Package 'dendrometry'

April 24, 2025

Type Package

Date 2025-04-24

Title Forest Estimations and Dendrometric Computations

Version 0.0.3

Description Computation of dendrometric and structural parameters from forest inventory data. The objective is to provide an user-friendly R package for researchers, ecologists, foresters, statisticians, loggers and others persons who deal with forest inventory data. Useful conversion of angle value from degree to radian, conversion from angle to slope (in percentage) and their reciprocals as well as principal angle determination are also included. Position and dispersion parameters usually found in forest studies are implemented. The package contains Fibonacci series, its extensions and the Golden Number computation. Useful references are Arcadius Y. J. Akossou, Soufianou Arzouma, Eloi Y. Attakpa, Noël H. Fonton and Kouami Kokou (2013) <doi:10.3390/d5010099> and W. Bonou, R. Glele Kakaï, A.E. Assogbadjo, H.N. Fonton, B. Sinsin (2009) <doi:10.1016/j.foreco.2009.05.032> .

License GPL-3

Depends R (>= 3.5.0)

VignetteBuilder knitr

Suggests knitr, rmarkdown

Imports graphics, ForestFit

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Config/usethis/last-upkeep 2025-04-23

NeedsCompilation no

Author Narcisse Yehouenou [aut, cre],
Information and Communication Technology for you ONG (ICT4U-ONG) [fnd]

Maintainer Narcisse Yehouenou <narcisstar211@gmail.com>

Repository CRAN

Date/Publication 2025-04-24 21:20:02 UTC

Contents

adjWeibull *Adjust (fit) three-parameter Weibull distribution*

Description

Adjust (fit) three-parameter Weibull distribution

Usage

```
adjWeibull(x, amplitude = 10, shape = 2, plot = TRUE, main = NULL,
title.col = "black", mid = TRUE, line.col = "blue", legendPos = "topright",
lowLim = NULL, ymax = NULL, bg = "aliceblue", method = "mps", cex.axis = 0.6,
cex.lab = 0.8, las = 1, xlab = "Diameter class (cm)",
ylab = "Relative frequency (%)", ...)
```

Arguments

x	numeric, vector of observations.
amplitude	numeric, amplitude of classes.
shape	numeric, optional initial values of shape for starting the iterative procedures such as Newton-Raphson.
plot	logical. Should plot?
main	character, overall title for the plot.
title.col, line.col	the color to be used for the overall title and the plot line plot respectively. Default are "blue" and "black".
mid	logical. Should the line stop at the first and last classes middle?
legendPos	character, keyword which is accepted by xy.coords. To be used to position the legend. Default is "topright".
lowLim, ymax	numeric, xlim and ylim lowest values.
bg	the color to be used for the background of the legend box.
method	used method for estimating the three-parameter Weibull distribution. See fitWeibull for details.
cex.axis, cex.lab, las	graphical parameters. See par.
xlab, ylab	title for the x and y axis.
...	additional arguments to pass through plot.

Examples

```
set.seed(2)
d <- rweibull(85, shape = 1, scale = 30) + 5
res <- adjWeibull(d,
  amplitude = 10, mid = FALSE, shape = 3, ymax = 30,
  main = "Weibull adjustment", line.col = "red", legendPos = "right",
  method = "mps"
)
res
```

`angle2slope` *Angle - Slope conversion and Principal Measure determination*

Description

Conversion of angle to slope values and reciprocally.
`angle2slope` converts angle to slope values.
`slope2angle` converts slope to angle values.
`principal` determines the principal measure of an angle value. Principal measure ranges from -pi to pi for radian unit while it ranges from -180 to 180 for degree unit.

Usage

```
angle2slope(angle, angleUnit = c("deg", "rad"))

slope2angle(slope, angleUnit = c("deg", "rad"))

principal(angle, angleUnit = c("deg", "rad"))
```

Arguments

`angle`	numeric, vector of angle to be converted to slope.
`angleUnit`	character, unit of `angle`. For `slope2angle`, the desired unit for the returned angle value. For `principal`, both the angle input and output unit. Either `deg` or `rad`. Default is `deg`.
`slope`	numeric, vector of slope to be converted to angle.

Value

Object of class `angle`.
`angle2slope` returns vector of slope values while `slope2angle` and `principal` return vector of angle values in unit specified in `angle` argument.

Note

Use `principal` in position computations, not distance computations.

See Also

deg and rad.

Examples

```
angle2slope(10)
angle2slope(angle = 45)
angle2slope(angle = pi / 4, angleUnit = "rad")
angle2slope(1.047198, "rad")
angle2slope(seq(0.2, 1.5, .4), angleUnit = "rad") #'

slope2angle(100)
slope2angle(100, "rad")
round(pi / 4, 2)

slope2angle(17.6327)
slope2angle(angle2slope(30))

principal(303)
principal(23 * pi / 8, "rad")
principal(7 * pi / 4, angleUnit = "rad")
deg(principal(7 * pi / 4, angleUnit = "rad"))
principal(7 * 45)
```

---

barkFactor *Bark factor The bark factor (k) is computed for trees in order to assess the importance of the valuable wood in the overall volume of a tree (Husch et al., 1982):*

Description

Bark factor The bark factor (k) is computed for trees in order to assess the importance of the valuable wood in the overall volume of a tree (Husch et al., 1982):

Usage

```
barkFactor(dbh, thickness)
```

Arguments

dbh	numeric, diameter over bark of the individual trees.
thickness	numeric, bark thickness measured on individual trees.

References

Husch, B., Miller, C., Beers, T., 1982. Forest mensuration. Ronald Press Company, London, pp. 1 – 410.

basal *The basal area of stands*

Description

The basal area is the cross sectional area of the bole or stem of a tree at breast height.

Usage

```
basal(dbh, area, k = 100)
```

Arguments

dbh	numeric, vector of diameter.
area	numeric, area of a plot (see details for unit).
k	numeric, used to convert diameter unit. Default is 100 (see details)

Details

If area is expressed in ha and dbh expressed in cm, the basal area unit is cm\u00b2/ha when k = 1. In order to convert centimeter (cm) to meter (m) for dbh, set k = 100. Because 1m = 100 cm. Then, basal area unit will be m\u00b2/ha.

If dbh is in meter (m), and area in in hectare (ha), setting k = 1 returns basal area in m\u00b2/ha.

If dbh is in feet, and area in acre, setting k = 1 returns basal area in ft\u00b2/ac.

If dbh is in inch, and area in acre, setting k = 12 returns basal area in feet\u00b2/acres (ft\u00b2/ac).

Value

A vector of basal area of stands.

basalContribution *Basal area contribution*

Description

The basal area contribution (in per cent) is defined as the part of a given species trees in the overall basal area of all trees in an area.

Usage

```
basalContribution(basal)
```

Arguments

basal	numeric, basal area per species.

basal_i *Individual Basal Area and DBH (diameter)*

Description

basal_i computes the basal area of a tree stem (individual), the area of a circle of diameter dbh. basal2dbh computes the dbh (diameter) based on the basal area.

Usage

```
basal_i(dbh, circum = NULL)

basal2dbh(basal)
```

Arguments

dbh	numeric, vector of diameter.
circum	numeric, vector of circumference. Is used only when dbh is not given.
basal	numeric, individual basal area.

Details

If circum is given, dbh is not used.

Value

basal_i returns individual basal area while basal2dbh returns DBH.

Examples

```
basal_i(dbh = 10)
basal_i(circum = 31.41)
basal2dbh(78.53982)
```

blackman *Index of Blackman*

Description

Index of Blackman

Usage

```
blackman(density)
```

Arguments

density	numeric, vector of the density.

Value

Index of Blackman.

---

dbh *Diameter (DBH) and Circumference*

Description

DBH computes diameter (at breast height) based on circumference (at breast height).
circum computes circumference (at breast height) based on diameter (at breast height).
They are based on circle diameter and perimeter formulas.

Usage

```
dbh(circum)

circum(dbh)
```

Arguments

circum	numeric, vector of circumference.
dbh	numeric, vector of diameter.

Value

dbh, returns diameter and circum, returns circumference.

See Also

See also height for tree height.

Examples

```
perimeter <- seq(30, 60, 1.4)
diameter <- dbh(perimeter)
circum(diameter)
```

decrease *The decrease coefficient*

Description

This coefficient expresses the ratio between the diameter (or circumference) at mid-height of the bole and the diameter (or circumference) measured at breast height.

Usage

```
decrease(middle, breast)
```

Arguments

middle	numeric, the diameter or circumference at middle height.
breast	numeric, the diameter or circumference at breast height.

Details

Both `middle` and `breast` arguments should be of the same type (either diameter or circumference). Not mixture.

Value

A vector of decrease coefficients.

Examples

```
decrease(30, 120)
decrease(middle = 40, breast = 90)
```

decreaseMetric *Metric scrolling or decay*

Description

The average metric decay expresses the difference, in centimeters per meter, between the diameter (or circumference) at breast height and its diameter at mid-height of a stem related to the difference between the height at mid-height and that at breast height.

Usage

```
decreaseMetric(dmh, dbh, mh, bh = 1.3)
```

Arguments

dmh	numeric, the diameter at middle height in centimeter (cm).
dbh	numeric, the diameter at breast height in centimeter (cm).
mh	numeric, the middle (or cut) height in meter (m).
bh	Either a numeric value standing for the breast height in meter (m) of all trees or a numeric vector standing for the breast height of each tree. Default is 1.3.

Value

Metric decay

See Also

reducecoef

Examples

```
decreaseMetric(dmh = 40, dbh = 90, mh = 7)
decreaseMetric(45, 85, 9)
```

---

deg *Degree and Radian*

Description

deg converts angle values from radians to degrees.
rad converts angle values from degrees to radians.

Usage

```
deg(radian)

rad(degree)
```

Arguments

radian	numeric, vector of radian values to be converted to degrees.
degree	numeric, vector of degree values to be converted to radians.

Value

deg returns vector of degree values while rad returns vector of radian values.

See Also

principal.

Examples

```
deg(pi / 2)
rad(180)
```

`densityRegen` *Density of regeneration*

Description

Computes the density per plot of tree regeneration based on counts in subplots.

Usage

```
densityRegen(data = NULL, plot = NULL, count, nbSubPlot, area)
```

Arguments

`data`	an optional data frame, list, tibble or object coercible by `as.data.frame` to a data frame containing the variables whose names are given in `count` and `plot`.
`plot`	an optional character, name of the variable containing the plot identities. If `data` is missing, a vector providing the plot identities.
`count`	character, name of the variable containing the counts: number of stems (individuals). If `data` is missing, a numeric vector providing the the counts: number of stems (individuals).
`nbSubPlot`	numeric, number of subplots per plot.
`area`	numeric, area of each subplot.

`densityTree` *Tree density*

Description

Density of trees per plot.

Usage

```
densityTree(number, area, overall = TRUE)
```

Arguments

`number`	numeric, vector of tree count in each plot.
`area`	numeric, area of a plot.
`overall`	logical, if `TRUE`, an overall mean density is computed, otherwise density is computed for each plot. Default is `TRUE`.

Details

If every plot have same area, `area` is a numeric value, otherwise `area` is a vector of each plot area.

Value

Vector of density.

See Also

`densityRegen` for regeneration density.

Examples

```
count <- setNames(
  c(87, 104, 83, 132, 107, 84, 110, 115, 112, 94),
  LETTERS[1:10]
)
densityTree(count, 10)
densityTree(count, area = 10, overall = FALSE)
densityTree(count, area = 10:19, overall = FALSE)
```

---

diameterMean *Mean diameter*

Description

Mean diameter of a forestry stand.

Usage

```
diameterMean(dbh)
```

Arguments

dbh	numeric, vector of diameter.

Value

Mean diameter.

See Also

`dbh`, `basal_i`

Examples

```
set.seed(1)
diameter <- rnorm(10, 100, 20)
diameterMean(dbh = diameter)
```

distanceH *Horizontal distance*

Description

Horizontal distance calculation for sloping area.

Usage

```
distanceH(
  distance,
  angle,
  type = c("slope", "angle"),
  angleUnit = c("deg", "rad")
)
```

Arguments

distance	numeric, vector of the distance measured on sloping area.
angle	numeric, vector of angle or slope values.
type	character, type of angle argument. Either "angle" or "slope". Default is "slope".
angleUnit	character, unit of angle measures if type = "angle". Either "deg" for degree or "rad" for radian. Default is "deg".

Value

A vector of horizontal distance.

Examples

```
distanceH(20, 30)
distanceH(20, angle = 30, type = "slope")
distanceH(20, angle = 25, type = "angle")
```

factorize *Making factor vectors*

Description

Changes character vectors of a data set to factor vectors.

Usage

```
factorize(data, binary = FALSE)
```

Arguments

data	data frame or tibble data set.
binary	logical indicating if binary numeric data should be considered as factor. Default is FALSE.

Details

When binary = TRUE, variables stored as numeric and which have exactly two levels are changed to factor.

Value

Data frame with all character vectors changed to factor vectors.

---

fibonacci *Fibonacci series*

Description

Generates numbers from Fibonacci series.

Usage

```
fibonacci(n, PrintFib = FALSE, Uo = 0, U1 = 1)
```

Arguments

n	integer, the size of the series.
PrintFib	logical, indicating if the series should be printed.
Uo, U1	integer, the first two numbers of the series.

Details

The series equation is Un = U_(n-2) /U_(n-1).

Value

Either an integer, result of the function or a vector of n first numbers of the series.

Author(s)

Narcisse Yehouenou <narcisstar211@gmail.com>

See Also

fiboRate

Examples

```
fibonacci(n = 10, PrintFib = TRUE)
fibonacci(n = 10, Uo = 1, U1 = 3, PrintFib = FALSE)
```

`fiboRate` *Fibonacci series ratio*

Description

Computes rates from Fibonacci series.

Usage

```
fiboRate(n, PrintSer = FALSE, Uo = 0, U1 = 1)
```

Arguments

`n`	integer, the size of the series.
`PrintSer`	logical, indicating if the series should be printed.
`Uo, U1`	integer, the first number of the series.

Details

The series equation is Un = U_(n-2) /U_(n-1). The function returns golden number when Uo = 0, and U1 = 1. Larger n is, more precise the number (result) is.

Value

Either a numeric, result of the rate of `nth` and `(n-1)th` numbers in Fibonacci series or all `(n-1)th` those rates.

Author(s)

Narcisse Yehouenou <narcisstar211@gmail.com>

See Also

`fibonacci`

Examples

```
## Golden number (Le Nombre d'Or)
fiboRate(n = 18, PrintSer = FALSE, Uo = 0, U1 = 1)
## (1+sqrt(5))/2
fiboRate(n = 10, PrintSer = TRUE, Uo = 0, U1 = 1)
```

`girard` *Girard Form Class Girard Form Class is a form quotient used to estimate taper.*

Description

Girard Form Class Girard Form Class is a form quotient used to estimate taper.

Usage

```
girard(dbh, dbhIn)
```

Arguments

`dbh`	numeric, diameter outside bark at breast height.
`dbhIn`	numeric, diameter inside bark at the top of the first log

References

Strimbu, B. (2021). Dendrometry Field Manual.

`green` *Index of Green*

Description

Index of Green

Usage

```
green(density)
```

Arguments

`density`	numeric, vector of the density.

Value

Index of Green.

height *Height of Tree or any vertical Object*

Description

Computes the height of tree, pillar, girder, mast or any vertical object. It allows either slope (in percent) or angle (in degrees or radians). No matter the relative position of the persons who measures the angle or the slope.

Usage

```
height(
  distance,
  top,
  bottom,
  type = c("angle", "slope"),
  angleUnit = c("deg", "rad")
)
```

Arguments

distance	numeric, vector of the horizontal distance between object and the person who measures angle.
top, bottom	numeric vector of top angle and bottom angle respectively (readings from a clinometer).
type	the type of top and bottom measures. Either "angle" or "slope". Default is "slope".
angleUnit	the unit of top and bottom measures when type = "angle". Either "deg" for degree or "rad" for radian. Default is "deg".

Value

A vector of heights.

Examples

```
height(10, 80, 17)
height(17, top = -18, bottom = -113)
height(distance = 18, top = 42, bottom = -12, type = "angle", angleUnit = "deg")
height(
  distance = 18:21, top = 42:45, bottom = -12:-15, type = "angle",
  angleUnit = "deg"
)
## Below shows warning messages
height(
  distance = 18:21, top = -42:-45, bottom = -12:-15, type = "angle",
  angleUnit = "deg"
)
```

Logging *Tree metrics for logging*

Description

Data frame of 24 rows and 8 columns containing tree measures.

Usage

```
data(Logging)
```

Format

Data frame with twenty five observations and eight variables:

tree Tree name (scientific gender).

hauteur Stem length in meter (m).

diametreMedian Tree median diameter in centimeter (cm).

perimetreMedian Tree median circumference in centimeter (cm).

diametreSection Tree diameter at the end in centimeter (cm).

perimetreSection Tree circumference at the end in centimeter (cm).

diametreBase Tree diameter at the base in centimeter (cm).

perimetreBase Tree circumference at the base in centimeter (cm).

Author(s)

Narcisse Yehouenou <narcisstar211@gmail.com>

Source

Fake data simulated for tutorial purposes.

Examples

```
# demo(volume)
```

loreyHeight *Lorey's mean height*

Description

The average height of the trees in a plot, weighted by their basal area.

Usage

```
loreyHeight(basal, height)
```

Arguments

basal	numeric, vector of trees' individual basal area.
height	numeric, vector of trees' individual height.

Value

Average Lorey height of a stand.

See Also

height, basal_i

Examples

```
set.seed(1)
donnee <- data.frame(
  hauteur = rnorm(10, 12, 3),
  area = basal_i(rnorm(10, 100, 20))
)
loreyHeight(basal = donnee$area, height = donnee$hauteur)
```

makedata *Make stand data*

Description

Make data of stands according to defined factor1,factor2,factor3.

Usage

```
makedata(data, factor1 = "", factor2 = "", factor3 = "")
```

Arguments

data	data frame containing optional factors factor1, factor2, factor3.
factor1, factor2, factor3	optional variables of the data frame that define subsets to consider.

Value

A list of data.

Examples

```
# require(BiodiversityR)
# data(ifri, package = "BiodiversityR")
# a1=makedata(ifri, factor1 = "forest", factor2 = "plotID", factor3 = "species")
# a2=makedata(ifri, factor1 = "species")
# makedata(ifri, factor2 = "")
# identical(makedata(ifri), ifri)
```

---

param_i *Structural parameters for stands*

Description

param computes structural parameters per stands specified in factor arguments.
param_i computes structural parameters for a stand.

Usage

```
param_i(data, plot = "", DBH = "", height = "", crown = "", area = NULL,
k = 100, kCrown = 1)

param(data, plot = "", DBH = "", height = "", crown = "", area = NULL,
k = 100, kCrown = 1, factor1 = "", factor2 = "", factor3 = "")
```

Arguments

data	a data frame, list, tibble or object coercible by as.data.frame to a data frame containing the variables whose names would be given in DBH, height and crown.
plot, DBH, height, crown	optional characters, names of the variables of data containing respectively the plot's identification or code, the diameter at breast height, the tree total height, and the crown diameter of each individual tree.
area	numeric, area of a plot (see basal section details for unit).
k, kCrown	numeric, used to convert diameter and crown diameter units respectively. Default are k = 100 and kCrown = 1. (see basal)
factor1, factor2, factor3	character, optional variables of the data frame that define subsets to consider.

Details

Blackman and Green indices are returned if combinations of specified `factor1..3` contain more than one plot. Otherwise, the right are returned as attributes.

Value

A vector, matrix or list of matrices containing of structural parameters.

Examples

```
param_i(
  data = Logging, plot = "tree", DBH = "diametreMedian",
  height = "hauteur", crown = "perimetreBase", area = 0.03, kCrown = 100
)
```

---

print.angle *Print Angle*

Description

Method to print angle and returns it invisibly.

Usage

```
## S3 method for class 'angle'
print(x, ...)
```

Arguments

x	an angle object.
...	further arguments passed to or from other methods.

---

print.slope *Print Slope*

Description

Method to print slope and returns it invisibly.

Usage

```
## S3 method for class 'slope'
print(x, ...)
```

Arguments

x	a slope object.
...	further arguments passed to or from other methods.

`reducecoef` *The reduction coefficient*

Description

The reduction coefficient is the ratio between the difference in size at breast height and mid-height on the one hand, and the size at breast height on the other. It is thus the complement to 1 of the coefficient of decrease.

Usage

```
reducecoef(middle, breast)
```

Arguments

`middle`	numeric, the diameter or circumference at middle height.
`breast`	numeric, the diameter or circumference at breast height.

Details

Both `middle` and `breast` arguments should be of the same type (either diameter or circumference). Not mixture.

Value

The reduction coefficient.

See Also

`decrease`

Examples

```
reducecoef(30, 120)
reducecoef(middle = 40, breast = 90)
```

rfreq *Relative Frequency*

Description

Relative Frequency in percentage.

Usage

```
rfreq(x)
```

Arguments

x	numeric vector.

sampleSize *Sample size*

Description

Sample size

Usage

```
sampleSize(
  confLev = 0.95,
  popPro = 0.5,
  errorMargin = 0.05,
  size = NULL,
  method = "",
  cv = NULL
)
```

Arguments

confLev	numeric, the confidence level. Default is 0.05.
popPro	numeric, proportion of population which have considered factor. Default is 0.5.
errorMargin	numeric, margin error. Default is 0.05.
size	integer, population size when it is known. If not specified, simple random sampling will be used. Allows infinite.
method	optional character string specifying method to use if not simple adjusted is desired. Only "cauchran" is implemented now.
cv	variation coefficient.

Value

The sample size.

Note

Population size to be considered as large or infinite heavily depends on error margin. Lower error margin increases population size to be conidered as large or infinite. For errorMargin = .05, size = 152 231 and cauchran 151 760 when confLev = .05

Examples

```
sampleSize(confLev = .95, popPro = 0.4, errorMargin = .05)
sampleSize(confLev = .95, popPro = 0.5, errorMargin = .05, size = 150)
sampleSize(
  confLev = .95, popPro = 0.5, errorMargin = .05, size = 150,
  method = "cauchran"
)
sampleSize()
```

---

shape *The shape coefficient*

Description

The shape coefficient of the tree is the ratio of the actual volume of the tree to the volume of a cylinder having as base the surface of the section at 1.3 m (or a given breast height) and as length, the height (at bole level) of the tree.

Usage

```
shape(volume, height, dbh, basal = NULL)
```

Arguments

volume	numeric, tree real volume.
height	numeric, tree height.
dbh	numeric, diameter at breast height (DBH).
basal	numeric, basal area. Is used when dbh is not specified.

Value

The shape coefficient.

See Also

volume, for tree real volume.

Examples

```
shape(volume = 10000, 11, dbh = 40)
shape(volume = 10000, 11, 40)
shape(volume = 10000, 11, basal = 2256.637)
## Bellow gives warning
shape(volume = 10000, height = 11, dbh = 40, basal = 2256.637)
```

---

skewness *Skewness coefficient*

Description

Skewness coefficient

Usage

```
skewness(x)
```

Arguments

x	numeric vector.

Value

The skewness coefficient.

Examples

```
data("Logging")
skewness(Logging$hauteur)
hist(Logging$hauteur, 3)
```

---

spNmReduce *Abbreviates a Botanical or Zoological Latin Name into an Eight-character from 'Gender epithet' to 'G. epithet'*

Description

To abbreviate species name from 'Gender epithet' to 'G. epithet'. Useful in plots with species names.

Usage

```
spNmReduce(name, sep = " ")
```

Arguments

name	a factor coercible vector of species name in forms 'Gender epithet'.
sep	character string which separates Gender and epithet. Default is space " ".

Details

Returned reduced names are made unique.

Value

A factor vector of species reduced names in forms 'G. epithet'.

See Also

make.cepnames in vegan package.

---

stacking *Stack all vectors of a data frame or list*

Description

Stacking all columns of a data frame or vectors of a list into a single vector.

Usage

```
stacking(data)
```

Arguments

data	data frame, tibble or list.

Value

A vector of all element of the argument data.

Tree *Dendrometric measures on tree*

Description

Data frame of 10 rows and 5 columns containing tree measures.

Usage

```
data(Tree)
```

Format

Data frame with ten observations and five variables:

circum Tree circumference in centimeter (cm).

dist Horizontal distance between the person measuring angles and the tree (m).

up Angle measured for the top part of the tree in degree (\u00b0). It is used to calculate the total tree height.

down Angle measured for the bottom part of the tree in degree (\u00b0).

fut Bole angle measure in degree (\u00b0); Bole is where the first branch occurs on the trunk. It is used to calculate the merchantable tree height.

Author(s)

Narcisse Yehouenou <narcisstar211@gmail.com>

Source

Fake data simulated for tutorial purposes.

Examples

```
# demo(dendro)
```

volume *Tree stem and log Volume*

Description

Determining the volume of the log or of the tree.

Usage

```
volume(height, dm, do, ds, circum, circumo, circums,
       method = "huber", successive = FALSE, log)
```

Arguments

height	numeric, stem (whole bole) length. When successive is "TRUE", it stands for log length.
do, dm, ds	numeric, respectively base, median and end diameter.
circumo, circum, circums	numeric, respectively base, median and end circumference.
method	character string, the method of volume computation. Can be one of "huber", "smalian", "cone", or "newton". Default is "huber".
successive	logical. If TRUE, Successive method is applied. is applied. Default is FALSE.
log	a vector indicating tree to which belongs each log. Is used only when successive is "TRUE".

Details

Using method = cone refers to truncated cone method.

Value

A numeric vector of logs or trees volume.

See Also

shape, for shape coefficient.

Examples

```
## huber method
volume(height = 10, dm = 35)
volume(height = 10, circum = 100)

## smalian method
volume(height = 10, do = 45, ds = 15, method = "smalian")
volume(height = 10, circumo = 200, circums = 110, method = "smalian")
```

```
## cone method
volume(height = 10, do = 45, ds = 15, method = "cone")
volume(height = 10, circumo = 200, circums = 110, method = "cone")

## newton method
volume(height = 10, dm = 35, do = 45, ds = 15, method = "newton")
volume(
  height = 10, circum = 100, circumo = 200, circums = 110,
  method = "newton"
)
```

Index