with the plot identifier, optional.
<code>species</code>	Unquoted column name with the tree species identifier/name, optional.
<code>expan</code>	Unquoted column name with the expansion factor (trees/ha), optional.
<code>plot_size</code>	Numeric. Size of the sampling plot (radius in meters if circular, area in m ² if rectangular).
<code>plot_shape</code>	Character. Shape of the sampling plot ("circular" or "rectangular"). Default is "circular".
<code>dmin</code>	Numeric. Minimum diameter for diametric classes (default: 7.5).
<code>dmax</code>	Numeric. Maximum diameter for diametric classes (default: NULL).
<code>class_length</code>	Numeric. Width of diametric classes (default: 5).
<code>include_lowest</code>	Logical. Whether to include lowest bound in classes (default: TRUE).
<code>compute_bal</code>	Logical. If TRUE and <code>plot_id</code> is supplied, computes BAL and BAS (default: FALSE).
<code>predict_volume</code>	Logical. If TRUE, predicts SNFI volume (vcc, vsc, iavc) using silv_predict_snfi_volume() (default: FALSE).
<code>province</code>	Unquoted column name or scalar string/integer with the province code/name for volume prediction.
<code>predict_biomass</code>	Logical. If TRUE, predicts tree biomass using silv_predict_biomass_auto() (default: FALSE).
<code>biomass_component</code>	Character. Tree component to predict for biomass (default: "tree").
<code>predict_carbon</code>	Logical. If TRUE, predicts tree carbon content using silv_predict_carbon_auto() (default: FALSE).

Value

The original data frame enriched with computed tree-level metrics.

Examples

```
library(dplyr)
silv_tree_summary(
  data      = inventory_samples,
  diameter  = diameter,
  height    = height,
  plot_id   = plot_id,
  species   = species,
  plot_size = 10,
  compute_bal = TRUE
)
```

silv_tree_volume	<i>Calculate Tree Volume</i>
------------------	------------------------------

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

silv_volume	<i>Calculate Tree Volume</i>
-------------	------------------------------

Description

[Deprecated]

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")
```

site_factor_models	<i>Site Factor models</i>
--------------------	---------------------------

Description

Coefficients for calculating the Site Factor (SF) based on Aguirre et al. (2022). The site factor is a variable that can be used in growth equations and is based on the dominant height and dominant dbh.

Usage

```
site_factor_models
```

Format

A tibble with 24 rows and 7 variables:

species Character. Scientific name of the tree species.
species_code Numeric. Species numeric code.
d_ref Numeric. Reference diameter (Dref).
model Character. Model shape and expanded parameter.
param_a Numeric. Parameter a.
param_b Numeric. Parameter b.
param_c Numeric. Parameter c.

References

Aguirre, A. et al. (2022).

snfi3_volume_coefficients
SNFI3 tree volume coefficients

Description

Allometric equation coefficients for tree volume and increment from the 3rd Spanish National Forest Inventory (SNFI3/IFN3), as documented by MITECO.

Usage

snfi3_volume_coefficients

Format

A tibble with 37,730 rows and 16 variables:

Codigo_provincia Integer. Province numeric code.

Nombre_provincia Character. Province name.

Codigo_especie Integer. Species numeric code.

Especie Character. Species scientific name.

Parametro Character. Volume component: VCC (merchantable volume with bark), VSC (merchantable volume without bark), IAVC (annual volume increment with bark), or VLE (coarse firewood volume).

F.c. Integer. Tree quality class (calidad del árbol), from 1 (healthy and straight tree) to 6 (dead tree).

Par_esp Character. Special parameter metadata.

Modelo Integer. Equation model index.

a, b, c, d, p, q, r Numeric. Equation coefficients (NA where not applicable).

R2 Numeric. Coefficient of determination of the model.

References

MITECO. 3rd Spanish National Forest Inventory - SIG database codes. https://www.miteco.gob.es/content/dam/miteco/es/biodiversidad/servicios/banco-datos-naturaleza/documentador_bdcampo_ifn3_tcm30-282240.pdf

snfi4_volume_coefficients
SNFI4 tree volume coefficients

Description

Allometric equation coefficients for tree volume and increment from the 4th Spanish National Forest Inventory (SNFI4/IFN4), as documented by MITECO.

Usage

snfi4_volume_coefficients

Format

A tibble with 20,925 rows and 15 variables:

Codigo_provincia Integer. Province numeric code.

Nombre_provincia Character. Province name.

Codigo_especie Integer. Species numeric code.

Especie Character. Species scientific name.

Parametro Character. Volume component: VCC (merchantable volume with bark), VSC (merchantable volume without bark), IAVC (annual volume increment with bark), or VLE (coarse firewood volume).

F.c. Integer. Tree quality class (calidad del árbol), from 1 (healthy and straight tree) to 6 (dead tree).

Modelo Integer. Equation model index.

a, b, c, d, p, q, r Numeric. Equation coefficients (NA where not applicable).

D.n.m. Character. Mean plot diameter metadata (diámetro normal medio), required for IAVC equation model 13.

References

MITECO. 4th Spanish National Forest Inventory - SIG database codes. https://www.miteco.gob.es/content/dam/miteco/es/biodiversidad/temas/inventarios-nacionales/documentador_sig_tcm30-536622.pdf

Index

* datasets

- biomass_models, [3](#)
- carbon_models, [4](#)
- inventory_samples, [17](#)
- sdi_coefficients, [21](#)
- sdimax_models, [22](#)
- site_factor_models, [79](#)
- snfi3_volume_coefficients, [80](#)
- snfi4_volume_coefficients, [81](#)

biomass_models, [3](#), [5–7](#), [9](#), [10](#), [12](#), [13](#), [15](#), [25](#), [37](#)

carbon_models, [4](#), [20](#)

eq_biomass_cudjoe_2024, [5](#)

eq_biomass_cudjoe_2024(), [7](#), [9](#), [10](#), [12](#), [13](#), [15](#), [37](#)

eq_biomass_dieguez_aranda_2009, [6](#)

eq_biomass_dieguez_aranda_2009(), [6](#), [9](#), [10](#), [12](#), [13](#), [15](#), [37](#)

eq_biomass_manrique_2017, [8](#)

eq_biomass_manrique_2017(), [6](#), [7](#), [10](#), [12](#), [13](#), [15](#), [37](#)

eq_biomass_menendez_2022, [9](#), [36](#), [39](#)

eq_biomass_menendez_2022(), [6](#), [7](#), [9](#), [12](#), [13](#), [15](#), [37](#)

eq_biomass_montero_2005, [11](#), [38](#)

eq_biomass_montero_2005(), [6](#), [7](#), [9](#), [10](#), [13](#), [15](#), [37](#)

eq_biomass_ruiz_peinado_2011, [12](#)

eq_biomass_ruiz_peinado_2011(), [6](#), [7](#), [9](#), [10](#), [12](#), [15](#), [37](#)

eq_biomass_ruiz_peinado_2012, [14](#)

eq_biomass_ruiz_peinado_2012(), [6](#), [7](#), [9](#), [10](#), [12](#), [13](#), [37](#)

eq_hd_vazquez_veloso_2025, [15](#)

eq_hd_vazquez_veloso_2025(), [42](#)

inventory_samples, [17](#)

lid_fcov, [17](#)

lid_lhdi, [18](#)

plot, [18](#)

plot, silviculture::SampleSize-method (plot), [18](#)

predict_carbon, [19](#)

sdi_coefficients, [21](#)

sdimax_models, [22](#)

silv_basal_area, [23](#)

silv_biomass, [24](#)

silv_density_hart, [25](#)

silv_density_hart(), [61](#)

silv_density_ntrees_ha, [26](#)

silv_density_ntrees_ha(), [61](#)

silv_density_sdi, [27](#)

silv_density_sdi_auto, [28](#)

silv_density_sdi_class, [30](#)

silv_density_sdimax, [29](#), [31](#)

silv_diametric_class, [32](#)

silv_dominant_height, [33](#)

silv_lorey_height, [34](#)

silv_ntrees_ha, [35](#)

silv_predict_biomass, [36](#)

silv_predict_biomass(), [6](#), [7](#), [9](#), [10](#), [12](#), [13](#), [15](#)

silv_predict_biomass_auto, [38](#)

silv_predict_biomass_auto(), [62](#), [76](#)

silv_predict_biomass_components, [40](#)

silv_predict_carbon (predict_carbon), [19](#)

silv_predict_carbon_auto (predict_carbon), [19](#)

silv_predict_carbon_auto(), [62](#), [76](#)

silv_predict_height, [41](#)

silv_predict_height(), [15](#), [16](#)

silv_predict_snfi_volume, [43](#)

silv_predict_snfi_volume(), [62](#), [76](#)

silv_sample_size, [45](#)

silv_sample_size_simple, [47](#)

silv_sample_size_stratified, 48
silv_site_factor, 51
silv_snfi_provinces, 51
silv_snfi_provinces(), 43, 44
silv_snfi_species, 52
silv_snfi_species(), 43, 44
silv_spacing_index, 52
silv_sqrmean_diameter, 23, 53
silv_stand_basal_area, 54
silv_stand_basal_area(), 69
silv_stand_dominant_diameter, 55
silv_stand_dominant_height, 56
silv_stand_dominant_height(), 61
silv_stand_lorey_height, 58
silv_stand_qmean_diameter, 59
silv_stand_qmean_diameter(), 55
silv_stand_slenderness, 60
silv_summary, 60
silv_summary(), 66
silv_summary_mortality, 63
silv_summary_species, 64
silv_treatment_thinning, 65
silv_tree_bal, 67
silv_tree_bas, 68
silv_tree_basal_area, 69
silv_tree_basal_area(), 55
silv_tree_basal_area_ha, 70
silv_tree_circumference, 70
silv_tree_coordinates, 71
silv_tree_crown_ratio, 72
silv_tree_dclass, 72
silv_tree_expansion_factor, 73
silv_tree_mean_dbh, 74
silv_tree_slenderness, 75
silv_tree_summary, 75
silv_tree_volume, 77
silv_tree_volume(), 44
silv_volume, 78
site_factor_models, 79
snfi3_volume_coefficients, 80
snfi4_volume_coefficients, 81