

# Package: PhytoIn (via r-universe)

June 1, 2026

**Type** Package

**Title** Vegetation Analysis and Forest Inventory

**Version** 0.2.0

**Description** Provides functions and example datasets for phytosociological analysis, forest inventory, biomass and carbon estimation, and visualization of vegetation data. Includes functions to compute structural parameters [phytoparam(), summary.param(), stats()], estimate above-ground biomass and carbon [AGB()], stratify wood volume by diameter at breast height (DBH) classes [stratvol()], generate collector and rarefaction curves [collector.curve(), rarefaction()], and visualize basal areas on quadrat maps [BAplot()], including rectangular plots and individual coordinates. Several example datasets are provided to demonstrate the functionality of these tools. For more details see FAO (1981, ISBN:92-5-101132-X) ``Manual of forest inventory'', IBGE (2012, ISBN:9788524042720) ``Manual técnico da vegetação brasileira'' and Heringer et al. (2020) ``Phytosociology in R: A routine to estimate phytosociological parameters'' <doi:10.22533/at.ed.3552009033>.

**Depends** R (>= 3.5.0)

**Imports** ggplot2, ggforce, packcircles, BIOMASS, scales, utils, stats

**Suggests** httr2, spelling

**License** GPL-3

**URL** <https://github.com/PhytoIn/PhytoIn>

**BugReports** <https://github.com/PhytoIn/PhytoIn/issues>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.2

**Language** en-US

**Config/pak/sysreqs** libabsl-dev cmake libfontconfig1-dev libfreetype6-dev libgdal-dev gdal-bin libgeos-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev

**Repository** <https://phytoin.r-universe.dev>

**Date/Publication** 2025-09-24 17:20:16 UTC

**RemoteUrl** <https://github.com/phytoin/phytoin>

**RemoteRef** HEAD

**RemoteSha** b6cef0afb2d8924ceca8dde984f8046dcb3ec2f2

## Contents

|                            |    |
|----------------------------|----|
| AGB . . . . .              | 2  |
| BAplot . . . . .           | 5  |
| collector.curve . . . . .  | 7  |
| phytoparam . . . . .       | 10 |
| plot.param . . . . .       | 13 |
| point.df . . . . .         | 14 |
| quadrat.df . . . . .       | 15 |
| quadrat2_plot.df . . . . . | 16 |
| quadrat2_tree.df . . . . . | 16 |
| quadrat3_rect.df . . . . . | 17 |
| rarefaction . . . . .      | 18 |
| stats . . . . .            | 21 |
| stratvol . . . . .         | 23 |
| summary.param . . . . .    | 24 |

|              |           |
|--------------|-----------|
| <b>Index</b> | <b>26</b> |
|--------------|-----------|

---

AGB

*Estimate the above-ground biomass*

---

## Description

Estimate the above-ground biomass (AGB), carbon (C) and CO<sub>2</sub> equivalent (CO<sub>2</sub>e) of trees.

## Usage

```
AGB(
  x,
  measure.label,
  h,
  taxon = "taxon",
  dead = "dead",
  circumference = TRUE,
  su = "quadrat",
  area,
  coord,
  rm.dead = TRUE,
  check.spelling = FALSE,
```

```

correct.taxon = TRUE,
sort = TRUE,
decreasing = TRUE,
cache = FALSE,
long = FALSE
)

```

### Arguments

|                             |  |
|-----------------------------|--|
| <code>x</code>              | A data.frame with the community sample data. See <b>Details</b> .  |
| <code>measure.label</code>  | Name of the column with circumference/diameter at breast height (cm).  |
| <code>h</code>              | Name of the column with tree height (m). If omitted in <code>x</code> , height is estimated from <code>coord</code> .  |
| <code>taxon</code>          | Name of the column with sampled taxon names. Default "taxon". Use UTF-8; accents and special characters are not allowed.   |
| <code>dead</code>           | String used to identify dead individuals. Default "dead".  |
| <code>circumference</code>  | Logical; if TRUE (default), CBH is assumed; otherwise DBH is assumed.  |
| <code>su</code>             | Name of the column with sample-unit identifiers. Default "quadrat".  |
| <code>area</code>           | Numeric scalar: total sampled area (ha).   |
| <code>coord</code>          | A vector <code>c(longitude, latitude)</code> or a two-column matrix/data.frame of site coordinates (decimal degrees). Required when <code>h</code> is missing in <code>x</code> .                                    |
| <code>rm.dead</code>        | Logical; if TRUE (default) dead individuals are removed prior to biomass calculation.  |
| <code>check.spelling</code> | Logical; if TRUE, near-matching taxon names are flagged for correction. Default FALSE.   |
| <code>correct.taxon</code>  | Logical; if TRUE (default) taxon names are standardized via TNRS.  |
| <code>sort</code>           | Logical; if TRUE (default) taxa are sorted by AGB.   |
| <code>decreasing</code>     | Logical; if TRUE (default) sorting is in decreasing order.   |
| <code>cache</code>          | Logical or NULL; if TRUE the function with <code>write</code> and <code>use</code> a cache to reduce online search of taxa names. (NULL means use cache but clear it first). Default is <code>cache = FALSE</code> . |
| <code>long</code>           | Logical; if FALSE (default) the <code>\$tree</code> component is omitted (see <b>Value</b> ).  |

### Details

AGB is a wrapper around **BIOMASS** functions `getWoodDensity`, `computeAGB`, and `correctTaxo` (Rejou-Mechain et al., 2017). Tree biomasses are computed using the allometric model of Chave et al. (2014).

It is expected that taxon names are binomials (genus and species). The function splits the taxon string into two columns (genus, species) to retrieve wood density. Single-word taxa (e.g., Indet, Dead) receive `species = NA`.

Wood density ( $\text{g/cm}^3$ ) is obtained from a global database (~16,500 species). If a species is missing, the genus mean is used; if the genus is missing, the sample-unit (su) mean is used.

The input `x` must include columns for sample-unit labels, taxon names, CBH/DBH, and optionally height. If height is absent, `coord` is mandatory to allow height estimation.

The CBH/DBH column allows multi-stem notation such as "17.1+8+5.7+6.8". The plus sign separates stems; decimal separators can be points or commas; spaces around "+" are ignored. Column names in `x` are coerced to lower case at runtime, making the function case-insensitive.

Measurement units: CBH/DBH in centimeters; height in meters.

## Value

An object of class "biomass" with up to four components:

- `$tree`: individual records (taxon, wood density `wd`, assignment level, AGB, C, CO<sub>2</sub>e);
- `$taxon`: AGB, C, and CO<sub>2</sub>e per taxon (Mg per ha);
- `$total`: total AGB, C, and CO<sub>2</sub>e per ha;
- `$WD.level`: percentage of wood-density assignments at species, genus, and sample-unit levels.

## References

- Boyle, B. et al. (2013) *BMC Bioinformatics* 14:16.  
 Chave, J. et al. (2014) *Global Change Biology* 20(10):3177–3190.  
 Rejou-Mechain, M. et al. (2017) *Methods in Ecology and Evolution* 8:1163–1167.  
 Zanne, A.E. et al. (2009) Global wood density database. Dryad.

## See Also

**BIOMASS** ([getWoodDensity](#), [computeAGB](#), [correctTaxo](#))

## Examples

```
data <- quadrat.df
head(data)

result1 <- AGB(
  data, measure.label = "CBH", h = "h", taxon = "Species", dead = "Morta",
  circumference = TRUE, su = "Plot", area = 0.0625, rm.dead = TRUE,
  check.spelling = FALSE, correct.taxon = TRUE, sort = TRUE,
  decreasing = TRUE, long = TRUE
)
head(result1$tree)
result1$taxon
result1$total
result1$WD.level

quadrat.default <- quadrat.df
colnames(quadrat.default) <- c("quadrat", "family", "taxon", "cbh", "h")

Result2 <- AGB(x = quadrat.default, measure.label = "cbh",
  circumference = TRUE, h = "h", dead = "Morta", area = 0.0625)
```

```

head(Resul2$tree)
Resul2$taxon
Resul2$total
Resul2$WD.level
## Not run:
Resul3 <- AGB(data, measure.label = "CBH", taxon = "Species", dead = "Morta",
              circumference = TRUE, su = "Plot", area = 0.0625,
              coord = c(-47.85, -21.17))
Resul3$taxon
Resul3$total
Resul3$WD.level

## End(Not run)

```

---

BAplot

*Plot basal areas on a map of quadrats*


---

### Description

Plot basal areas of trees on a map of quadrats. If individual tree coordinates are not known, the coordinates inside the quadrats are randomly defined according to the uniform distribution.

### Usage

```

BAplot(
  formula,
  data,
  taxon = "taxon",
  circumference = TRUE,
  quadrat.size,
  dead = "dead",
  rm.dead = FALSE,
  origin = c(0, 0),
  col = "grey40",
  alpha = 1,
  cex.radius = 1,
  ind.coord = FALSE,
  legend = TRUE,
  long = FALSE
)

```

### Arguments

|         |   |
|---------|---|
| formula | A model formula indicating the trunk measure (circumference [default] or diameter) in centimeters and the xy coordinates of each quadrat the tree belongs (ind.coord = FALSE) or, the actual tree coordinates (ind.coord = TRUE). Example: $\text{measure} \sim x + y$ . See Details. |
|---------|---|

|                            |  |
|----------------------------|--|
| <code>data</code>          | A data frame containing the community sample data. See Details.  |
| <code>taxon</code>         | Name of the column representing the sampled taxa. Default is "taxon".  |
| <code>circumference</code> | Logical. If TRUE (the default), the function assumes that the circumference at breast height was measured.   |
| <code>quadrat.size</code>  | A vector indicating the side lengths (in meters) of the x and y quadrat sides (e.g., $c(x, y)$ ). It can be given as a single value if the quadrat is a square.  |
| <code>dead</code>          | String used to identify the dead individuals. Default is "dead".   |
| <code>rm.dead</code>       | Logical. If FALSE (the default) basal areas of dead individuals are plotted.   |
| <code>origin</code>        | A numeric vector indicating the map origin coordinates. Default is $c(0, 0)$ .   |
| <code>col</code>           | Circle color of represented basal areas. This argument has value only if <code>legend = FALSE</code> . Default is "grey40".  |
| <code>alpha</code>         | Value of transparency factor: zero (100% transparent) – 1 (no transparency). Default is 1.   |
| <code>cex.radius</code>    | A numerical value giving the amount by which the tree radius should be magnified relative to the actual measure. Default is 1.   |
| <code>ind.coord</code>     | Logical indicating whether the individual coordinates are given. If FALSE (the default) the tree coordinates inside the quadrats are randomly defined. If TRUE, the actual tree coordinates are plotted. |
| <code>legend</code>        | Logical. If TRUE (default), the taxon is used as the color legend; otherwise, circle color will be defined by the argument <code>col</code> .  |
| <code>long</code>          | Logical. If FALSE (default) the function does not return the result data frame, which contains xy coordinates, radius and taxon name of each sampled tree.   |

### Details

BAplot uses the function `circleRepelLayout()` from the **packcircles** package to rearrange circle coordinates to avoid overlapping. The minimum distance allowed among trees is 1 meter. The packages **ggforce** and **ggplot2** are used to draw the map.

The data frame passed to the `data` argument must include two columns indicating x and y coordinates of each quadrat that the tree belongs or the actual tree coordinates. If actual coordinates are supplied, the `ind.coord` argument must be set TRUE.

Circumference/diameter measures accept the traditional notation for multiple trunks, e.g., "17.1+8+5.7+6.8". The plus sign is the separator for each trunk measure. Decimal separator can be point or comma and spaces after or before "+" are ignored by the function.

### Value

A plot representing the quadrat map and tree basal areas. If `long = TRUE`, the function returns a data frame containing taxon name, xy coordinates and radius of each sampled tree.

### Author(s)

Rodrigo A. S. Pereira (<raspereira@usp.br>)

## References

- Collins, C. R., and Stephenson, K. (2003). A circle packing algorithm. *Computational Geometry*, 25(3), 233–256. doi:10.1016/S09257721(02)000998
- Wang, W., Wang, H., Dai, G., and Wang, H. (2006). Visualization of large hierarchical data by circle packing. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 517–520. doi:10.1145/1124772.1124851

## Examples

```
# Using plot coordinates (random coordinates for individuals)
data1 <- quadrat2_plot.df
BAplot(formula = CBH ~ x + y, data = data1, taxon = "Species",
        circumference = TRUE, quadrat.size = 5, dead = "Morta",
        rm.dead = FALSE, alpha = 0.4, cex.radius = 2,
        legend = TRUE, long = FALSE, ind.coord = FALSE)

# Using actual coordinates
data2 <- quadrat2_tree.df
BAplot(formula = CBH ~ x + y, data = data2, taxon = "Species",
        circumference = TRUE, quadrat.size = 5, dead = "Morta",
        rm.dead = FALSE, alpha = 0.4, cex.radius = 2,
        legend = TRUE, long = FALSE, ind.coord = TRUE)

# Rectangular plots and plot coordinates
data3 <- quadrat3_rect.df
BAplot(formula = DBH ~ x + y, data = data3, taxon = "Species",
        circumference = FALSE, quadrat.size = c(20, 10),
        dead = "Morta", rm.dead = FALSE, col = "blue",
        alpha = 0.4, cex.radius = 2, legend = FALSE,
        long = FALSE, ind.coord = FALSE)
```

---

collector.curve

*Species–area (collector’s curve) function*

---

## Description

Computes species accumulation (collector’s) curves based on sample units (SUs). The function performs random resampling of the input matrix or data frame to estimate the expected species richness per number of SUs, with confidence intervals derived from multiple permutations.

## Usage

```
collector.curve(
  formula,
  data,
  x,
  times = 1000,
```

```

replace = FALSE,
prob = 0.95,
spar = 0,
xlab,
ylab,
plot = TRUE,
long = FALSE,
theme = "theme_classic"
)

```

### Arguments

|         |   |
|---------|---|
| formula | An optional formula specifying the relationship between taxa and sample units (e.g., Taxon ~ Sample). If provided, the function extracts variables from data. A third variable may be included to remove dead individuals (e.g., Taxon ~ Sample - Dead).  |
| data    | A data frame containing the variables specified in formula ('long format'). It must contain one column representing the sample unit labels (e.g., quadrats or points) and one column representing the taxon names of the individual plants. This argument accepts the data frame used in the argument x in <a href="#">phytoparam</a> . |
| x       | Species-by-sample matrix, with rows representing SUs and columns representing taxa ('wide format'). Can be either an abundance or presence-absence matrix. Ignored if formula and data are used.  |
| times   | Integer. Number of random permutations used in calculations. Default is 1000. Larger values (> 1000) yield more stable estimates.   |
| replace | Logical. Indicates whether resampling is performed with replacement (TRUE, bootstrap) or without replacement (FALSE, default).  |
| prob    | Numeric. Probability level used for computing confidence intervals around species accumulation (default = 0.95).  |
| spar    | Numeric. Controls the smoothing parameter for plotted confidence intervals via spline interpolation. Default = 0 (no smoothing).  |
| xlab    | Character. Label for the x-axis in the plot. Default = "Number of samples".   |
| ylab    | Character. Label for the y-axis in the plot. Default = "Number of species".   |
| plot    | Logical. If TRUE (default), the species accumulation curve is plotted.  |
| long    | Logical. If TRUE, returns detailed results, including the full set of resampling matrices. Default = FALSE.   |
| theme   | Character string specifying the ggplot2 theme to apply (e.g., "theme_classic", "theme_bw", "theme_minimal"). Defaults to "theme_classic".   |

### Details

Species accumulation curves are computed by sequentially adding sample units and recording species richness across permutations. Confidence intervals are estimated from the empirical distribution of resampled richness values. The plotted confidence intervals are smoothed using spline interpolation if  $spar > 0$ .

It is recommended to assign the output to an object, as the complete output (particularly with `long = TRUE`) can be large.

**Value**

If long = FALSE (default), returns a data frame with columns:

- n: number of sample units.
- s: mean number of species.
- lower: lower confidence interval bound.
- upper: upper confidence interval bound.

If long = TRUE, returns a list with:

- matrix.s: full matrix of species richness per permutation.
- s: the summarized data frame described above.

**Note**

With long = TRUE, the function provides access to the complete set of resampling results, useful for additional data analyses.

**Author(s)**

Rodrigo Augusto Santinelo Pereira <raspereira@usp.br>  
Adriano Sanches Melo

**References**

- Magurran, A. E. (1988). *Ecological Diversity and Its Measurement*. Croom Helm.  
Magurran, A. E. (2004). *Measuring Biological Diversity*. Blackwell Publishing.

**Examples**

```
## Using 'formula' (long format)
## Without smoothing confidence intervals
collector.curve(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  times = 1000, long = FALSE, plot = TRUE
)

## Smoothing confidence intervals
collector.curve(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  spar = 0.6, times = 1000, long = FALSE, plot = TRUE
)

## Using different plot themes
collector.curve(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  times = 1000, long = FALSE, plot = TRUE, theme = "theme_light"
```

```

)
collector.curve(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  times = 1000, long = FALSE, plot = TRUE, theme = "theme_bw"
)
collector.curve(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  times = 1000, long = FALSE, plot = TRUE, theme = "theme_minimal"
)

## Using a matrix (wide format)
data.matrix <- with(
  quadrat.df,
  table(Plot, Species, exclude = "Morta")
)
collector.curve(x = data.matrix, times = 1000)

## Alternatively...
data.matrix <- as.matrix(
  xtabs(~ Plot + Species, data = quadrat.df, exclude = "Morta")
)
collector.curve(x = data.matrix, times = 1000)

```

---

phytoparam

*Estimate phytosociological parameters and diversity indices*


---

## Description

Estimate the phytosociological parameters and the Shannon–Wiener, Pielou, and Simpson diversity indices, using the quadrat or the point-centered quarter methods.

## Usage

```

phytoparam(
  x,
  measure.label = NULL,
  h = "h",
  taxon = "taxon",
  family = "family",
  dead = "dead",
  circumference = TRUE,
  su = "quadrat",
  height = TRUE,
  quadrat = TRUE,
  su.size,

```

```

    d = "distance",
    shape.factor = 1,
    rm.dead = FALSE,
    check.spelling = TRUE
  )

```

### Arguments

|                |  |
|----------------|--|
| x              | A data.frame containing the community sample data. See 'Details'.  |
| measure.label  | Name of the column representing the circumference/diameter at breast height. If omitted the function assumes the default names "cbh" or "dbh" for circumference or diameter at breast height, respectively (see circumference).  |
| h              | Name of the column representing trunk height. Default is "h".  |
| taxon          | Name of the column representing the sampled taxa. Default is "taxon". Use UTF-8 encoding; accents and special characters are not allowed.  |
| family         | Name of the column representing the family names of the sampled taxa. Default is "family". Used to calculate the number of individuals and number of species per family. If you do not want these parameters, set family = NA. Use UTF-8 encoding; accents and special characters are not allowed. |
| dead           | String used to identify dead individuals. Default is "dead".   |
| circumference  | Logical. If TRUE (default) circumference at breast height was measured; otherwise "dbh" is assumed.  |
| su             | Name of the column representing the sample-unit identifier. Default is "quadrat" for the quadrat method and "point" for the point-centered quarter method.   |
| height         | Logical. If FALSE (default) trunk volume is not calculated.  |
| quadrat        | Logical. If TRUE (default) data were sampled using the quadrat method; if FALSE, the point-centered quarter method is assumed.   |
| su.size        | Numeric scalar giving the quadrat area ( $m^2$ ); required only if quadrat = TRUE.   |
| d              | Name of the column representing the point-to-tree distance; required only if quadrat = FALSE. Default is "distance".   |
| shape.factor   | Numeric value between in 0 and 1, indicating the trunk shape. Value 1 assumes a perfect cylinder.  |
| rm.dead        | Logical. If FALSE (default) phytosociological parameters for dead individuals are calculated.  |
| check.spelling | Logical. If TRUE (default) taxon names are checked for misspelling.  |

### Details

The function estimates phytosociological parameters for tree communities sampled by quadrat or point-centered quarter methods (quadrat = TRUE or FALSE, respectively).

For the quadrat method, x must contain columns for sample-unit labels, taxon names, and "cbh" or "dbh" measurements for each sampled tree. Additionally, trunk height and family can be included to estimate volume and family-level parameters.

For the point-centered quarter method, `x` must contain (in addition to the mandatory quadrat columns) a column for the distance from the point to each individual.

The "cbh"/"dbh" column accepts multiple-stem notation, e.g. "17.1+8+5.7+6.8". The plus sign delimits stems. Decimal delimiter may be period or comma; spaces around "+" are ignored. Column names in `x` are coerced to lowercase at runtime, making matching case-insensitive. If `x` contains the default column names, the arguments `h`, `taxon`, `family`, `dead`, `su` and `d` can be omitted.

Unbiased absolute density for the point-centered quarter method follows Pollard (1971) and Seber (1982).

**Measurement units:** individual "cbh"/"dbh" in centimeters; trunk height and point-to-individual distance in meters.

## Value

An object of class `param` with two or four data frames:

- `data`: A data frame containing the original community sample data, added with a column of the individual basal area (`ABi`).
- `global`: total parameters and diversity indices. Sampled area in hectares (`ha`), total density in individuals/ha, total dominance in  $m^2 ha^{-1}$  (basal area) or  $m^3 ha^{-1}$  (volume, when computed), and Shannon–Wiener  $H'$  in nats/individual (natural log).
- `param`: taxon-level table with observed number of individuals (`N`), absolute/relative density (`ADe`, `RDe`), absolute/relative frequency (`AFr`, `RFr`), absolute/relative dominance (`ADo`, `RDo`), absolute/relative volume (`AVol`, `RVol`), Importance Value Index (`IV`), and Cover Value Index (`CV`). Absolute parameters per hectare; relative parameters in percentage.
- `family`: If `family != NA`, a table listing the number of individuals and the number of species per family is presented.
- `vars`: A list containing objects used in the functions `AGB`, `stats` and `stratvol`.

## References

- Pollard, J. H. (1971). On distance estimators of density in randomly distributed forests. *Biometrics*, 27, 991–1002.
- Seber, G. A. F. (1982). *The Estimation of Animal Abundance and Related Parameters*. New York: Macmillan, pp. 41–45.

## See Also

[summary.param](#), [plot.param](#)

## Examples

```
## Quadrat method
quadrat.param <- phytoparam(
  x = quadrat.df, measure.label = "CBH", taxon = "Species",
  dead = "Morta", family = "Family", circumference = TRUE, su = "Plot",
  height = TRUE, su.size = 25, rm.dead = FALSE
)
summary(quadrat.param)
```

```

head(quadrat.param$data)
quadrat.param$global
quadrat.param$family
quadrat.param$param

## Point-centered quarter method
point.param <- phytoparam(
  x = point.df, measure.label = "CBH", taxon = "Species",
  dead = "Morta", family = "Family", circumference = TRUE, su = "Point",
  height = TRUE, quadrat = FALSE, d = "Distance", rm.dead = FALSE
)
summary(point.param)
head(point.param$data)
point.param$global
point.param$family
point.param$param

## Using default column names
point.default <- point.df
colnames(point.default) <- c("point", "family", "taxon", "distance", "cbh", "h")
point.param.default <- phytoparam(
  x = point.default, dead = "morta",
  circumference = TRUE, height = TRUE, quadrat = FALSE
)
summary(point.param.default)
point.param.default$global

## Plotting
plot(quadrat.param)
plot(point.param)
plot(point.param, theme = "theme_light")
plot(point.param, theme = "theme_bw")
plot(point.param, theme = "theme_minimal")

```

---

plot.param

*Plot relative phytosociological parameters by taxon*


---

### Description

Produce a stacked bar chart of relative dominance (RDo), relative frequency (RFr), and relative density (RDe) for each taxon contained in a param object returned by [phytoparam](#). Taxa are ordered by the Importance Value (IV).

### Usage

```

## S3 method for class 'param'
plot(x, theme = "theme_classic", ...)

```

**Arguments**

|       |   |
|-------|---|
| x     | An object of class <code>param</code> (output of <a href="#">phytoparam</a> ) whose <code>\$param</code> data frame contains at least the columns <code>Taxon</code> , <code>RDe</code> , <code>RFr</code> , <code>RDo</code> , and <code>IV</code> .   |
| theme | A <code>ggplot2</code> theme to apply. Either a character string naming a theme constructor in <b>ggplot2</b> (e.g., <code>"theme_light"</code> , <code>"theme_bw"</code> , <code>"theme_minimal"</code> ), or a <code>ggplot2</code> theme object. Invalid inputs fall back to <code>ggplot2::theme_classic()</code> . |
| ...   | Ignored.  |

**Details**

The function reshapes the taxon-level table to long format and draws a horizontal stacked bar chart (`RDo`, `RFr`, `RDe`) with taxa ordered by increasing `IV`.

**Value**

A `ggplot` object.

**See Also**

[phytoparam](#), [summary.param](#), and **ggplot2**.

**Examples**

```
res <- phytoparam(x = quadrat.df, measure.label = "CBH", taxon = "Species",
                 dead = "Morta", family = "Family", circumference = TRUE,
                 su = "Plot", height = TRUE, su.size = 25)
plot(res) # default theme (theme_classic)
plot(res, theme = "theme_light") # theme by name
plot(res, theme = ggplot2::theme_minimal()) # theme object
```

---

point.df

*Point-centered quarter dataset*

---

**Description**

A dataset containing tree measurements collected using the point-centered quarter method at 25 sampling points in a forest fragment located at the campus of the University of Sao Paulo, Ribeirao Preto, Brazil. At each point, the nearest tree in each of the four quadrants was sampled, totaling 100 individuals.

**Usage**

```
data(point.df)
```

**Format**

A data frame with 100 rows and 6 variables:

**Point** Sampling point identifier (1–25).

**Family** Botanical family of the tree.

**Species** Scientific name of the species.

**Distance** Distance from the point to the tree (m).

**CBH** Circumference at breast height (cm).

**h** Total height of the tree (m).

---

quadrat.df

*Forest inventory dataset from 25 quadrats*

---

**Description**

A dataset containing tree measurements from 25 quadrats (5 × 5 m each) sampled in a forest fragment located at the campus of the University of Sao Paulo, Ribeirao Preto, Brazil.

**Usage**

```
data(quadrat.df)
```

**Format**

A data frame with 171 rows and 5 variables:

**Plot** Quadrat identifier (e.g., X2Y3).

**Family** Botanical family of the tree.

**Species** Scientific name of the species.

**CBH** Circumference at breast height (cm).

**h** Total height of the tree (m).

---

`quadrat2_plot.df`*Quadrat dataset with coordinates for basal area plotting*

---

**Description**

A dataset containing tree measurements from 25 quadrats ( $5 \times 5$  m each) sampled in a forest fragment located at the campus of the University of Sao Paulo, Ribeirao Preto, Brazil. This dataset is organized to demonstrate the use of the function `BAPlot()` for visualizing basal areas. In addition to tree measurements, it provides the spatial coordinates of the lower-left corner of each quadrat.

**Usage**

```
data(quadrat2_plot.df)
```

**Format**

A data frame with 219 rows and 6 variables:

**Plot** Quadrat identifier (e.g., X1Y1).

**CBH** Circumference at breast height (cm).

**h** Total height of the tree (m).

**Species** Scientific name of the species.

**x** X coordinate (m) of the lower-left corner of the quadrat.

**y** Y coordinate (m) of the lower-left corner of the quadrat.

---

`quadrat2_tree.df`*Quadrat dataset with simulated individual tree coordinates*

---

**Description**

A dataset containing tree measurements from 25 quadrats ( $5 \times 5$  m each) sampled in a forest fragment located at the campus of the University of Sao Paulo, Ribeirao Preto, Brazil. This dataset corresponds to the same trees as in `quadrat2_plot.df`, but includes simulated coordinates (in meters) for each individual tree inside the quadrats. It is organized to demonstrate the use of the function `BAPlot()` with the argument `ind.coord = TRUE`.

**Usage**

```
data(quadrat2_tree.df)
```

**Format**

A data frame with 219 rows and 7 variables:

**Plot** Quadrat identifier (e.g., X1Y1).

**CBH** Circumference at breast height (cm).

**h** Total height of the tree (m).

**Species** Scientific name of the species.

**x** Simulated X coordinate (m) of the tree inside the quadrat.

**y** Simulated Y coordinate (m) of the tree inside the quadrat.

---

|                  |                                    |
|------------------|------------------------------------|
| quadrat3_rect.df | <i>Rectangular quadrat dataset</i> |
|------------------|------------------------------------|

---

**Description**

A dataset containing tree measurements from 25 rectangular quadrats (20 × 10 m each) sampled in a seasonal semideciduous forest fragment in the State of Sao Paulo, Brazil. The dataset includes total height and commercial bole height of trees, along with the spatial coordinates of the lower-left corner of each quadrat. It is organized to demonstrate the use of the function `BAPlot()` with rectangular quadrats.

**Usage**

```
data(quadrat3_rect.df)
```

**Format**

A data frame with 497 rows and 8 variables:

**Plot** Quadrat identifier.

**x** X coordinate (m) of the lower-left corner of the quadrat.

**y** Y coordinate (m) of the lower-left corner of the quadrat.

**Species** Scientific name of the species.

**Family** Botanical family of the tree.

**DBH** Diameter at breast height (cm).

**h** Total height of the tree (m).

**hcom** Commercial bole height of the tree (m).

---

 rarefaction

*Rarefaction Analysis*


---

### Description

Performs a rarefaction analysis, a method widely used in ecology to estimate species richness based on sample size. The function computes the expected number of species for increasing numbers of individuals, along with confidence intervals, following classical approaches by Hurlbert (1971), Heck et al. (1975), and related developments.

### Usage

```
rarefaction(
  formula,
  data,
  x,
  step = 1,
  points = NULL,
  prob = 0.95,
  xlab,
  ylab,
  plot = TRUE,
  theme = "theme_classic"
)
```

### Arguments

|         |  |
|---------|--|
| formula | An optional formula specifying the relationship between taxa and sample units (e.g., Taxon ~ Sample). If provided, the function extracts variables from data. A third variable may be included to remove dead individuals (e.g., Taxon ~ Sample - Dead).   |
| data    | A data frame containing the variables specified in formula ('long format'). It must contain one column representing the sample unit labels (e.g., quadrats or points) and one column representing the taxon names of the individual plants. This argument accepts the data frame used in the argument x in the function phytotparam. |
| x       | An optional contingency table of species (rows) by samples (columns). If not provided, it is calculated from formula and data. Alternatively, it can be a vector representing the number of individuals per species (see Examples).  |
| step    | Step size for the sequence of sample sizes in the rarefaction curve. Default is 1.   |
| points  | Optional vector of specific sample sizes (breakpoints) for which to calculate rarefaction. If NULL, a sequence from 1 to the total number of individuals is used.  |
| prob    | The confidence level for the confidence intervals. Default is 0.95.  |
| xlab    | Label for the x-axis of the plot (defaults to "Number of individuals").  |

|       |   |
|-------|---|
| ylab  | Label for the y-axis of the plot (defaults to "Number of species").   |
| plot  | Logical; if TRUE, a rarefaction curve is plotted. Default is TRUE.  |
| theme | Character string with the name of a ggplot2 theme to be applied to the plot (e.g., "theme_light", "theme_bw", "theme_minimal"). Default is "theme_classic". |

## Details

Rarefaction analysis provides a standardized way to compare species richness among samples of different sizes. It is based on probabilistic resampling without replacement and produces an expected species accumulation curve. Confidence intervals are calculated following variance estimators proposed by Heck et al. (1975) and Tipper (1979).

The function accepts data in three formats:

- long format (formula + data arguments),
- contingency matrix,
- vector of individuals per species.

Dead individuals can be excluded by specifying an additional term in the formula.

## Value

A data frame with the following components:

- n: number of sample units.
- s: mean number of species.
- lower: lower confidence interval bound.
- upper: upper confidence interval bound.

If `plot = TRUE`, a rarefaction curve with confidence ribbons is produced using **ggplot2**.

## Author(s)

Rodrigo Augusto Santinelo Pereira <raspereira@usp.br>

## References

- Colwell, R. K., Mao, C. X., & Chang, J. (2004). Interpolating, extrapolating, and comparing incidence-based species accumulation curves. *Ecology*, 85(10), 2717–2727. doi:10.1890/030557
- Heck, K. L., Van Belle, G., & Simberloff, D. (1975). Explicit calculation of the rarefaction diversity measurement and the determination of sufficient sample size. *Ecology*, 56(6), 1459–1461. doi:10.2307/1934716
- Hurlbert, S. H. (1971). The nonconcept of species diversity: A critique and alternative parameters. *Ecology*, 52(4), 577–586. doi:10.2307/1934145
- Tipper, J. C. (1979). Rarefaction and rarefaction—The use and abuse of a method in paleoecology. *Paleobiology*, 5(4), 423–434. doi:10.1017/S0094837300016924

**Examples**

```
## Using 'formula' (long format)
rarefaction(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  plot = TRUE
)

## Using different plot themes
rarefaction(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  plot = TRUE,
  theme = "theme_light"
)
rarefaction(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  plot = TRUE,
  theme = "theme_bw"
)
rarefaction(
  formula = Species ~ Plot - Morta,
  data = quadrat.df,
  plot = TRUE,
  theme = "theme_minimal"
)

## Using a matrix (wide format)
data.matrix <- with(
  quadrat.df,
  table(Plot, Species, exclude = "Morta")
)
rarefaction(x = data.matrix, plot = TRUE)

data.matrix <- as.matrix(
  xtabs(~ Plot + Species, data = quadrat.df, exclude = "Morta")
)
rarefaction(x = data.matrix, plot = TRUE)

## Using a vector
data.vector <- sort(
  as.vector(apply(data.matrix, 2, sum)),
  decreasing = TRUE
)
rarefaction(x = data.vector, plot = TRUE)

## Using breakpoints
pts <- c(1, 10, 30, 50, 80)
rarefaction(
  formula = Species ~ Plot - Morta,
```

```

data = quadrat.df,
points = pts,
plot = TRUE
)
rarefaction(x = data.matrix, points = pts, plot = TRUE)
rarefaction(
  x = data.vector,
  points = pts,
  plot = TRUE,
  theme = "theme_light"
)
rarefaction(x = data.vector, points = 50, plot = FALSE)

```

stats

*Representativeness and confidence statistics for a forest inventory***Description**

Computes representativeness and confidence statistics for a forest inventory. Summarizes inventory coverage, estimates absolute stand parameters per hectare with uncertainty (density, volume, basal area), and extrapolates results to the total habitat area, including sample-size requirements for target permissible errors.

**Usage**

```
stats(obj, area.tot, prob = 0.95, shape.factor = 1, rm.dead = FALSE)
```

**Arguments**

|              |  |
|--------------|--|
| obj          | An object of class <code>param</code> created by <code>phytoparam</code> . Must contain the raw data ( <code>obj\$data</code> ), variable mappings ( <code>obj\$vars</code> ), and global inventory metadata ( <code>obj\$global</code> ). |
| area.tot     | Total habitat area (in hectares) to which estimates will be extrapolated (e.g., the whole forest fragment). <b>Required.</b>   |
| prob         | Confidence level used to compute $t$ -based confidence limits (default 0.95).  |
| shape.factor | Stem form correction factor used in the individual volume calculation ( $V_i = abi \times h \times shape.factor$ ). Default 1 (cylindrical shape).   |
| rm.dead      | Logical. If TRUE, dead trees are excluded from all calculations (rows whose taxon label matches the “dead” code stored in <code>obj\$vars</code> ).  |

**Details**

Extracts the sample-unit (SU) identifier, taxon label, height/length variable, and “dead” code from `obj$vars`, and the surveyed (inventoried) area from `obj$global`. If `rm.dead = TRUE`, individuals flagged as dead are removed before analysis.

Individual volume is computed as  $V_i = abi \times h \times shape.factor$ , where `abi` and `h` are columns present in `obj$data`. Per-SU totals of volume and basal area are converted to per-hectare values

using the SU area. The function then derives, for density (A<sub>De</sub>), volume (A<sub>Vol</sub>), and basal area (A<sub>BA</sub>):

- mean, variance, standard deviation, standard error;
- coefficient of variation (CV);
- absolute and relative sampling error (based on the *t* quantile with |SU| - 1 df);
- lower and upper confidence limits per hectare.

Inventory representativeness (total area, number/area of SUs, and percentage inventoried) is reported, and population totals (for `area.tot`) are produced with corresponding confidence limits. Required numbers of SUs to attain 10% and 20% permissible relative errors are computed from standard finite-population sampling formulae.

### Value

A list with three components:

- `inventory`: data frame summarizing inventory coverage (fields `Stat` and `Values`).
- `ha`: data frame of per-hectare statistics for stand parameters (fields `Stat`, `ADe`, `AVol`, `ABA`).
- `population`: data frame of population-level statistics for the total area (`area.tot`), including required numbers of SUs for 10% and 20% permissible errors.

### Note

- Representativeness is reported both as inventoried area (`ha`) and as percentage of the total area.
- The number of SUs required for a target permissible error depends on the observed variance, the SU area, and the chosen confidence level (`prob`).

### Author(s)

Rodrigo Augusto Santinelo Pereira <raspereira@usp.br>

### References

FAO (1981). *Manual of Forest Inventory—With Special Reference to Mixed Tropical Forests*. Food and Agriculture Organization of the United Nations, Rome.

### Examples

```
## Creating the param object containing the phytosociological parameters
quadrat.param <- phytoparam(x = quadrat.df, measure.label = "CBH",
                           taxon = "Species", dead = "Morta", family = "Family",
                           circumference = TRUE, su = "Plot", height = TRUE,
                           su.size = 25, rm.dead = FALSE)

## Calculating the statistics
stats(obj = quadrat.param, area.tot = 4)
```

stratvol

*Stratified wood volume by DBH classes***Description**

stratvol computes wood volume ( $m^3 ha^{-1}$ ) stratified by diameter at breast height (DBH, in centimeters) classes for each taxon in a forest inventory. Individual tree volume is calculated as  $V_i = ABi \times h \times shape.factor$ , where  $ABi$  is the individual basal area at breast height and  $h$  is tree height. Volumes are then summed within DBH classes and standardized per hectare using the inventoried area stored in obj.

**Usage**

```
stratvol(obj, classes = 20, shape.factor = 1, rm.dead = FALSE)
```

**Arguments**

|              |  |
|--------------|--|
| obj          | An object of class "param" produced by <a href="#">phytoparam</a> , containing the inventory data, variable names, and global area statistics used for standardization.    |
| classes      | Numeric vector of breakpoints (in centimeters) defining the DBH classes. If a single value is supplied, two classes are formed ( $\leq$ value; $>$ value). Defaults to 20. |
| shape.factor | Stem form correction factor used in the individual volume calculation ( $V_i = ABi \times h \times shape.factor$ ). Default 1 (cylindrical shape).                         |
| rm.dead      | Logical. If TRUE, individuals labeled as dead (rows whose taxon string equals the <i>dead</i> code stored in obj\$vars) are excluded from all calculations. Default FALSE. |

**Details**

- DBH classes are defined from the numeric breakpoints provided in classes, using closed–open intervals internally and labeled for readability as:  $\leq a$ ,  $]a-b]$ ,  $\dots$ ,  $> z$  (all in centimeters). - Individual volume is computed as  $V_i = ABi \times h \times shape.factor$ . Summed volumes per class are divided by the inventoried area (in hectares) retrieved from obj\$global, yielding  $m^3 ha^{-1}$ .

**Value**

A data.frame with one row per taxon and the following columns:

- Taxon: Taxon label.
- One column per DBH class (labeled in centimeters), containing the summed wood volume per hectare ( $m^3 ha^{-1}$ ) for that taxon within the class. Missing combinations are returned as 0.

**Note**

- **Units:** Supply size classes in centimeters. The function assumes ABi and h yield volume in cubic meters before per-hectare standardization.
- **Class limits:** Class breakpoints should not exceed the maximum observed DBH; otherwise the function stops with an error.

**Author(s)**

Rodrigo Augusto Santinelo Pereira (<raspereira@usp.br>)

**References**

FAO (1981). *Manual of forest inventory—With special reference to mixed tropical forests*. Food and Agriculture Organization of the United Nations.

**See Also**

[phytoparam](#)

**Examples**

```
# Creating the 'param' object with phytosociological parameters
point.param <- phytoparam(x = point.df, measure.label = "CBH",
                          taxon = "Species", dead = "Morta", family = "Family",
                          circumference = TRUE, su = "Point", height = TRUE,
                          quadrat = FALSE, d = "Distance", rm.dead = FALSE)

# Stratified volumes with a single breakpoint (<= 20 cm; > 20 cm)
stratvol(point.param, classes = 20)

# Stratified volumes with multiple classes (<= 5], ]5-10], > 10 cm)
stratvol(point.param, classes = c(5, 10))

# Using a taper/form correction factor and excluding dead trees
stratvol(point.param, classes = c(10, 20, 30), shape.factor = 0.7, rm.dead = TRUE)
```

---

summary.param

*Summarize global phytosociological parameters*

---

**Description**

Display a concise summary of the global parameters computed by [phytoparam](#). If family-level outputs are present (i.e., the string "N. of families" occurs in the first column of object\$global), the first seven rows are shown; otherwise, the first six rows are shown.

**Usage**

```
## S3 method for class 'param'  
summary(object, ...)
```

**Arguments**

|        |   |
|--------|---|
| object | An object of class param returned by <a href="#">phytoparam</a> . |
| ...    | Ignored.  |

**Details**

Row names of `object$global` are removed before printing. The function is intended for quick inspection of the main global metrics.

**Value**

Used mainly for its side effect of printing to the console. Invisibly returns the displayed data frame.

**See Also**

**PhytoIn** ([phytoparam](#), [plot.param](#)).

**Examples**

```
res <- phytoparam(x = quadrat.df, measure.label = "CBH",  
                 taxon = "Species", family = "Family",  
                 su = "Plot", su.size = 25)  
summary(res) # calls summary.param (S3)
```

# Index

## \* datasets

- point.df, [14](#)
- quadrat.df, [15](#)
- quadrat2\_plot.df, [16](#)
- quadrat2\_tree.df, [16](#)
- quadrat3\_rect.df, [17](#)

AGB, [2](#), [12](#)

BAplot, [5](#)

collector.curve, [7](#)

computeAGB, [4](#)

correctTaxo, [4](#)

getWoodDensity, [4](#)

phytoparam, [8](#), [10](#), [13](#), [14](#), [21](#), [23–25](#)

plot.param, [12](#), [13](#), [25](#)

point.df, [14](#)

quadrat.df, [15](#)

quadrat2\_plot.df, [16](#)

quadrat2\_tree.df, [16](#)

quadrat3\_rect.df, [17](#)

rarefaction, [18](#)

stats, [12](#), [21](#)

stratvol, [12](#), [23](#)

summary.param, [12](#), [14](#), [24](#)