

# Package ‘biometria’

December 16, 2020

**Type** Package

**Title** Functions for biometrics, applied statistics, and ecological data analysis

**Version** 0.1

**Date** June 3, 2020

**Authors** Christian Salas-Eljatib, Joaquin Riquelme, Nicolas Pino, and Renato Cifuentes.

**Maintainers**

Nicolas Pino Cea <n.pino.cea@gmail.com>, Christian Salas-Eljatib <csejlatib@gmail.com>

**Description** Provides several functions for biometrics, applied statistics, and ecological analysis, It helps to accompany applied statistics analysis and exercises beside several data sets are also included.

**License** GPL-2

**NeedsCompilation** no

**RoxygenNote** 7.1.0

**Encoding** UTF-8

**Imports** bootstrap, data.table, dplyr, ff, geoR, ggplot2, gtools, httr, ineq, lattice, lubridate, magrittr, spatstat, stats, graphics, csabounds, methods, nlme, lme4, utils, grDevices, vctrs, tidyr

**Depends** R (>= 2.10)

## R topics documented:

biometria-package . . . . .	4
a.cdf . . . . .	4
a.contrast . . . . .	5
airquality . . . . .	6
annualppCities . . . . .	6
araucaria . . . . .	7
assignDomi . . . . .	8
assignLPclassi . . . . .	9
assignspp . . . . .	10
assignsTol . . . . .	11
asypm . . . . .	12
asypmOri . . . . .	13
baiTreelines . . . . .	14
bal . . . . .	15
balmodi . . . . .	16

bat . . . . .	17
bears . . . . .	18
bears2 . . . . .	19
beta.rich.si . . . . .	20
biomass . . . . .	21
carbohydrateTreelines . . . . .	21
chicksw . . . . .	23
ClarkE . . . . .	23
comparCorreFunc . . . . .	24
creaIncrData . . . . .	25
curtis . . . . .	27
deleteRight . . . . .	28
demograph . . . . .	28
descstat . . . . .	29
election . . . . .	30
eucaleaf . . . . .	31
expMod . . . . .	32
expModV1 . . . . .	33
expModV2 . . . . .	34
extractRight . . . . .	35
fancypairs.panel . . . . .	36
fertilizaexpe . . . . .	38
findColumn.byname . . . . .	38
fishgrowth . . . . .	39
floraChile . . . . .	40
forestHawaiian . . . . .	41
geomean . . . . .	42
gompertz . . . . .	43
gompertzMod . . . . .	44
gtree . . . . .	45
hawaii . . . . .	46
hossfeld . . . . .	47
huang . . . . .	48
iccMeff . . . . .	49
idahohd . . . . .	49
inequality . . . . .	50
interp.a . . . . .	52
interp.b . . . . .	52
invasivesRCI . . . . .	53
inverse . . . . .	54
kurt.coef . . . . .	55
lleuque . . . . .	55
loaded.pack . . . . .	56
loessfit . . . . .	56
logistic . . . . .	57
logisticMod . . . . .	58
lrt . . . . .	59
meyer . . . . .	60
mmenten . . . . .	61
moda . . . . .	62
naslund . . . . .	62
nele.list . . . . .	63

nnDIST	64
orange	65
pinaster	66
pinusContorta	67
pinusSpp	68
plantsHawaiian	69
plotLleuque	70
poder	70
poissonWeibull	71
presenceIce	72
prodan	73
pspLlancahue	74
ptaeda	75
pvalue.t	75
pvalue.z	76
qmd	77
radiatapl	77
radiosCopa	78
ranking2	79
ratkowsky	79
rdRoraco	80
regeneraNothofagus	81
richards	82
schnute	83
schumacher	84
sibbesen	85
simula	86
sizeDist.byGroupAllPlot	87
sizeDistPlot	89
skew.coef	91
sludge	92
snaspeChile	93
spatAustria	94
spatdistplot	95
speciesList	97
sppAbundance	98
sppTraits	99
stage	100
standVar	101
stat.rss	102
stat.st	103
stat.sys	103
statVar	104
timeSerPlot	105
topHeight	107
topHeight.tradi	108
trailCameraTrap	108
traits	109
treegrowth	110
treevol	110
treevolroble	111
treevolruca	112

Uestimator . . . . .	113
Uestimator.alt . . . . .	114
valesta . . . . .	115
valida.gr . . . . .	116
weibull . . . . .	117
wykoﬀ . . . . .	118
<b>Index</b>	<b>119</b>

---

biometria-package	<i>Functions for biometrics, applied statistics, and ecological data analysis</i>
-------------------	---

---

**Description**

Provides several functions for biometrics, applied statistics, and ecological analysis, It helps to accompany applied statistics analysis and exercises beside several data sets are also included.

**Details**

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.  
~~ An overview of how to use the package, including the most important functions ~~

**Author(s)**

NA  
Maintainer: NA

**References**

~~ Literature or other references for background information ~~

---

a.cdf	<i>Computes the cumulative distribution for a given random variable.</i>
-------	--

---

**Description**

Computes the cumulative distribution for a given random variable.

**Usage**

a.cdf(X = X)

**Arguments**

X                      a vector of a random variable

**Value**

Returns a dataframe having two columns: the first is the random variable and the second the cumulative distribution for the random variable.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
Y<- rnorm(10)
a.cdf(Y)
```

---

a.contrast	<i>Gives the statistics for statistical inference</i>
------------	---

---

**Description**

Gives the statistics for statistical inference for a given contrast

**Usage**

```
a.contrast(mod, coef.cont, g.means, g.ns, alpha, full)
```

**Arguments**

mod	object containing the fitted model
coef.cont	vector with the coefficients to establish the contrasts
g.means	mean per group
g.ns	sample size per group
alpha	is the confidence level in percentage for building the confidence intervals
full	0 if want short output, 1 for longer (i.e. more details)

**Value**

This function computes the Estimated contrast, standard error of the contrast, and confidence interval of the contrast, subject a given alpha level

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#not yet implemented
```

---

airquality	<i>Contains information of data airquality of datasets library.</i>
------------	---

---

**Description**

Daily air quality measurements in New York, May to September 1973.

**Usage**

```
data(airquality)
```

**Format**

Contains 6 variables, as follows:

**ozone** numeric Ozone (ppb).  
**solar** numeric Solar R (lang).  
**wind** numeric Wind (mph).  
**temp** numeric Temperature (degrees F).  
**month** numeric Month (1–12).  
**day** numeric Day of month (1–31).

**Source**

The data were provided from datasets library datasets.

**References**

Chambers J, Cleveland W, Kleiner B, Tukey P. 1983. Graphical Methods for Data Analysis. Belmont. CA: Wadsworth.

**Examples**

```
data(airquality)  
head(airquality)
```

---

annualppCities	<i>Contains information of annual precipitations in cities of Chile.</i>
----------------	--

---

**Description**

Data contains annual precipitations in six cities in Chile (Santiago, Talca, Chillan, Temuco, Valdivia, and Puerto Montt) at different years.

**Usage**

```
data(annualppCities)
```

**Format**

The data frame contains three variables as follows:

**city** Name of city.

**year** Year of registry.

**annual** Value of the annual precipitation of a given year (mm).

**Source**

The data were provided from <http://explorador.cr2.cl/>.

**Examples**

```
data(annualppCities)
head(annualppCities)
```

---

araucaria	<i>Contains plot-level variables in Araucaria araucana forests in southern Chile.</i>
-----------	---

---

**Description**

These are plot-level measurement data from the Araucaria araucana forests in the Araucania region in southern Chile, measured in 2009. The data inventory was based on fixed-area plots of 1000 m<sup>2</sup>. They are two forest stands.

**Usage**

```
data(araucaria)
```

**Format**

Contains plot-level variables as follows:

**stand** Stand number

**plot.no** Plot sample identificator number

**x.utm** UTM coordinate in X-axis, in km

**y.utm** UTM coordinate in Y-axis, in km

**slope** Slope, in %

**aspect** Aspect, in degrees

**eleva** Elevation, in msnm

**nha** Tree density, in trees/ha

**gha** Basal area, in m<sup>2</sup>/ha

**hdom** Dominant height, in m

**vha** Gross stand volume, in m<sup>3</sup>/ha

**dg** Diameter of the average basal area tree of the plot, in cm

## Source

The data are provided courtesy of Dr. Nelson Ojeda at the Universidad de La Frontera (Temuco, Chile).

## References

Salas C, Ene L, Ojeda N, Soto H. 2010. Metodos estadisticos parametricos y no parametricos para predecir variables de rodal basados en Landsat ETM+: una comparacion en un bosque de Araucaria araucana en Chile [Parametric and non-parametric statistical methods for predicting plotwise variables based on Landsat ETM+: a comparison in an Araucaria araucana forest in Chile]. Bosque 31(3): 179-194.

## Examples

```
data(araucaria)
head(araucaria)
```

---

assignDomi

*Assing dominance of species by tree density and stand basal area.*

---

## Description

This function assings dominance of species by tree density and stand basal area.

## Usage

```
assignDomi(data = data, num.def.dom = 3)
```

## Arguments

data	a dataframe having the stand-level variables per plot. The style of the file must be "nha.sppname1" and "gha.sppname1" to represent tree density and basal area of "sppname1", respectively. Notice that each row must be a unique sample plot.
num.def.dom	the number of positions that define dominance. By default is equal to 3, and the possibles values are 1, 2, or 3.

## Value

This function returns a dataframe having the following columns: plot.id, list.of.species, dom.by.nha and dom.by.gha.

## Note

Be sure to prepare the input data as requested.

## Author(s)

Christian Salas-Eljatib



## Examples

```
n <-7; plot <- 1:n
nha <- runif(n, 850, 2500); gha <- runif(n, 20, 45)
nha.Roble <- c(0.5, 0.2, 0.1,0.4,0,0,0)*nha
gha.Roble <- c(0.7, 0.2,0.1,0.3,0,0,0)*gha
nha.Rauli <- c(0.35, 0.6,0.2,0.5,0,0,0.5)*nha
gha.Rauli<- c(0.15, 0.6,0.2,0.3,0,0,0.5)*gha
nha.Coigue <- c(0.15, 0.2,0.7,0.1,0,0,0.2)*nha
gha.Coigue<- c(0.1, 0.2,0.7,0.4,0,0,0.3)*gha
nha.Laurel <- c(0.05, 0.02,0.2,0,0,0,0)*nha
gha.Laurel<- c(0.1, 0.04,0.01,0,0,0,0)*gha
nha.Alerce <- c(0, 0,0,0,50,0,0)
gha.Alerce<- c(0, 0,0,0,60,0,0)
nha.Canelo <- c(0, 0,0,0,850,0,0)
gha.Canelo<- c(0, 0,0,0,10,0,0)
nha.Araucaria <- c(0, 0,0,0,0,300,0)
gha.Araucaria<- c(0, 0,0,0,0,35,0)
nha.Nhirre <- c(0, 0,0,0,0,600,0)
gha.Nhirre<- c(0, 0,0,0,0,10,0)
nha.Tepa <- c(0, 0,0,0,0,1000,50)
gha.Tepa<- c(0, 0,0,0,0,5,4)
nha.Lenga <- c(0, 0,0,0,0,500,100)
gha.Lenga<- c(0, 0,0,0,0,15,4)
nha.Manhio <- c(0, 0,0,0,0,0,50)
gha.Manhio<- c(0, 0,0,0,0,0,25)
df <- data.frame(plot,nha,nha.Alerce,nha.Araucaria,
                 nha.Roble,nha.Coigue,nha.Rauli,
                 nha.Tepa,nha.Laurel,nha.Lenga,nha.Manhio,
                 nha.Canelo,nha.Nhirre,gha,gha.Alerce,
                 gha.Araucaria,gha.Coigue,
                 gha.Roble,gha.Rauli,gha.Tepa,gha.Laurel,
                 gha.Lenga,gha.Manhio,gha.Canelo,
                 gha.Nhirre)

head(df)
assignDomi(data=df)
assignDomi(data=df, num.def.dom = 2)
assignDomi(data=df, num.def.dom = 1)
```

---

assignLPclassi

Assings Luebert and Pliscoff vegetative floor

---

## Description

Returns a string vector containing Luebert and Pliscoff vegetative floor abbreviation by taking the vegetative floor id

## Usage

```
assignLPclassi(data = data, lyp.id = "lyp.id")
```

## Arguments

data	data frame having data
lyp.id	column name having Luebert and Pliscoff vegetative floor

**Details**

No details are given

**Value**

This function returns a data frame with one column containing Lueber and Pliscoff vegetative floor, abbreviated

**Author(s)**

Joaquin Riquelme

**References**

Luebert F, Pliscoff P, 2006. Sinopsis bioclimatica y vegetacional de Chile. Editorial Universitaria, Santiago, Chile.

**Examples**

```
df <- data.frame(code.id=c(61,48,53,54,56,58,59,61,62))
assignLPclassi(data = df, lyp.id = "code.id")
```

---

assignspp	<i>creates a dataframe including scientific and common name of species</i>
-----------	--

---

**Description**

creates a new dataframe that includes columns with scientific and common name of species, abbreviation and code of scientific name.

**Usage**

```
assignspp(data = data, name.esp = name.esp, cod = "yes")
```

**Arguments**

data	data frame having tree data, including species common names
name.esp	column name having the species common names
cod	user must indicate whether the codified name is provided in name.esp ('yes') or not ('no')

**Details**

No details are given

**Value**

This function returns a data frame including scientific and common name of species, abbreviation and code of scientific name.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
# How to use this function

#data(speciesList) #Loads list of tree species from the datasets

# Sample dataframe
# solo con la columna esp
# db1 <- data.frame(narb=1:100,esp=sample(x = c("nob","nal"), size = 100, replace = TRUE))
# con spp.name pero con otro nombre
#db2 <- data.frame(narb=1:100,spp.nammm=sample(x = c("Roble","Rauli"), size = 100, replace = TRUE))
# esp y spp.name
#db3 <- data.frame(narb=1:100,
#                  esp=sample(x = c("nob","nal"), size = 100, replace = TRUE),
#                  spp.name=sample(x = c("Roble","Rauli"), size = 100, replace = TRUE))

#e1 <- assignspp(data = db1, name.esp = "esp", cod = c("yes"))
#head(e1)

#e2 <- assignspp(data = db2, name.esp = "spp.nammm", cod = c("no"))
#head(e2)

#e3 <- assignspp(data = db3, name.esp = "esp", cod = c("yes"))
#head(e3)
```

---

assignsTol

*creates a dataframe including species shade-tolerance*


---

**Description**

creates a new dataframe that includes columns with species shade-tolerance.

**Usage**

```
assignsTol(data = data, name.esp = name.esp)
```

**Arguments**

data	data frame having tree data, including species code name (i.e., esp)
name.esp	column name having the species code name (i.e., esp)

**Details**

No details are given

**Value**

This function returns a data frame including information on shade tolerance of species.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#data(sppTraits) #tree species with their functional traits from the datasets

#Sample dataframe
#narb<- c(1:10)
#esp <- c("nob", "nal", "ec", "ls",
#         "pn", "sc", "la", "em", "lh", "lf");
#needTol<-data.frame(narb,esp)

#Using the function
#tol1 <- assignsTol(data = needTol, name.esp = "esp")
#head(tol1)
```

asyp

---

*Mathematical expression for the asymptotic function for relating variable Y versus X*

---

**Description**

Asymptotic function

**Usage**

```
asyp(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the asymptotic function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Note**

Please read the reference

**Author(s)**

Christian Salas-Eljatib.

**References**

- Pinheiro JC & DM Bates (2000) Mixed effects Models in S and Splus. Springer Verlag, New York, USA. 528p.

**Examples**

```
b0<-0.3
b1<-0.7
b2<-1.3
params<-c(b0,b1,b2)
X <- c(70)
y<-asyp(params,X,intercept=1.3)
plot(y~X, type="p")
```

---

asypOri	<i>Mathematical expression for an asymptotic function that pass through the origin</i>
---------	--

---

**Description**

Mathematical expression for an asymptotic function that pass through the origin

**Usage**

```
asypOri(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Asymptotic function that pass through the origin

**Value**

The values of the function for every values of the vector X and the parameters.

**Note**

Please read the reference for further details.

**Author(s)**

Christian Salas-Eljatib.

**References**

- Pinheiro JC & DM Bates (2000) Mixed-effects Models in S and Splus. Springer Verlag, New York, USA. 528p.

**Examples**

```
b0<-0.3
b1<-0.7
params<-c(b0,b1)
X<- c(70)
y<-asymptOri(params,X,intercept=1.3)
plot(y~X, type="p")
```

---

baiTreelines	<i>Contains information of annual basal area increment (BAI) for different species.</i>
--------------	---

---

**Description**

Dataset contains 157 observations, of the last 10 years in 6-8 adult trees of different species at three elevations of altitudinal gradients sampled in four locations of Chile and two in Spain.

**Usage**

```
data(baiTreelines)
```

**Format**

Contains 7 variables, as follows:

**climate** Climate of each location, mediterranean and temperate.

**site** Name of Location of study (termmas:Termas de Chillan , antillanca:Antillanca area within Puyehue National Park, castillo:Cerro Castillo Natural Reserve, farellones:Farellones in Central Chile, pyrenees: Sierra de Cutas area in Spanish Central Pyrenees,sierra:Sierra Nevada).

**species** name species of study (lenga: Nothofagus pumilio, frangel: Kageneckia angustifolia, uncinata: Pinus uncinata, sylvestris: Pinus sylvestris).

**elevation** Type of elevation. "Treeline", intermediate named as "inter", and closed or montane forest named as low.

**tree** Id for tree.

**bai** Value of annual basal area increment.

**mean.bai** Mean of annual basal area increment.

**Source**

The data were provided from DRYAD repository.

## References

Piper F, Vinegla B, Linares J, Camarero J, Cavieres L, Fajardo A. 2016. Mediterranean and temperate treelines are controlled by different environmental drivers. *Journal Ecology*. 104: 691-702. DOI:10.1111/1365-2745.12555

## Examples

```
data(baiTreelines)
head(baiTreelines)
```

---

bal	<i>Computes basal area in larger trees</i>
-----	--

---

## Description

Computes the basal area in larger (by basal area) trees for any given tree within a sample plot

## Usage

```
bal(data = data, dbh = dbh, plot.area = plot.area)
```

## Arguments

data	data frame having plot level data
dbh	column name having the diameter at breast height in centimeters
plot.area	column name containing the plot size in square meters

## Value

This function returns a data frame with two columns, the first is BAL and the second the percentile of BAL

## Author(s)

Arne Pommerening, slightly modified by Christian Salas-Eljatib

## Examples

```
#Creating an example dataframe
plot.no <- 1
area <- 1000 #in m2
n <- 13; species <- c(rep("Roble",n)); diam <- round(rnorm(n,25,20),1);
df<-data.frame(plot.no,area,species,diam)
n <- 19; species <- c(rep("Rauli",n)); diam <- round(rnorm(n,20,10),1);
df2<-data.frame(plot.no,area,species,diam)
df <- rbind(df,df2)
n <- 18; species <- c(rep("Coihue",n)); diam <- round(rnorm(n,30,10),1);
df2<-data.frame(plot.no,area,species,diam)
df <- rbind(df,df2)
n <- 15; species <- c(rep("Olivillo",n)); diam <- round(rexp(n,1/30),1);
df2<-data.frame(plot.no,area,species,diam)
```

```
df <- rbind(df,df2)
df <- subset(df,diam>5)

#Using the function
df$bal <- bal(data=df, dbh="diam", plot.area = "area")$bal
df$p.bal <- bal(data=df, dbh="diam", plot.area = "area")$p.bal
head(df)
```

---

balmodi

*Computes modified basal area in larger trees*


---

### Description

Computes a modification of the basal area in larger trees for any given tree within a sample plot, that depends on tree density and top height. This index was originally proposed by Schroder and Gadow (1999)

### Usage

```
balmodi(data = data, bal = bal, nha = nha, gha = gha, hdom = hdom)
```

### Arguments

data	data frame having plot level data
bal	column name having the basal area in larger trees (BAL)
nha	column name having tree density in trees per hectare
gha	column name having basal area of sample plot in square meters per hectare
hdom	column name containing top height in meters

### Value

The function returns an object, vector, the modified BAL

### Note

Please review the reference for further details.

### Author(s)

Created by Christian Salas-Eljatib

### References

Schroder J. and von Gadow K, 1999. Testing a new competition index for maritime pine in north-western Spain. Canadian Journal of Forest Research 29: 280-283.



## Examples

```
#Creating an example dataframe
BAL<-sort(c(0,runif(10,1,36),36.7))
n <- length(bal)
plot.id <- c(rep(1,n),rep(2,n))
TPH <- c(rep(1000,n),rep(600,n))
BA <- c(rep(37,n),rep(37,n))
Toph<- c(rep(25,n),rep(30,n))
df <- data.frame(plot.id,BAL,TPH,BA,Toph)
head(df)
#Using the function
df$balmodi <- balmodi(data=df, bal="BAL", nha = "TPH", gha = "BA", hdom = "Toph")
head(df)
```

---

bat	<i>Computes basal area in taller trees (BAT)</i>
-----	--

---

## Description

Computes the basal area in taller trees (BAT) for any given tree within a sample plot. The BAT is similar to BAL (basal area in larger trees), but here the reference variable is height instead of tree basal area.

## Usage

```
bat(data = data, dbh = dbh, h = h, plot.area = plot.area)
```

## Arguments

data	data frame having plot level data
dbh	column name having the diameter at breast height on trees in centimeters
h	column name having the height of trees in meters
plot.area	column name containing the plot size in square meters

## Value

This function returns a data frame with two columns, the first is BAT and the second the percentile of BAT

## Author(s)

Christian Salas-Eljatib

## Examples

```
#df<-data(pinaster)
#head(df)
#df$sup<-500
#bat(data=df,dbh='dbh',h = 'htot',plot.area = 'sup')
```

---

bears

*Age and physical measurement data for wild bears.*

---

### Description

Wild bears were anesthetized, and their bodies were measured and weighed. One goal of the study was to make a table (or perhaps a set of tables) for people interested in estimating the weight of a bear based on other measurements. This would be used because in the forest it is easier to measure the length of a bear, for example, than it is to weigh it. Notice that there are missing values for some of the variables.

### Usage

```
data(bears)
```

### Format

Contains individual-level variables, as follows:

**id** Bear id

**age** age in months

**month** Diameter at breast height, in cm

**sex** 1 =male, 2 = female

**headL** length of head, in cm

**headW** width of head, in cm

**neckG** girth of neck, in cm

**length** body length, in cm

**chestG** girth of chest, in cm

**weight** body weight, in kg

**obs** observation number for bear

**name** name given to bear

### Source

Minitab, Inc. The data description is courtesy of Prof. Timothy Gregoire at Yale University (USA).

### References

According to Prof. Gregoire, This data set was supplied by Gary Alt. Entertaining references are in Reader's Digest April, 1979, and Sports Afield September, 1981.

### Examples

```
data(bears)
head(bears)
```

---

bears2	<i>Age and physical measurement data for wild bears. Dataframe same as "bears" but without missing values.</i>
--------	--

---

## Description

Wild bears were anesthetized, and their bodies were measured and weighed. One goal of the study was to make a table (or perhaps a set of tables) for people interested in estimating the weight of a bear based on other measurements. This would be used because in the forest it is easier to measure the length of a bear, for example, than it is to weigh it.

## Usage

```
data(bears2)
```

## Format

Contains individual-level variables, as follows:

**id** Bear identifier  
**age** age in months  
**month** Diameter at breast height, in cm  
**sex** 1 =male, 2 = female  
**headL** length of head, in cm  
**headW** width of head, in cm  
**neckG** girth of neck, in cm  
**length** body length, in cm  
**chestG** girth of chest, in cm  
**weight** body weight, in kg  
**obs** observation number for bear  
**name** name given to bear

## Source

Minitab, Inc. The data description is courtesy of Prof. Timothy Gregoire at Yale University (New Haven, CT, USA).

## References

According to Prof. Gregoire, This data set was supplied by Gary Alt. Entertaining references are in Reader's Digest April, 1979, and Sports Afield September, 1981.

## Examples

```
data(bears2)  
head(bears2)
```

---

beta.rich.si	<i>Computes the value of the parameter beta of the Richards growth model, depending on site index and the base-age.</i>
--------------	---

---

### Description

Expression for computing the value of Beta in the Richards growth model, depending on the parameters alpha (a.param) and gamma (c.param), site index (sitio), and based-age (tb), for a polymorphic version of the growth model.

### Usage

```
beta.rich.si(a.param, c.param, site.index, tb)
```

### Arguments

a.param	is the estimated alpha parameter
c.param	is the estimated gamma parameter
site.index	is the value of site index
tb	is the base-age for the site index curves.

### Details

Please check that the resulting values of beta, are similar to the one fitted for the expected curve.

### Value

This function returns the value of the beta parameter, depending upon the value of site index, and the rest of the parameters already fitted for the Richards growth model.

### Author(s)

Christian Salas-Eljatib

### References

Garcia O. 1983. A stochastic differential equation model for the height growth of forest stands. Biometrics 39(4) 1059 1072.

### Examples

```
#Using the function  
beta.rich.si(a.param=50,c.param=1,site.index=25,tb=20)
```

---

biomass	<i>Contains tree-level biomass data for several species in Canada.</i>
---------	--

---

### Description

These are tree-level variables for several species in Canada.

### Usage

```
biomass
```

### Format

**treenum** tree number.  
**spp** species common name.  
**dbh** diameter at breast height, in cm.  
**height** total height, in m.  
**totbiom** total biomass, in kg.  
**bolebiom** stem biomass, in kg.  
**branchbiom** branches biomass, in kg.  
**foliagebiom** foliage biomass, in kg.

### Source

The data are provided courtesy of Prof. Timothy Gregoire at the School of Forestry and Environmental Studies at Yale University (New Haven, CT, USA).

### Examples

```
data(biomass)
head(biomass)
```

---

carbohydrateTreelines	<i>Contains information of carbohydrates concentrations .</i>
-----------------------	---

---

### Description

Dataset contains 863 observations, about of total soluble carbohydrate, starch, and non structural carbohydrates concentrations per mass unit and per volume unit, in three tissues in early summer and early autumn 6-8 adult trees of different specie at three elevations of altitudinal gradients sampled in four locations of Chile, and Spain.

### Usage

```
data(carbohydrateTreelines)
```

## Format

Contains 16 variables, as follows:

**climate** Climate of each location, mediterranean and temperate.

**site** Name of Location of study (termas:Termas de Chillan , antillanca:Antillanca area within Puyehue National Park, castillo:Cerro Castillo Natural Reserve, farellones:Farellones in Central Chile, pyrenees: Sierra de Cutas area in Spanish Central Pyrenees,sierra:Sierra Nevada).

**species** name species of study (lenga: Nothofagus pumilio, frangel: Kogeneckia angustifolia, uncinata: Pinus uncinata, sylvestris: Pinus sylvestris).

**tissue** Type of tissue, new developing twigs, stem sapwood and branches.

**time** Measurement season (spring or autumn).

**elevation** Type of elevation. "Treeline", intermediate named as "mid", and closed or montane forest named as "low".

**tree** Id for tree.

**tree.site** Id site for each location of study.

**tss** Value of concentrations soluble carbohydrate per mass unit.

**st** Value of concentrations starch per mass unit.

**nsc** Value of concentrations non structural carbohydrates per mass unit.

**tss.nsc** .

**wd** .

**tss.mv** Value of concentrations soluble carbohydrate per volume unit.

**st.mv** Value of concentrations starch per volume unit.

**nsc.mv** Value of concentrations non structural carbohydrates per volume unit.

## Source

The data were provided from DRYAD repository.

## References

Piper F, Vinegla B, Linares J, Camarero J, Cavieres L, Fajardo A. 2016. Mediterranean and temperate treelines are controlled by different environmental drivers. *Journal Ecology*. 104: 691-702. DOI:10.1111/1365-2745.12555

## Examples

```
data(carbohydrateTreelines)
head(carbohydrateTreelines)
```

---

chicksw

---

*Contains information of ChichWeigth data of alr4 library.*


---

### Description

The body weights of the chicks were measured at birth and every second day thereafter until day 20. They were also measured on day 21. There were four groups on chicks on different protein diets.

### Usage

```
data(chicksw)
```

### Format

Contains 4 variables, as follows:

**weight** a numeric vector giving the body weight of the chick (gm).

**time** a numeric vector giving the number of days since birth when the measurement was made.

**chick** an ordered factor with levels different giving a unique identifier for the chick. The ordering of the levels groups chicks on the same diet together and orders them according to their final weight (lightest to heaviest) within diet.

**diet** a factor with levels 1,2,3 and 4 indicating which experimental diet the chick received.

### Source

The data were provided from alr4 library of R.

### References

Crowder M, Hand D. 1990. Analysis of Repeated Measures. Chapman and Hall

### Examples

```
data(chicksw)
head(chicksw)
```

---

Clarke

---

*Computes the Clark and Evans aggregation index*


---

### Description

Computes the Clark and Evans aggregation index

### Usage

```
Clarke(pppData, plot.area)
```

**Arguments**

pppData            is a ppp object (the data with the Cartesian positions as an R object)  
 plot.area        is the area of the plot in the same units that the distance was measured

**Value**

Computes the Clark & Evans spatial statistic indice

**Note**

this takes a while to run for large datasets

**Author(s)**

Christian Salas-Eljatib

**References**

- Clark, P.J. and Evans, F.C. (1954) Distance to nearest neighbour as a measure of spatial relationships in populations. Ecology 35, 445-453.

**Examples**

```
# Load data with trees and their position (coordinates)

#aa <- data(araucaria)
#area.plot <- 500

# Create an object of class "ppp",
# representing the point pattern dataset in the two-dimensional plane
#df <- spatstat::ppp(aa$x.utm, aa$y.utm,
#                    c(min(aa$x.utm),max(aa$x.utm)),
#                    c(min(aa$y.utm),max(aa$y.utm)))

# Calculate Clark & Evans spatia
```

---

comparCorreFunc

*Compares variant models depending on the spatial correlation functions*

---

**Description**

Compares fitted models (gls or lme) using different spatial correlation functions with and without nugget effects. The main output is the model as an object.

**Usage**

```
comparCorreFunc(mod = mod, data = data, is.lme = TRUE)
```



**Arguments**

<code>mod</code>	a model object, fitter either using <code>gls()</code> or <code>lme()</code> .
<code>data</code>	the dataframe for fitting the object model.
<code>is.lme</code>	is a logical value to identify if the model was fitted by <code>lme</code> , otherwise (FALSE) the model "mod" must be fitted by <code>gls</code> . The default is set to TRUE.

**Value**

an object with fitting results.

**Author(s)**

Christian Salas-Eljatib.

**References**

Pinheiro JC, and Bates DM. 2000. Mixed-effects models in S and Splus. Springer-Verlag, New York, NY. 528 p.

**Examples**

```
#not yet implemented
```

---

<code>creaIncrData</code>	<i>Create increment data from time series observations</i>
---------------------------	--

---

**Description**

It creates an increment data from a time series dataframe. From a data containing measurements of variables on time, this function will create a period-wise dataframe for all the variables, depending upon the ID observation and the variable representing the temporal scale.

**Usage**

```
creaIncrData(
  data = data,
  time = time,
  unit.id = unit.id,
  multi.time = F,
  n.measu = NA,
  identifica.simple = NA,
  static.columns = NA
)
```

## Arguments

<code>data</code>	data frame having the time series data
<code>time</code>	column name having the time vector
<code>unit.id</code>	column name containing the info of the observational unit
<code>multi.time</code>	logical value for creating the data frame the default FALSE if you only want increment successive pairs, otherwise TRUE returns all possible time combinations
<code>n.measu</code>	logical value for adding a column with the total number of measurement on time for a given observational unit
<code>identifica.simple</code>	returns 1 when the incremente is a successive pair and 0 if it doesn't
<code>static.columns</code>	columns that do not change

## Details

Be careful with the output, uses with caution.

## Value

This function returns a data frame with an increment structure. The main columns are as follows: "tot.n.measure" represents the total number of observations on time for a given ID unit; the suffix ".0" is added to all the variables at the beginning (at time "t.0") of the period of analysis; the suffix ".1" is added to all the variables at the end (at time "t.1") of the period of analysis. If the option "identifica.simple" is set to be TRUE, the column "incr.sim" will appear in the output indicating with a number 1 those observations having the simple or traditional consecutive steps, and 0 otherwise.

## Note

In order to use this function, the package 'gtools' must be previously installed.

## Author(s)

Christian Salas-Eljatib and Joaquin Riquelme-Alarcon

## Examples

```
#Creating an example dataframe
id <- rep(1,4) #observation ID
year <- c(2000,2001,2002,2003) # time variable
age <- c(1, 2, 3, 4)
height <- c(10, 14, 15.8, 16.3)
spp <- rep("Nothofagus obliqua",4) # a static variable

df<-data.frame(id,year,age,height,spp)

#Alternative 1. increment-successive data
creaIncrData(data = df, time = "year", unit.id = "id", static.columns = "spp")

#Alternative 2. all possible combinations of increment data
creaIncrData(data = df, time = "year", unit.id = "id", static.columns = "spp", multi.time = TRUE)
creaIncrData(data = df, time = "year", unit.id = "id", multi.time = TRUE,
              n.measu = TRUE, identifica.simple = TRUE, static.columns = "spp")
```

---

curtis

---

*Mathematical expression for the Curtis's model to relate Y versus X*


---

**Description**

Curtis' function

**Usage**

```
curtis(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Robert Curtis' function

**Value**

The value of the function for every value of the vector X depending on the parameters.

**Note**

Please read the reference for further details.

**Author(s)**

Christian Salas-Eljatib.

**References**

- Curtis RO (1967) Height diameter and height diameter age equations for second growth Douglas. Forest Science 13(4) 365 375.

**Examples**

```
b0<- 37.68
b1<- 18.6
params<-c(b0,b1)
X <- c(70)
y<-curtis(params,X,intercept=1.3)
plot(y~X,type='p')
```

`deleteRight`*It deletes the last n-characters of a string from the right-hand side.*

---

**Description**

This function deletes the last n-characters of a string

**Usage**

```
deleteRight(fac, n)
```

**Arguments**

<code>fac</code>	is an object of class string or factor
<code>n</code>	is the number of characters to be deleted of a the string given in fac.

**Details**

As always, please check the output after applying the function.

**Value**

This function returns an object having n-less characters from the right-hand side.

**Author(s)**

Christian Salas-Eljatib.

**Examples**

```
last.names.id <- c("Stage-1924", "Gregoire-1958", "Robinson-1967")
deleteRight(last.names.id, 5)
deleteRight(last.names.id, 4)
```

---

`demograph`*Contains information of demography of species.*

---

**Description**

Dataset contains 61 observations about life histories values for each species and site, as obtained from the parameterization carried out in studies that used the model SORTIE

**Usage**

```
data(demograph)
```

**Format**

Contains 15 variables, as follows:

**sp** Name specie.  
**site** Name of site of study.  
**country** Name of country.  
**site.n** Code of site.  
**code** Code of specie.  
**genus** Genus of specie.  
**sps** Abbreviated name specie.  
**family** Family of specie.  
**phyl** Type of phylogeny.  
**l.hab** Type of leaf habit.  
**l.type** .  
**leaf** Type of leaf.  
**growth.l** Growth at full light (time in years).  
**growth.d** Growth in shade.  
**surv.d** Survival in shade.

**Source**

The data were provided from DRYAD repository

**References**

- Ameztegui A, Paquette A, Shipley B, Heym M, Messier C, Gravel D. 2016 . Shade tolerance and the functional trait: demography relationship in temperate and boreal forests. Functional Ecology, 31: 821-830. DOI:10.1111/1365-2435.12804

**Examples**

```
data(demograph)
head(demograph)
```

---

descstat

*A descriptive statistics table for continuous variables*


---

**Description**

It creates a descriptive statistics table for all continuous variables in a dataframe excluding missing values.

**Usage**

```
descstat(data = data, decnum = NA)
```

**Arguments**

<code>data</code>	a dataframe containing variables as columns
<code>decnum</code>	the number of decimals to be used in the output

**Details**

As always, please check the output after applying the function.

**Value**

This function wraps descriptive statistics into a summarize table having the following descriptive statistics: sample size, minimum, maximum, mean, median, SD, and coefficient of variation (

**Author(s)**

Christian Salas-Eljatib and Tomas Cayul.

**Examples**

```
#creating a fake dataframe
set.seed(1234)
df <- as.data.frame(cbind(variable1=rnorm(5, 0), variable2=rnorm(5, 2)))
## adding one missing value
df[3,1] <- NA
df
#using the function
descstat(data=df)
descstat(data=df,decnum=1)
descstat(df,2)
```

---

election

*Contains information of florida datasets of alr4 library .*

---

**Description**

County-by-county vote for president in Florida in 2000 for Bush, Gore and Buchanan.

**Usage**

```
data(election)
```

**Format**

Contains 3 variables, as follows:

**gore** Vote for Gore.

**bush** Vote for Bush.

**buchaman** Vote for Buchaman.

**Source**

The data were provided from alr4 library of R.

## References

Weisberg S. 2014. Applied Linear Regression. 4th edition. Hoboken NJ: Wiley

## Examples

```
data(election)
head(election)
```

---

eucaleaf

*Leaf measurements for Eucalyptus nitens trees in Tasmania, Australia.*

---

## Description

The length, width, and area of Eucalyptus nitens leaves were measured.

## Usage

```
data(eucaleaf)
```

## Format

Contains leaf-level variables, as follows:

**time** Early or Late

**tree** an identifier for a given sample tree

**shoot** shoot description

**l** length of the leaf, in mm

**w** width of the leaf, in mm

**la** leaf area, in cm<sup>2</sup>

## Source

Although the original source of the measurements is the Dissertation of Dr. Candy (1999), the data file used here was courtesy of Prof. Timothy Gregoire at Yale University (New Haven, CT, USA). Furthermore, these data were used by Gregoire and Salas (2008).

## References

- Candy SG. (1999). Predictive models for integrated pest management of the leaf beetle *Chrysophtharta bimaculata* in *Eucalyptus nitens* in Tasmania. Doctoral dissertation, University of Tasmania, Hobart, Australia.

- Gregoire TG, and Salas C. 2009. Ratio estimation with measurement error in the auxiliary variate. *Biometrics* 65(2):590-598

## Examples

```
data(eucaleaf)
head(eucaleaf)
```

---

expMod	<i>Mathematical expression for the exponential function for relating variable Y versus X</i>
--------	--

---

**Description**

Modificated-exponential

**Usage**

```
expMod(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the exponential function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**Examples**

```
b0<-0.1
b1<-0.2
b2<-0.3
b3<-0.4
b4<-0.5
params<-c(b0,b1,b2,b3,b4)
X <- seq(10,70,by=0.01)
y<-expMod(params,X,intercept=1.3)
plot(y~X,type='l')
```



---

expModV1	<i>Mathematical expression for the exponential function for relating variable Y versus X</i>
----------	--

---

**Description**

Modificated-exponential-v1

**Usage**

```
expModV1(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the exponential function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**Examples**

```
b0<-0.1
b1<-0.2
b2<-0.3
b3<-0.4
b4<-0.5
params<-c(b0,b1,b2,b3,b4)
X <- c(70)
y<-expModV1(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

expModV2	<i>Mathematical expression for the exponential function for relating variable Y versus X</i>
----------	--

---

## Description

Modificated-exponential-v2

## Usage

```
expModV2(params, X, intercept = NA)
```

## Arguments

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

## Details

Mathematical expression for the exponential function for relating variable Y versus X

## Value

The value of the function for every value of the vector X and the parameters.

## Author(s)

Christian Salas.

## Examples

```
b0<-0.1
b1<-0.2
b2<-0.3
b3<-0.4
params<-c(b0,b1,b2,b3)
X <- c(70)
y<-expModV2(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

extractRight	<i>Extracts the last n-characters of a string</i>
--------------	---

---

### Description

This function extracts the last n-characters of a string

### Usage

```
extractRight(fac, n)
```

### Arguments

fac	is an object (vector or scalar) of class string or factor
n	is the number of characters to be deleted of a the string given in 'fac'.

### Details

As always, please check the output after applying the function.

### Value

This function returns an having the last-n characters of the object 'fac'

### Note

This function returns a string.

### Author(s)

Christian Salas-Eljatib

### Examples

```
last.names.id <- c("Stage-1924","Gregoire-1958","Robinson-1967")
extractRight(last.names.id,5)
years <- extractRight(last.names.id,4)
years
```

---

fancypairs.panel

*Generates a multi panel dispersion plot panel with trend curves*


---

### Description

This function is a modified version of pairs.panels() from library(psych). It produces a panel of multiple dispersion plots.

### Usage

```
fancypairs.panel(
  x,
  smooth = TRUE,
  scale = FALSE,
  density = TRUE,
  ellipses = FALSE,
  digits = 2,
  method = "pearson",
  pch = 20,
  lm = FALSE,
  cor = TRUE,
  jiggle = FALSE,
  factor = 2,
  hist.col = "blue",
  show.points = TRUE,
  rug = TRUE,
  breaks = "Sturges",
  cex.cor = 1,
  wt = NULL,
  smoother = FALSE,
  stars = FALSE,
  ci = FALSE,
  alpha = 0.05
)
```

### Arguments

x	is a dataframe containing all the numeric variables to be used for drawing the panel plot
smooth	a logical value for drawing smooth curves. Defaults is set to TRUE.
scale	scales the correlation font by the size of the absolute correlation. Defaults is set to FALSE.
density	a logical value for drawing a density curve. Defaults is set to TRUE.
ellipses	a optional logical value for drawing an ellipse for the scatter-plots. Defaults is set to FALSE.
digits	an optional numeric value for the digits to be used for drawing the correlation coefficient in the panel. Defaults is set to 2.
method	a string giving the method to be used for computing the correlation coefficient. Defaults is set to "pearson".

pch	The plot character (defaults to 20 which is a '.').
lm	Plot the linear fit rather than the LOESS smoothed fits. The default is FALSE.
cor	If plotting regressions, should correlations be reported? The default is TRUE.
jiggle	Should the points be jittered before plotting? The default is FALSE.
factor	factor for jittering (1-5), therefore only needed if "jiggle" is set to TRUE.
hist.col	a string giving the color to be used for the histograms of the panel. Defaults is set to "blue".
show.points	a logical value for drawing the points in the scatter-plots. Defaults is set to TRUE.
rug	a logical value for drawing the rugs in the histograms. Defaults is set to TRUE.
breaks	a string giving the method to be used for obtaining the breaks of the histogram. Defaults is set to "Sturges".
cex.cor	If this is specified, this will change the size of the text in the correlations. this allows one to also change the size of the points in the plot by specifying the normal cex values. If just specifying cex, it will change the character size, if cex.cor is specified, then cex will function to change the point size.
wt	If specified, then weight the correlations by a weights matrix (see note for some comments)
smoother	If TRUE, then smooth.scatter the data points – slow but pretty with lots of subjects
stars	a logical value for drawing stars for the significance of the correlation coefficient. Defaults is set to FALSE.
ci	Draw confidence intervals for the linear model or for the loess fit, defaults to ci=FALSE. If confidence intervals are not drawn, the fitting function is lowess.
alpha	an optional numeric value for the significance level. Defaults is set to 0.05.

## Details

Notice that the data must contain only numeric variables

## Value

This function returns a multi-panel plot

## Author(s)

From "pairs.panels" (library "psych"), and modified by Christian Salas-Eljatib.

## Examples

```
#df <- data.frame(x=runif(1000),y=rnorm(1000),z=rbeta(1000,.2,2),x3=rnorm(1000,mean=43,sd=10))
#fancypairs.panel.(df)
```

---

fertilizaexpe	<i>Contains information of.</i>
---------------	---------------------------------

---

**Description**

Data contains.

**Usage**

```
data(fertilizaexpe)
```

**Format**

Contains 3 variables, as follows:

**years** Year at capture.

**length** Length at capture (mm).

**Source**

The data were provided.

**References**

not yet

**Examples**

```
data(fertilizaexpe)
head(fertilizaexpe)
```

---

findColumn.byname	<i>Finds the position of a specific variable</i>
-------------------	--

---

**Description**

Sometimes in data manipulation we are facing the task of locating the position of a specific variable within a dataframe. This function finds the position in which a column name is in an object (either a dataframe or vector).

Find the position of a specific variable, but can be used for more than one variable.

**Usage**

```
findColumn.byname(data = data, col.name = col.name)
```

**Arguments**

data                    is a dataframe

col.name                is a string specifying the name of the variable

**Value**

This function returns the number of a specific column-name.

**Note**

It can be used for a vector of specified column-names as well.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
df <- data.frame(varX=1:5, varY=letters[1:5], varZ=rep("a",5), varK=rep("b",5))
df
#using the function
findColumn.byname(df, c("varY", "varZ"))
findColumn.byname(df, "varK")
#Creating an example vector
vector <- letters
vector
findColumn.byname(vector, c("h", "z"))
```

---

fishgrowth

---

*Contains information of wblake datasets of alr4 library.*


---

**Description**

Data on samples of small mouth bass collected in West Bearskin Lake, Minnesota, in 1991. The file wblake includes only fish of ages 8 or younger.

**Usage**

```
data(fishgrowth)
```

**Format**

Contains 3 variables, as follows:

**years** Year at capture.

**length** Length at capture (mm).

**scale** radius of a key scale (mm).

**Source**

The data were provided from alr4 library of R.

**References**

Weisberg S. 2014. Applied Linear Regression. 4th edition. Hoboken NJ: Wiley

**Examples**

```
data(fishgrowth)
head(fishgrowth)
```

---

floraChile

*Contains information of.*

---

**Description**

Dataset contains E

**Usage**

```
data(floraChile)
```

**Format**

Contains xx variables, as follows:

**family** .

**genus** .

**scientific.name** .

**author** .

**origin** .

**life.form** .

**lat...** .

**Source**

The data were provided from Jan Bannister researcher at Institute National Forest in Chile (INFOR).

**References**

not yet

**Examples**

```
data(floraChile)
head(floraChile)
```



---

forestHawaiian	<i>Contains information of plants Hawaiians.</i>
----------------	--

---

## Description

Dataset contains 43590 observations,

## Usage

```
data(forestHawaiian)
```

## Format

Contains 18 variables, as follows:

**island** Island name.

**plot.id** Unique numeric identifier for each plot.

**study** Brief name of study.

**plot.area** Plot area in m2.

**longitude** Longitude of plot in decimal degrees; WGS84 coordinate system.

**latitude** Latitude of plot in decimal degrees; WGS84 coordinate system.

**year** Year in which plot data was collected.

**census** Numeric identifier for each census.

**tree.id** Unique numeric identifier for each individual.

**scientific.name** Genus and species of each individual following TPL v. 1.1.

**family** Family of each individual following TPL v. 1.1.

**angiosperm** Binary variable (1 = yes, 0 = no) indicating whether an individual is classified as an angiosperm following APG III.

**monocot** Binary variable (1 = yes, 0 = no) indicating whether an individual is classified as a monocot following APG III.

**native.status** Categorical variable ('native', 'alien', 'uncertain') indicating alien status of each individual following Wagner et al. (2005).

**cultivated.status** Binary variable (1 = yes, 0 = no, NA = not applicable) indicating if species is cultivated following PIER.

**abundance** Number of individuals (all = 1).

**abundance.ha** Abundance of each individual on a per hectare basis.

**dbh** Diameter at 1.3 m (DBH in cm) for each individual; NA indicates that size was not measured, but was classified by size class.

## Source

The data were provided from DRYAD repository.

## References

- Craven D, Knight T, Barton K, Bialic-Murphy L, Cordell S, Giardina C, Gillespie T, Ostertag R, Sack L, Chase J. 2018. OpenNahele: the open Hawaiian forest plot database. Biodiversity Data Journal 6: e28406. <https://doi.org/10.3897/BDJ.6.e28406>

**Examples**

```
data(forestHawaiian)
head(forestHawaiian)
```

---

`geomean`*Computes the geometric mean*

---

**Description**

This function computes the geometric mean

This function computes the PRESS statistics for a simple linear regression model

**Usage**

```
geomean(v)
```

```
press(x, y)
```

**Arguments**

<code>v</code>	is a numeric vector
<code>x</code>	is the predictor variable
<code>y</code>	is the response variable

**Details**

Be carefull for using it

**Value**

This function returns the geometrics mean, a numeric scalar.

This function returns the PRESS statistics.

**Note**

Init seed

**Author(s)**

Christian Salas-Eljatib.

Christian Salas-Eljatib.

**Examples**

```

set.seed(1234)
y <- rnorm(10, 45)
#using the function
geomean(y)

set.seed(1234)
x <- rnorm(10, 550)
y <- rnorm(10, 45)
#using the function
press(x,y)

```

---

gompertz	<i>Mathematical expression for the Gompertz function for relating variable Y versus X</i>
----------	---

---

**Description**

Gompertz

**Usage**

```
gompertz(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Gompertz function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Nokoe S (1978) Demonstrating the flexibility of the Gompertz function as yield model using mature speciesdata. *Commonw. For. Rev.* 57(1), 35-42.
- Schabenberger O & FJ Pierce (2002) *Contemporary Statistical Models for the Plant and Soil Sciences*. CRCPress, Boca Raton, FL, USA. 738 p.

**Examples**

```

b0<- 30.16
b1<- 0.06
b2<- 25.59
params<-c(b0,b1,b2)
X <- c(70)
y<-gompertz(params,X,intercept=1.3)
plot(y~X,type='p')

```

---

gompertzMod	<i>Mathematical expression for the Gompertz function for relating variable Y versus X</i>
-------------	---

---

**Description**

Gompertz Reparametrizado

**Usage**

```
gompertzMod(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Gompertz function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Grosenbaugh LR (1965) Generalization and reparameterization of some sigmoid and other non-linear functions. *Biometrics* 21(3), 708-714.
- Rawat AS & F Franz (1974) Detailed non-linear asymptotic regression studies on tree and stand growth with particular reference to forest yield research in Bavaria (Federal Republic of Germany) and India. In J. Fries (Ed.) *Growth Models for Tree and Stand Simulation*, Res.Note 30. Proc. IUFRO Working Party S4.01-4 Meetings, Dep. For. Yield Res., Royal Coll. For., Stockholm, Sweden, p. 180-221.
- Ricker WE (1979) Growth rates and models. In Hoar WS, DJ Randall & JR Brett (Eds.) *Fish Physiology. Volume VIII Bioenergetics and Growth*. Academic Press, New York, USA, p. 677-743.

**Examples**

```
b0<- 30.16
b1<- 0.06
b2<- 25.59
params<-c(b0,b1,b2)
X <- c(70)
y<-gompertzMod(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

**gtree***Computes basal area of any given tree*

---

**Description**

Computes basal area of any given tree. Actually provides the area for a given circle of radius x.

**Usage**

```
gtree(x, in.m2)
```

**Arguments**

x	is the vector having tree diameter, in cm
in.m2	is an indicator variable: 1 to obtain the result in m2 if x was measured in cm; and 0 to obtain the resulting area in the same units of x.

**Details**

No details are given

**Value**

The value of basal area.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#Creating an example dataframe
dbh<- round(rnorm(4,25,20),1);
#Using the function
in.m2=1
gtree(dbh,in.m2)
```

---

hawaii*Metrosideros polymorpha in Hawaii*

---

### Description

Data containing 64 observations at the current annual growth rate (defined as dbh increment within one calendar year) of each tree was measured from 1986 to 1988 using band dendrometers.

### Usage

```
data(hawaii)
```

### Format

The data frame contains eight variables as follows:

**tree.code** Tree number identification.

**dbh** Initial stem diameter, in cm.

**htot** Total height in m.

**crown.area** Crown outline area, in square meters.

**comp.ind** Competition index (Basal area of nearest neighbor divided by square of distance to nearest neighbor plus basal area of second nearest neighbor divided by square of distance to second nearest neighbor).

**cai.1986** Current annual stem diameter increment during 1986, in mm.

**cai.1987** Current annual stem diameter increment during 1987, in mm.

**cai.1988** Current annual stem diameter increment during 1988, in mm.

### Source

The data were provided from .

### References

Gerrish G, Mueller-Dombois D. 1999. Measuring stem growth rates for determining age and cohort analysis of a tropical evergreen tree. *Pacific Science*. 53(4): 418-429.

### Examples

```
data(hawaii)
head(hawaii)
```

---

hossfeld	<i>Mathematical expression for the Hossfeld function for relating variable Y versus X</i>
----------	---

---

**Description**

Hossfeld

**Usage**

```
hossfeld(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Hossfeld function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Mehthatalo L, S de Miguel & TG Gregoire (2015) Modeling height diameter curves for prediction. Canadian Journal of Forest Research 45 826 837.

**Examples**

```
b0<-47
b1<-0.03
b2<-0.79
params<-c(b0,b1,b2)
X <- c(70)
y<-hossfeld(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

huang	<i>Mathematical expression for the Huang function for relating variable Y versus X</i>
-------	--

---

**Description**

Huang

**Usage**

```
huang(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Huang function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**Examples**

```
b0<-0.1
b1<-0.2
b2<-0.3
params<-c(b0,b1,b2)
X <- c(70)
y<-huang(params,X,intercept=1.3)
plot(y~X,type='p')
```



---

`iccMeff`*Computes the intra-class correlation of a mixed-effects model*

---

**Description**

This function computes the intra-class correlation of a mixed-effects model

**Usage**

```
iccMeff(mod, num.f = NA)
```

**Arguments**

<code>mod</code>	is an object obtained by fitting a linear or non-linear mixed-effects model.
<code>num.f</code>	number of factors (default = 2)

**Value**

This function returns the ICC for a given fitted mixed-effects model.

**Note**

In order to use this function, you most likely will need to load the nlme or the lme4 packages.

**Author(s)**

Christian Salas-Eljatib.

**Examples**

```
#Not yet implemented
```

---

`idahohd`*Contains information of data ufc of alr4 library.*

---

**Description**

These data are forest inventory measures from the Upper Flat Creek stand of the University of Idaho Experimental Forest, dated 1991.

**Usage**

```
data(idahohd)
```

**Format**

Contains 5 variables, as follows:

**plot** plot number.

**tree** tree within plot.

**species** a factor with levels DF = Douglas-fir, GF = Grand fir, SF = Subalpine fir, WL = Western larch, WC = Western red cedar, WP = White pine.

**dbh** Diameter 137 cm perpendicular to the bole, mm.

**height** Height of the tree, in decimeters.

**Source**

The data were provided from alr4 library of R.

**References**

Weisberg S. 2014. Applied Linear Regression. 4th edition. New York: Wiley.

**Examples**

```
data(idahohd)
head(idahohd)
```

---

inequality	<i>calculate different index about inequality of a variable, for example IBA,GINI,Atkinson.</i>
------------	---

---

**Description**

Index of inequality

**Usage**

```
inequality(
  data = data,
  dbh = dbh,
  superficie = superficie,
  tolerancia = NA,
  esp = esp,
  ref.og.nha = NA,
  ref.og.gha = NA,
  ref.og.gha.80 = NA,
  ref.og.gha.tol = NA,
  ref.og.gini = NA,
  ref.sg.nha = NA,
  ref.sg.gha = NA,
  ref.sg.gha.80 = NA,
  ref.sg.gha.tol = NA,
  ref.sg.gini = NA
)
```

**Arguments**

data	dataframe with variables
dbh	diameter of tree
superficie	surface of plot
tolerancia	type of tolerant at shade
esp	especie of tree
ref.og.nha	column name containing reference value of density plot for forest old growth
ref.og.gha	column name containing reference value of basal area plot for forest old growth
ref.og.gha.80	column name containing reference value of density plot for forest old growth
ref.og.gha.tol	column name containing reference value of density plot for forest old growth
ref.og.gini	column name containing reference value of gini coefficient of plot for forest old growth
ref.sg.nha	column name containing reference value of density plot for forest secondary growth
ref.sg.gha	column name containing reference value of basa area plot for forest secondary growth
ref.sg.gha.80	column name containing reference value of density plot for forest secondary growth
ref.sg.gha.tol	column name containing reference value of density plot for forest secondary growth
ref.sg.gini	column name containing reference value of density plot for forest secondary growth

**Value**

This function returns the iba value, with the different variables that employing for the calculate index

**Author(s)**

Christian Salas-Eljatib and Nicolas Pino

**Examples**

```
#df<-data.frame(d=rnorm(10,mean=80,sd=20))
#df$supe<-1200
#df$tol<-sample(x=c('SI','NO'),size = nrow(df),replace=TRUE)#input
```

---

interp.a	<i>Linear interpolation for three data of x and y only works if the first y is missing.</i>
----------	---

---

**Description**

Linear interpolation

**Usage**

```
interp.a(xs, ys)
```

**Arguments**

xs	Vector number of size 3
ys	Vector number of size 3, with first position empty or NA

**Details**

Linear interpolation for three data of x and y only works if the first y is missing.

**Value**

Output description (simple).

**Author(s)**

Christian Salas.

**Examples**

```
x<-c(1,2,3)
y<-c(NA,4,6)
interp.a(x,y)
```

---

interp.b	<i>Linear interpolation for three data of x and y only works if the second y is missing.</i>
----------	--

---

**Description**

Linear interpolation

**Usage**

```
interp.b(xs, ys)
```

**Arguments**

xs	Vector number of size 3
ys	Vector number of size 3, with second position empty or NA

**Details**

Linear interpolation for three data of x and y only works if the second y is missing.

**Value**

Output description (simple).

**Author(s)**

Christian Salas.

**Examples**

```
x<-c(1,2,3)
y<-c(4,NA,6)
interp.b(x,y)
```

---

invasivesRCI	<i>Contains regeneration microsite data in Robinson Crusoe Island forest</i>
--------------	--

---

**Description**

These are plot-level measurement data from the forests in the Robinson Crusoe Island, located in the Pacific Ocean, 667 km from mainland Chile. Measurements correspond to transects of 100 to 240 meters

**Usage**

```
data(invasivesRCI)
```

**Format**

Base de datos que contiene 14 columnas y 51 filas:

**plot.id** Plot identification code

**Gap.type** Canopy gap classified as invaded=Inv, non invaded= Nat or treated =Treat(considering the estimated cover of invasive plant species)

**Forest.zone** Location of the plot (gap, border or forest)

**Ferns** Estimated cover of fern species (in 2x2 plots)

**Moss.liverw** Estimated cover of mosses and liverworts ( in 2x2 plots)

**Cwd** Estimated cover of coarse woody debris > 3 cm diameter ( in 2x2 plots)

**Litter** Estimated cover of litter (in 2x2 plots)

**Ms** Estimated cover of mineral soil ( in 2x2 plots)

**Rock** Estimated cover of rocks (in 2x2 plots)

**Est.age** Age category for the canopy gap associated to each plot

**Source**

The data are provided courtesy of Prof. Rodrigo Vargas-Gaete at Universidad de La Frontera (Temuco, Chile).

**References**

Vargas R, Salas C, Gartner SM, Vidal OJ, Bannister JR, Pauchard A. (2018). Invasive plant species thresholds in the forests of Robinson Crusoe Island, Chile. *Plant Ecology & Diversity*. 11(2): 205-215.

---

inverse	<i>Mathematical expression for the Inverse function for relating variable Y versus X</i>
---------	--

---

**Description**

Inverse

**Usage**

```
inverse(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Inverse function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**Examples**

```
b0<-28.177
b1<-150.21
params<-c(b0,b1)
X <- c(70)
y<-inverse(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

kurt.coef	<i>Computes the estimator of Pearson's measure kurtosis</i>
-----------	---

---

**Description**

Computes the estimator of Pearson's measure kurtosis

**Usage**

```
kurt.coef(x, na.rm = T)
```

**Arguments**

x	vector of numbers
na.rm	if want to remove NA values

**Value**

the value of the estimator of Pearson's measure kurtosis

**Author(s)**

Christian Salas-Eljatib and Joaquin Riquelme-Alarcon

**Examples**

```
kurt.coef(rnorm(100))
```

---

l1euque	<i>Contains specie composition data L1euque forest</i>
---------	--

---

**Description**

—

**Usage**

```
l1euque
```

**Format**

Base de datos que contiene 72 columnas y 26 filas:

**stand** —

**plot.num** —

**Aus.chi** —

**May.dis** —

**Not.obl** —

**Pru.and** —

**Source**

The data are provided courtesy of Prof. Rodrigo Vargas-Gaete at Universidad de La Frontera (Temuco, Chile).

**References**

Vargas R, Salas C, Penneckamp D, Neira Z, Diez C, Vargas R. 2020. Estructura y regeneracion de bosques de Prumnopitys andina en los Andes del sur de Chile. Gayana botanica (to appear)

---

loaded.pack	<i>Computes basal area in larger trees</i>
-------------	--

---

**Description**

Gives a list of all the listed packages of an R session, and their respective versions

**Usage**

```
loaded.pack()
```

**Details**

No details are given

**Value**

Returns a list of all the listed packages

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
loaded.pack()
```

---

loessfit	<i>Draw a curve for the trend of points, from a LOESS fit, over a previously produced scatter-plot</i>
----------	--

---

**Description**

Estimates a simple locally weighted regression, as a way to depict a curve for

**Usage**

```
loessfit(x, y, col = NA, lty = NA, lwd = NA)
```



**Arguments**

x	predictor variable
y	response variable
col	is the color to be used for the curve, default is "blue"
lty	defains the option "lty" for plotting. Default is set to 2.
lwd	defains the option "lwd" for plotting. Default is set to 2.

**Details**

Be careful for interpreting the trend.

**Value**

This function draw a curve for the trend between two variables in a previously produced scatter-plot.

**Author(s)**

Timothy G. Gregoire and Christian Salas-Eljatib

**Examples**

```
#creating a fake dataframe
set.seed(1234)
x <- runif(100, 5, 60)
e <- runif(100, -20, 20)
y <- 1.5 + 2.4*x - .05*x^2 + e
df <- as.data.frame(cbind(Y=y,X=x))
df
#fitting a candidate model
plot(Y~X, data=df)
#using the function
loessfit(x=df$X, y=df$Y)
#qplot(mpg, wt, data = mtcars, geom = c("point", "smooth"))
```

---

logistic	<i>Mathematical expression for the Logistic function for relating variable Y versus X</i>
----------	---

---

**Description**

Logistic

**Usage**

```
logistic(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Logistic function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Pinheiro JC & DM Bates (2000) Mixed-effects Models in S and Splus. Springer-Verlag, New York, USA. 528p.

**Examples**

```
b0<- 29.28
b1<- 0.82
b2<- 0.07
params<-c(b0,b1,b2)
X <- c(70)
y<-logistic(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

logisticMod

---

*Mathematical expression for the Logistic function for relating variable Y versus X*


---

**Description**

LogisticoModificado

**Usage**

```
logisticMod(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Logistic function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Monserud RA (1984) Height growth and site index curves for Inland Douglas-fir based on stem analysis data and forest habitat type. Forest Science 30(4)943-965.

**Examples**

```
b0<- 29.28
b1<- 0.82
b2<- 0.07
params<-c(b0,b1,b2)
X <- c(70)
y<-logisticMod(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

lrt	<i>Computes a likelihood ratio test between a reduced model and a full model.</i>
-----	---

---

**Description**

Computes a likelihood ratio test between a reduced model (modr) and a full model (modf). Both models must be previously fitted by maximum likelihood.

Double-check the order of the reduced and full model, before of using the model

**Usage**

```
lrt(modr, modf)
```

**Arguments**

modr	is a previously fitted model having less parameters than modf
modf	is a previously fitted model having more parameters than modr

**Value**

This function returns an object having the following elements: "loglik.Modr" maximized log-likelihood of modr; "loglik.Modf" maximized log-likelihood of modf; "dif.loglik" difference in log-likelihood between both models, and "dif.df" difference in degrees of freedom of both models, and "p-value" is the p-value for the LRT.

**Author(s)**

Christian Salas-Eljatib.

**References**

Pinheiro JC, and Bates DM. 2000. Mixed-effects models in S and Splus. Springer-Verlag, New York, NY. 528 p.

**Examples**

```
#not yet implemented
```

---

meyer	<i>Mathematical expression for the Meyer function for relating variable Y versus X</i>
-------	--

---

**Description**

Meyer

**Usage**

```
meyer(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Meyer function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

Meyer HA (1940) A mathematical expression for height curves. Journal of Forestry 38(5) 415 420.

**Examples**

```
b0<-28.4
b1<-0.0584
params<-c(b0,b1)
X <- c(70)
y<-meyer(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

mmenten	<i>Mathematical expression for the Michaelis-Menten function for relating variable Y versus X</i>
---------	---

---

**Description**

Michaelis-Menten

**Usage**

```
mmenten(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Michaelis-Menten function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Pinheiro JC & DM Bates (2000) Mixed effects Models in S and Splus. Springer Verlag, New York, USA. 528p.

**Examples**

```
b0<-37.6843
b1<-18.6048
param<-c(b0,b1)
X <- c(70)
y<-mmenten(param,X,intercept=1.3)
plot(y~X,type='p')
```

---

moda	<i>Computes the mode</i>
------	--------------------------

---

**Description**

This function computes the mode of a vector

**Usage**

```
moda(v)
```

**Arguments**

`v` is a numeric vector

**Value**

This function returns the mode, a numeric scalar.

**Author(s)**

Christian Salas-Eljatib.

**Examples**

```
set.seed(1234)
y <- rnorm(10, 45)
#using the function
moda(y)
```

---

naslund	<i>Mathematical expression for the Naslund function for relating variable Y versus X</i>
---------	--

---

**Description**

Naslund

**Usage**

```
naslund(params, X, intercept = NA)
```

**Arguments**

<code>params</code>	Vector with value of parameters (The values must be added in an orderly way, see example)
<code>X</code>	Vector containing the values for the predictor variable
<code>intercept</code>	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Naslund function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Mehthatalo L, S de Miguel & TG Gregoire (2015) Modeling height diameter curves for prediction. Canadian Journal of Forest Research 45 826 837.

**Examples**

```
b0<- 1.08
b1<- 0.17
params<-c(b0,b1)
X <- c(70)
y<-naslund(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

nele.list

---

*Extract the n-th element from a list*


---

**Description**

Extract the n-th element from a list

**Usage**

```
nele.list(lst, n)
```

**Arguments**

lst	is the list object
n	is the position of the element in the list to be retrieved

**Value**

object with elements of each list

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
x <- list(list("z","x","y"), list(3,4,99,23,45), list(1,67,5,6,89))
nele.list(x,1)
nele.list(x,2)
nele.list(x,3)
```

---

nndist	<i>Computes nearest neighbor distances for objects in a spatial dataframe.</i>
--------	--

---

**Description**

Computes nearest neighbor distances for objects in a spatial dataframe. Be aware that this takes a while to run for large datasets.

**Usage**

```
nndist(pppData, PlotArea)
```

**Arguments**

pppData	a ppp data object (the data with the Cartesian coordinates)
PlotArea	a numeric scalar with the plot area.

**Value**

nearest neighbor distances for objects in a spatial dataframe

**Note**

Be aware that this takes a while to run for large datasets. Furthermore the pppData must be obtained from a previous step where the "spatstat" package is needed.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#not yet implemented
```



---

orange	<i>Diameter growth of orange trees</i>
--------	--

---

## Description

The orange data frame has 35 rows and four columns of records of the growth of orange trees.

## Usage

```
data(orange)
```

## Format

An object of class `c("nfnGroupedData", "nfGroupedData", "groupedData", "data.frame")` containing the following columns:

**tree.id** an ordered factor indicating the tree on which the measurement is made. The ordering is according to increasing maximum diameter.

**time** a numeric vector giving the numbers of days since 1968/12/31

**girth** a numeric vector of trunk perimeter (mm). This is probably a circumference at breast height, a standard measurement in forestry.

**dbh** a numeric vector of diameter at breast height (mm).

## Source

Draper NR and Smith H. (1998), Applied Regression Analysis (3rd ed), Wiley (exercise 24.N).

## Examples

```
#data(orange)

#coplot(dbh ~ time | tree.id, data = orange, show.given = FALSE)
#m1 <- nls(dbh ~ SSlogis(age, Asym, xmid, scal),
#         data = orange, subset = tree.id == 3)
#plot(dbh ~ time, data = orange, subset = tree.id == 3,
#     xlab = "Time (number of days since 1968/12/31)",
#     ylab = "Tree diameter (mm)", las = 1,
#     main = "Diameter growth data of orange trees and fitted model (tree.id 3 only)")
#time <- seq(0, 1600, length.out = 101)
#lines(time, predict(m1, list(time = time)))
```

---

pinaster	<i>Contains tree-level variables for Pinus pinaster in the Baixo-Mino, Galicia, Spain.</i>
----------	--

---

## Description

These are tree-level measurement data of sample trees in the Baixo-Mino region in Galicia, Spain.

## Usage

```
data(pinaster)
```

## Format

Contains tree-level variables, as follows:

**stand** stand number from the sample tree was selected.

**si** Site index of the stand.

**tree.no** tree number.

**dbh** Diameter at breast height, in cm.

**htot** Total height, in m.

**d4** Upper-stem diameter at 4 m, in cm.

**vol.wb** Tree gross volume, in m<sup>3</sup> with bark.

**vol.wob** Tree gross volume, in m<sup>3</sup> without bark.

## Source

The data are provided courtesy of Dr. Christian Salas at the Universidad Mayor (Santiago, Chile).

## References

- Salas C, Nieto L, Irisarri A. 2005. Modelos de volumen para Pinus pinaster Ait. en la comarca del Baixo Mino, Galicia, Espana. Quebracho 12: 11-22.

## Examples

```
data(pinaster)
head(pinaster)
```

---

pinusContorta	<i>Contains information of invasive of pinus contorta.</i>
---------------	--

---

## Description

These are tree-level measurement data, with x,y location of each tree, from Pinus contorta invasion in Patagonian steppe in Coyhaique in southern Chile, measured in 2011. The plots area was 10000 square meters.

## Usage

```
data(pinusContorta)
```

## Format

Contains 8 variables, as follows:

**plot.id** Plot sample identificativo number.

**tree.id** Tree identifier number in each plot. Same indiv/id for multi-stem trees.

**y.coord** coordinate of S latitude.

**x.coord** coordinate of W longitude.

**substrate** Ground cover in which each pine grow. Bare soil, Festuca pallescens, Baccharis magellanica, Oreopulus glacialis, Acaena integerrima and others species.

**drc** Diameter at the root collar on trees, in mm.

**h** Height of trees, in cm.

**canopy.area** Projection of canopy area of each tree, in square meters.

## Source

The data are provided courtesy of Drs. Anibal Pauchard and Rafael Garcia at the Laboratorio de Invasiones Biologicas, Universidad de Concepcion (Chile).

## References

Pauchard A, A Escudero, RA Garcia, M de la Cruz, B Langdon, LA Cavieres & J Esquivel. 2016. Pine invasions in treeless environments: dispersal overruns microsite heterogeneity. Ecology and Evolution. 6(2): 447 - 459

## Examples

```
data(pinusContorta)
head(pinusContorta)
```

---

pinusSpp

*Contains information of invasive of pinus spp.*


---

### Description

These are tree-level measurement data from Pinus spp invasion in Araucaria-Nothofagus forests in the Malalcahuello National Reserve in La Araucania region in southern Chile, measured in 2012. The plots area was 100 square meters

### Usage

```
data(pinusSpp)
```

### Format

Contains 8 variables, as follows:

**plot.id** Plot sample indentificator number.

**size.plot** Plot size in square meters.

**Lat.s** Decimal coordinate of S latitude.

**Long.w** Decimal coordinate of W longitude.

**indv.id** Tree identificador number in each plot. Same indv/id for multi-stem trees.

**stem.id** Stem identificador number in each plot.

**sp** Specie.

**dbh** Diameter at breast height on trees, in cm.

**h** Height of trees, in m.

**canopy.h** Height at which the live canopy begins, in m.

**canopy.lenght** Lenght of live canopy, in m.

**obs** Extra information.

### Source

The data are provided courtesy of Drs. Anibal Pauchard and Rafael Garcia at the Laboratorio de Invasiones Biologicas, Universidad de Concepcion (Chile).

### References

Cobar-Carranza A, Garcia R, Pauchard A & Pena E. 2014. Effect of Pinus contorta invasion on forest fuel properties and its potential implications on the fire regime of Araucaria araucana and Nothofagus antarctica forests. Biological Invasions. 16(11): 2273 - 2291

### Examples

```
data(pinusSpp)
head(pinusSpp)
```

---

plantsHawaiian	<i>Contains information of plants Hawaiians.</i>
----------------	--

---

## Description

Dataset contains 58 observations,

## Usage

```
data(plantsHawaiian)
```

## Format

Contains 6 variables, as follows:

**scientific.name** Genus and epithet of each individual following The Plant List v. 1.1 (2013).

**family** Family of each individual following The Plant List v. 1.1 (2013).

**native.status** Categorical variable ('native', 'alien', 'uncertain') indicating alien status of each individual following Wagner et al. (2005).

**n** Number of individuals used to estimate maximum plant size.

**d.95** Maximum plant size, estimated as D950.1 (King et al. 2006).

**d.max.3** Maximum plant size, estimated as Dmax3 (King et al. 2006).

## Source

The data were provided from DRYAD repository.

## References

- Craven D, Knight T, Barton K, Bialic-Murphy L, Cordell S, Giardina C, Gillespie T, Ostertag R, Sack L, Chase J. 2018. OpenNahele: the open Hawaiian forest plot database. Biodiversity Data Journal 6: e28406. <https://doi.org/10.3897/BDJ.6.e28406>

## Examples

```
data(plantsHawaiian)
head(plantsHawaiian)
```

---

plotLleuque	<i>Contains plot level data Lleuque forest</i>
-------------	--

---

**Description**

—

**Usage**

plotLleuque

**Format**

Base de datos que contiene 15 columnas y 26 filas:

- stand** —
- plot.num** —
- elevation** —
- aspect** —
- slope** —
- stump** —
- cattle.faeces** —
- dist.to.river** —
- fruits.ha** —
- browse** —

**Source**

The data are provided courtesy of Prof. Rodrigo Vargas-Gaete at Universidad de La Frontera (Temuco, Chile).

**References**

Vargas R, Salas C, Penneckamp D, Neira Z, Diez C, Vargas R. Estructura y regeneracion de bosques de Prumnopitys andina en los Andes del sur de Chile (in Press). Gayana botanica

---

poder	<i>Mathematical expression for the Poder function for relating variable Y versus X</i>
-------	--

---

**Description**

poder

**Usage**

poder(params, X, intercept = NA)

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Poder function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

Stage AR (1975) Prediction of height increment for models of forest growth. USDA For. Serv. Res. Pap. INT164, USA. 20 p.

**Examples**

```
b0<-5.22
b1<-0.43
params<-c(b0,b1)
X <- c(70)
y<-poder(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

poissonWeibull	<i>Mathematical expression for the Poisson-Weibull function for relating variable Y versus X</i>
----------------	--

---

**Description**

Poisson-Weibull

**Usage**

```
poissonWeibull(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Poisson-Weibull function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

Piedade V, G Oliveira & F Louzada (2017) The Poisson-Weibull regression model. The Chilean Journal of Statistics 8(1) 25-51.

**Examples**

```
b0<- 126.174
b1<- 0.013
b2<- 3.49
params<-c(b0,b1,b2)
X <- c(70)
y<-poissonWeibull(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

presenceIce

*Presence or absence of sea ice from logbook records of annual cruises*

---

**Description**

Data containing 52717 observations , about presence of sea ice from logbook records of annual cruises to the B-C-B in an unbroken record between years 1850 to 1910.

**Usage**

```
data(presenceIce)
```

**Format**

The data frame contains nine variables as follows:

**ship.id** The code number for ships.

**move.type** Type of movement of ships. 0 indicates a sail-powered vessel and 1 indicates an auxiliary-powered vessel.

**year** Year of registry.

**month** Month of registry.

**day** Day of registry.

**lat.dec** Decimal latitude.

**long.dec** Decimal longitude.

**e.w** East or west of the Prime Meridian.

**ice.cov** Sea Ice Observed. 0 no see (Not registered) and 1 presence sea ice (Registered).



## Source

The data were provided from Sea Ice Group at the Geophysical Institute.

## References

Mahoney A, Bockstoce J, Botkin D, Eicken H, Nisbet R. 2011. Sea-Ice Distribution in the Bering and Chukchi Seas: Information from Historical Whaleships' Logbooks and Journals ARCTIC. 64(4): 465-477.

## Examples

```
data(presenceIce)
head(presenceIce)
```

---

prodan	<i>Mathematical expression for the Prodan function for relating variable Y versus X</i>
--------	---

---

## Description

Prodan

## Usage

```
prodan(params, X, intercept = NA)
```

## Arguments

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

## Details

Mathematical expression for the Prodan function for relating variable Y versus X

## Value

The value of the function for every value of the vector X and the parameters.

## Author(s)

Christian Salas.

## References

- van Laar A & A Akca (1997) Forest Mensuration. Cuvillier Verlag, Gottingen, Germany. 418 p.

### Examples

```
b0<- -0.5467
b1<-0.5477
b2<-0.0254
params<-c(b0,b1,b2)
X<- c(70)
y<-prodan(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

pspLlancahue	<i>Tree locations for a sample plot in the Llancahue experimental forest, near Valdivia, Chile.</i>
--------------	---

---

### Description

The Cartesian position, species, and diameter of trees within a plot were measured. The sample plot is rectangular of 130 m by 70 m. Further details can be #’ reviewed in the reference.

### Usage

```
data(pspLlancahue)
```

### Format

Contains tree-level variables, as follows:

**tree.code** Tree identifier

**spp.name** species abbreviation as follows: AP= Aextocicon punctatum, EC=Eucryphia cordifolia, GA=Gevuina avellana, LP= Laureliopsis philippiana, LS= Laurelia sempervirens, ND=Nothofagus dombeyi, Ot=Other, PS=Podocarpus saligna

**dbh** diameter at breast height, in cm

**x.coord** Cartesian position in the X-axis, in m

**y.coord** Cartesian position in the Y-axis, in m

### Source

The data are provided courtesy of Prof. Daniel Soto at Universidad de Aysen (Coyhaique, Chile).

### References

- Soto DP, Salas C, Donoso PJ, Uteau D. 2010. Heterogeneidad estructural y espacial de un bosque mixto dominado por Nothofagus dombeyi despues de un disturbio parcial. Revista Chilena de Historia Natural 83(3): 335-347.

### Examples

```
data(pspLlancahue)
head(pspLlancahue)
```

ptaeda

*Height growth of Pinus taeda (Loblolly pine) trees***Description**

The Loblolly data frame has 84 rows and tree columns of records of the tree height growth of Loblolly pine trees. This dataframe is a slight modification to the original dataframe "Loblolly" from the datasets R package.

**Usage**

```
data(ptaeda)
```

**Format**

An object of class c("nfnGroupedData", "nfGroupedData", "groupedData", "data.frame") containing the following columns:

**seed.id** an ordered factor indicating the seed source for the tree. The ordering is according to increasing maximum height.

**age** a numeric vector of tree ages, in yr.

**height** a numeric vector of tree heights, in m.

**Source**

Pinheiro, J. C. and Bates, D. M. (2000) Mixed-effects Models in S and S-PLUS. Springer.

**Examples**

```
#data(ptaeda)
#plot(height ~ age, data = ptaeda, subset = seed.id == 329,
#      xlab = "Tree age (yr)", las = 1,
#      ylab = "Tree height (m)",
#      main = "Loblolly data and fitted curve (seed.id 329 only)")
#fm1 <- nls(height ~ SSasym(age, Asym, R0, lrc),
#           data = ptaeda, subset = seed.id == 329)
#age <- seq(0, 30, length.out = 101)
#lines(age, predict(fm1, list(age = age)))
```

pvalue.t

*Computes the p-value for a Standar t-distributed random variable***Description**

Computes the p-value for a Standar t-distributed random variable.

**Usage**

```
pvalue.t(t.value, df)
```

**Arguments**

t.value            a random variable following a t-student pdf distribution  
 df                degrees of freedom of the random variable following a t-student pdf distribution.

**Value**

This function returns the p-value or probability of getting a value as large as t.value.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
# Load dataset

#aa <- biometria::araucaria
#
## Calculate t value
#t.value <- t.test(aa$hdom)
#t.v <- as.numeric(t.value$statistic)
#deg.f <- as.numeric(t.value$parameter)

## Calculate p value
#pvalue.t(t.v,deg.f)
```

---

pvalue.z	<i>Computes the p-value for a Standar t-distributed random variable</i>
----------	---

---

**Description**

Computes the p-value for a Standar t-distributed random variable.

**Usage**

```
pvalue.z(z.value)
```

**Arguments**

z.value            a random variable following a normal distribution

**Value**

This function returns the p-value or probability of getting a value as large as x.

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
pvalue.z(1.96)
```

---

qmd	<i>Computes the quadratic mean diameter of a sample plot.</i>
-----	---

---

### Description

This function computes the quadratic mean diameter of a sample plot.

### Usage

```
qmd(tph = tph, gha = gha)
```

### Arguments

tph	is tree density, in trees/ha;
gha	is basal area, in m <sup>2</sup> /ha

### Value

Returns the quadratic mean diameter (in cm) for a given plot.

### Author(s)

Christian Salas-Eljatib.

### Examples

```
#using the function
qmd(tph=1023, gha=50)
```

---

radiatapl	<i>Data from a Pinus radiata plantation near Capitan Pastene, Region de La Araucania, Chile.</i>
-----------	--

---

### Description

Tree-level information collected within sample plots in a forestry plantation of Pinus radiata near Capitan Pastene, Southern Chile. Sample plots size is 150 square meters.

### Usage

```
data(radiatapl)
```

### Format

The data frame contains four variables as follows:

**plot** Plot number identification.

**tree** Tree number identification.

**dbh** Diameter at breast height in cm.

**heigth** Total height in m.

**Source**

The data are provided courtesy of Dr. Christian Salas at the Universidad Mayor (Santiago, Chile).

**Examples**

```
data(radiatapl)
head(radiatapl)
```

---

`radiosCopa`

*Data with information radios crown for different directions on site ru-camanque*

---

**Description**

Dataset contains information for species Roble, Coigue y Olivillo about radios of crown for different directions, coordinates and diameter crown.

**Usage**

```
data(radiosCopa)
```

**Format**

Contains of variables, as follows:

**specie** Code of specie. ro is Roble, co is Coigue and ol is Olivillo.

**dap** Diameter at breast height.

**htot** Total height in meters.

**north** Radio of crown in direction north in meters.

**east** Radio of crown in direction east in meters.

**south** Radio of crown in direction south in meters.

**west** Radio of crown in direction west in meters.

**x** Coordinate x.

**y** Coordinate y.

**crown** Diameter of crown in meters.

**Source**

not yet

**References**

not yer

**Examples**

```
data(radiosCopa)
head(radiosCopa)
```

---

ranking2	<i>Computes the skewness of given vector</i>
----------	--

---

**Description**

asigna posicion de de menor a mayor a una columna especifica dentro de la db, si tienen el mismo valor asigna el mismo puesto en el ranking

**Usage**

```
ranking2(data, vector, ranking)
```

**Arguments**

data	is a db that contains at least a numeric vector to apply the ranking
vector	is the name of the numeric column to apply the ranking
ranking	is the name of the column where will save the ranking

**Value**

the value of the the skewness of given vector

**Author(s)**

Joaquin Riquelme-Alarcon

**Examples**

```
vector <- sample(2:20, 10, replace = TRUE)
asd<-letters[1:10]
data<-data.frame(vector=vector,asd= asd, stringsAsFactors = FALSE)
ranking2(data=data, vector = "vector", ranking = "posi")
```

---

ratkowsky	<i>Mathematical expression for the Ratkowsky function for relating variable Y versus X</i>
-----------	--

---

**Description**

Ratkowsky

**Usage**

```
ratkowsky(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters at model (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Ratkowsky function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Zhang L (1997) Cross validation of non linear growth functions for modelling tree height diameter relationships. *Annals of Botany* 79(3) 251 257.

**Examples**

```
b0<- 39.5
b1<- 21.12
b2<- 9.68
params<-c(b0,b1,b2)
X <- c(70)
y<-ratkowsky(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

rdRoraco

*Computes relative density of forest plots from the study of Salas-Eljatib & Weiskittel (2018)*

---

**Description**

Computes relative density of forest plots from the study of Salas-Eljatib & Weiskittel (2018).

**Usage**

```
rdRoraco(data = data, qmd = qmd, nha = nha)
```

**Arguments**

data	a dataframe containing variables as columns
qmd	indicates the name of the column having the quadratic mean diameter (in cm) of the sample plot
nha	indicates the name of the column having the tree density (in trees/ha) of the sample plot



**Details**

You must be careful in using it outside the range for which the model was built.

**Value**

This function returns a vector having the relative density value.

**Author(s)**

Christian Salas-Eljatib.

**References**

Salas-Eljatib C, Weiskittel AR. 2018. Evaluation of modeling strategies for assessing self-thinning behavior and carrying capacity. *Ecology and Evolution* 8 (22): 10768-10779.

**Examples**

```
#creating a fake dataframe
set.seed(1234)
plot.no <- 1:10 # plot ID column
dg <- round(rnorm(10,30,5),1)
tph <- round(rnorm(10,600,100),1)
df<-data.frame(plot.no,dg,tph)
#using the function
rdRoraco(data=df, qmd="dg", nha = "tph")
```

---

regeneraNothofagus	<i>Contains information about regeneration of nothofagus.</i>
--------------------	---

---

**Description**

Dataset contains 442 observations.

**Usage**

```
data(regenerationNothofagus)
```

**Source**

The data were provided from DRYAD repository

**References**

- Soto D, Puettmann K.2018.Topsoil removal through scarification improves natural regeneration in high-graded Nothofagus old-growth forests.*Journal Applied Ecology*. 55: 967- 976. <https://doi.org/10.1111/1365-2664.12989>

**Examples**

```
data(regeneraNothofagus)
head(regeneraNothofagus)
```

---

richards	<i>Mathematical expression for the Richards function for relating variable Y versus X</i>
----------	---

---

**Description**

Richards

**Usage**

```
richards(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Richards function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Richards FJ (1959) A flexible growth function for empirical use. Journal of Experimental Botany 10(29), 290-300.

**Examples**

```
b0<- 34.19
b1<- 0.03
b2<- 2.91
params<-c(b0,b1,b2)
X <- c(70)
y<-richards(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

schnute	<i>Mathematical expression for the Schnute function for relating variable Y versus X</i>
---------	--

---

**Description**

Schnute

**Usage**

```
schnute(params, X, intercept)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Schnute function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Schnute I (1981) A versatile growth model with statistically stable parameters. Can. J. Fish. Aquat. Sci.38(9), 1128-1140.

**Examples**

```
b0<-37
b1<-0.05
b2<-0.03
params<-c(b0,b1,b2)
X <- c(70)
y<-schnute(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

schumacher	<i>Mathematical expression for the Schumacher function for relating variable Y versus X</i>
------------	---

---

**Description**

Schumacher

**Usage**

```
schumacher(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Schumacher function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Schumacher FX (1939) A new growth curve and its application to timber yield studies. Journal of Forestry 37(10) 819 820.

Burkhardt HE & MR Strub (1974) A model for simulation of planted loblolly pine stands. In J. Fries (Ed.) Growth models for tree and stand simulation, Res.Note 30. Proc. IUFRO Working Party S4.01-4 Meetings, Dep. For. Yield Res., Royal Coll. For., Stockholm, Sweeden, p. 128-135.

**Examples**

```
b0<- 31.92
b1<- 9.34
params<-c(b0,b1)
X <- c(70)
y<-schumacher(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

sibbesen	<i>Mathematical expression for the Sibbesen function for relating variable Y versus X</i>
----------	---

---

**Description**

Sibbesen

**Usage**

```
sibbesen(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Sibbesen function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Mehthatalo L, S de Miguel & TG Gregoire (2015) Modeling height diameter curves for prediction. Canadian Journal of Forest Research 45 826 837

**Examples**

```
b0<-43
b1<-4.27
b2<-0.92
params<-c(b0,b1,b2)
X <- c(70)
y<-sibbesen(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

simula

---

*Simulated yield of forestry plantations of exotic species in Chile.*


---

## Description

The yield tables of simulated plantations of *Pinus radiata*, *Eucalyptus globulus*, and *Eucalyptus nitens* are obtained from the Radiata simulator and EucaSim simulator built in Chile. Several stand-level variables are part of the output.

## Usage

```
data(simula)
```

## Format

Contains stand-level variables, as follows:

**species** "P. radiata" is *Pinus radiata*, "E. globulus" is *Eucalyptus globulus*, and "E. nitens" is *Eucalyptus nitens*.

**age** plantation age, in years

**tph** Tree density, in trees/ha

**gha** Basal area, in m<sup>2</sup>/ha

**toph** Dominant height, in m

**qmd** quadratic mean diameter, in cm

**totvol** gross stand volume, in m<sup>3</sup>/ha

**viu.10** stand volume below an utilization index of 10 cm, in m<sup>3</sup>/ha

**viu.15** stand volume below an utilization index of 15 cm, in m<sup>3</sup>/ha

**viu.20** stand volume below an utilization index of 20 cm, in m<sup>3</sup>/ha

**viu.25** stand volume below an utilization index of 25 cm, in m<sup>3</sup>/ha

## Source

The data were obtained as outputs for plantations without management in Chile. The academic version of the simulator was used. You can visit [mnssimulacion.cl](http://mnssimulacion.cl)

## Examples

```
data(simula)
```

---

sizeDist.byGroupAllPlot

*Size distribution plot by levels of a group and all in a single figure.*


---

## Description

The main feature of this function is to produce a single figure having: a size distribution plot by a grouping factor and a general (i.e., no considering the grouping factor) size distribution plot as an insert. This sort of plot is specially useful in forest ecology, and mimics Figure 1 in Salas et (2006). Size here is a random variable (in ecology this could be the biomass of a tree, or the weight of a bear), as well as graphic representations of those tables that we refer to them as size distribution plots. For instance, in forest ecology the size distribution tables are usually referred as "stand tables", and the size distribution plots as "diameter distribution" when the random variable tree diameter is used.

## Usage

```
sizeDist.byGroupAllPlot(
  data = data,
  var.int = var.int,
  plot.area = plot.area,
  w.amp = 10,
  group.var = NULL,
  max.class.var.int = NA,
  max.cy,
  max.y.by.group = NA,
  max.y.all = NA,
  col.bar.all = "white",
  posi.leg.group = "right",
  posi.all.groups.plot = NA,
  eng = TRUE,
  filenamestablegroup = "standTableGroup",
  filenamestable = "standTable",
  levels.i.want = FALSE,
  col.lev.i.want = FALSE,
  xlab.mainplot = NA,
  ylab.mainplot = NA,
  time.var = NA,
  at.N.all = NA,
  metric = TRUE,
  unit.density = NA,
  unit.var.int = NA
)
```

## Arguments

data	a dataframe with at least the columns representing a variable of interest ("var.int"), the are of a given sample plot ("plot.area"), and a grouping variable ("group.var").
var.int	is the name of the column having the size variable or variable of interest.
plot.area	is the plot area, in m <sup>2</sup> , of the sample plot.

w.amp	is a number for the size-variable class amplitud to be used in the plots. The default value is set to 10. It is expressed in the same units that "var.int".
group.var	a string with the name of the column having the grouping variable (e.g., species or treatment).
max.class.var.int	the value for the maximum class to be used for the "var.int", and It must be a multiple number of "w.amp".
max.cy	an optional scalar containing the maximum class for the variable of interest "var.int"
max.y.by.group	an optional number specifying the maximum value to be used for the density of the main histogram of the figure (the one by the grouping variable "group.var").
max.y.all	an optional number specifying the maximum value to be used for the density of the histogram when a grouping variable is not defined.
col.bar.all	the colour for the bars (when grouping variable is not defined) drawn in the insert total level sizeDist plot.
posi.leg.group	an optional string specifying the position of the legend. The default is "topright"
posi.all.groups.plot	an optional string specifying the position of the small plot not segregating for levels of the factor variable. The default is "topright"
eng	a logical value for the language of the axis. By default is set to TRUE, otherwise spanish is used.
filenametablegroup	a string specifying the name of the file where the stand table by group will be stored. The default is "standTableGroup". Two files will be saved, with both .out and .csv extensions.
filenametable	a string specifying the name of the file where the stand table will be stored. The default is "standTable". Two files will be saved, with both .out and .csv extensions.
levels.i.want	a string providing the name of the levels of the factor variable "group.var" to be used for the plotting.
col.lev.i.want	is a vector of length equal to the number of levels of the "group.var", specifying the colours to be used for each level.
xlab.mainplot	a string with the label used for the X-axis of the main plot.
ylab.mainplot	a string with the label used for the Y-axis of the main plot.
time.var	a string with the name of the column having the time variable (e.g., year of measurement).
at.N.all	Explain parameter
metric	a logical value if the Metric system is used. The default value is TRUE, otherwise the Imperial units are used.
unit.density	an optional string providing the name to be used for the Y-axis when plotting.
unit.var.int	an optional string providing the name to be used for the X-axis when plotting.

## Value

This function returns a plot, as well as two outputs (file with extension ".out") with the size distribution tables by the levels of the grouping variable, and the other without considering any segregation.



**Note**

Be aware that this function has several options that can be specified by an user. Be sure that all values of the variable of interest are available in the dataframe before of using the function.

**Author(s)**

Christian Salas-Eljatib

**References**

- Husch B, TW Beers, JA Kershaw. 2003. Forest Mensuration. 4th edition. Wiley, New York, USA. 443 p.
- Salas C, LeMay V, Nunez P, Pacheco P, Espinosa A. 2006. Spatial patterns in an old-growth Nothofagus obliqua forest in south-central Chile. Forest Ecology and Management 231:38-46

**Examples**

```
data(pspLlancahue)
head(pspLlancahue)
pspLlancahue$area <- 70*130
sizeDist.byGroupAllPlot(pspLlancahue, var.int="dbh", plot.area = "area", group.var = "spp.name")
sizeDist.byGroupAllPlot(pspLlancahue, var.int="dbh", plot.area = "area", group.var = "spp.name",
  max.class.var.int = 100)
sizeDist.byGroupAllPlot(pspLlancahue, var.int="dbh", plot.area = "area", group.var = "spp.name",
  max.y.by.group = 500, max.y.all = 500, posi.leg.group = "topleft")
```

---

sizeDistPlot

---

*Creates size distribution tables and size distribution plots*


---

**Description**

This function creates size distribution tables, where size is a random variable (in ecology this could be the biomass of a tree, or the weight of a bear), as well as graphic representations of those tables that we refer to them as size distribution plots. For instance, in forest ecology the size distribution tables are usually referred as "stand tables", and the size distribution plots as "diameter distribution" when the random variable tree diameter is used.

**Usage**

```
sizeDistPlot(
  data = data,
  var.int = var.int,
  plot.area = plot.area,
  w.amp = 10,
  at.N.all,
  max.class.var.int = NA,
  max.y.all = NA,
  filename.tablegroup = "standTableGroup",
  filename.table = "standTable",
  max.y.by.group = NA,
```

```

posi.leg.group = "topright",
group.var = NULL,
eng = TRUE,
ylab = NA,
xlab = NA,
col.bar.all = "gray",
levels.i.want = FALSE,
col.lev.i.want = FALSE,
metric = TRUE,
unit.density = NA,
unit.var.int = NA
)

```

### Arguments

<code>data</code>	a dataframe with at least the columns representing a variable of interest (" <code>var.int</code> "), the are of a given sample plot (" <code>plot.area</code> "), and a grouping variable (" <code>group.var</code> ").
<code>var.int</code>	is the name of the column having the size variable or variable of interest.
<code>plot.area</code>	is the plot area, in m2, of the sample plot.
<code>w.amp</code>	a number for the size-variable class amplitud to be used in the plots. By default is equal to 10. It is expressed in the same units that " <code>var.int</code> ".
<code>at.N.all</code>	an optional vector containing the values of density to be drawn in the insert total level sizeDist plot
<code>max.class.var.int</code>	an optional numeric scalar containing the maximum class for the variable of interest " <code>var.int</code> "
<code>max.y.all</code>	an optional numeric scalar containing the maximum level of the Y-axis
<code>filenametablegroup</code>	a string specifying the name of the file where the stand table by group will be stored. The default is "standTableGroup". Two files will be saved, with both <code>.out</code> and <code>.csv</code> extensions.
<code>filanetable</code>	a string specifying the name of the file where the stand table will be stored. The default is "standTable". Two files will be saved, with both <code>.out</code> and <code>.csv</code> extensions.
<code>max.y.by.group</code>	an optional number specifying the maximum value to be used for the density of the main histogram of the figure (the one by the grouping <code>#' #'</code> variable " <code>group.var</code> ").
<code>posi.leg.group</code>	an optional string specifying the position of the legend. The default is "topright"
<code>group.var</code>	a string with the name of the column having the grouping variable (e.g., species or treatment).
<code>eng</code>	a logical value for the language of the axis. By default is set to TRUE, otherwise spanish is used.
<code>ylab</code>	a string with the label used for the Y-axis.
<code>xlab</code>	a string with the label used for the X-axis.
<code>col.bar.all</code>	is the colour for the bars (when grouping variable is not defined) drawn in the insert total level sizeDist plot.
<code>levels.i.want</code>	a string providing the name of the levels of the factor variable " <code>group.var</code> " to be used for the plotting.

<code>col.lev.i.want</code>	is a vector of length equal to the number of levels of the "group.var", specifying the colours to be used for each level.
<code>metric</code>	a logical value if the Metric system is used. The default value is TRUE, otherwise the Imperial units are used.
<code>unit.density</code>	an optional string providing the name to be used for the Y-axis when plotting.
<code>unit.var.int</code>	an optional string providing the name to be used for the X-axis when plotting.

### Value

This function returns a plot, as well as two outputs (file with extension ".out") with the size distribution tables by the levels of the grouping variable, and the other without considering any segregation.

### Note

Be aware that this function has several options that can be specified by an user. Be sure that all values of the variable of interest are available in the dataframe before of using the function.

### Author(s)

Christian Salas-Eljatib

### References

- Husch B, TW Beers, JA Kershaw. 2003. Forest Mensuration. 4th edition. Wiley, New York, USA. 443 p.
- Salas C, LeMay V, Nunez P, Pacheco P, Espinosa A. 2006. Spatial patterns in an old-growth *Nothofagus obliqua* forest in south-central Chile. Forest Ecology and Management 231:38-46

### Examples

```
data(pspLlancahue);
head(pspLlancahue)
pspLlancahue$area <- 70*130
sizeDistPlot(pspLlancahue, var.int="dbh", plot.area = "area")
sizeDistPlot(pspLlancahue, var.int="dbh", plot.area = "area", w.amp=5)
sizeDistPlot(pspLlancahue, var.int="dbh", plot.area = "area", group.var = "spp.name")
sizeDistPlot(pspLlancahue, var.int="dbh", plot.area = "area", group.var = "spp.name",
  col.lev.i.want =c("black","white","blue","gray","darkgreen","yellow"),
  max.class.var.int = 120, max.y.by.group = 250)
```

---

skew.coef

*Computes the skewness of given vector*

---

### Description

Computes the skewness of given vector

### Usage

```
skew.coef(x, na.rm = T)
```

**Arguments**

**x** vector of numbers  
**na.rm** if want to remove NA values

**Value**

the value of the the skewness of given vector

**Author(s)**

Christian Salas-Eljatib and Joaquin Riquelme-Alarcon

**Examples**

```
#not yet implemented
```

---

sludge	<i>Contains information of sludge in a different cities, with a value of concentration zinc.</i>
--------	--

---

**Description**

Dataset contains 36 observations

**Usage**

```
data(sludge)
```

**Format**

Contains 4 variables, as follows:

**city** Name of city.  
**rate** Concentration rate of sludge.  
**zinc** Value of concentration ( in ppm).  
**trt.comb** Combination between city and rate factors.

**Source**

The data were provided from.

**References**

not yet

**Examples**

```
data(sludge)  
head(sludge)
```

---

snaspeChile*Data with information on the National System of State Protected Wild Areas (SNASPE)*

---

## Description

Dataset contains the protected wild areas of Chile that are part of the National System of State Protected Wild Areas (SNASPE).

## Usage

```
data(snaspeChile)
```

## Format

Contains of variables, as follows:

**g.id** Id.

**unit** Name of the protected area.

**category** Category of the unit. It can be either a National Park, a National Reserve or a Natural Monument.

**commune** Name of the commune (the smallest Chilean territorial division) where the unit is located.

**province** Province where the comunne is located (one territorial division level above the commune).

**region** Region where the province is located (one territorial division level above the province and the biggest Chilean territorial division).

**perim.km** Perimeter of the unit in kilometers.

**area.ha** Area of the unit in hectares.

**area.m2** Area of the unit in square meters.

## Source

These data is freely available at <http://ide.minagri.gob.cl/geoweb/2019/11/21/medio-ambiente/>

## References

The SNASPE has been created and is currently managed by the National Forest Corporation (CONAF). More information and documentation can be found at <https://www.conaf.cl/parques-nacionales/parques-de-chile/>

## Examples

```
data(snaspeChile)
head(snaspeChile)
```

---

spatAustria

*Tree locations for different plots of the spruce Norway*


---

## Description

The Cartesian position, species, year, ID tree, and diameter of trees within a plot were measured.

## Usage

```
data(spatAustria)
```

## Format

Contains tree-level variables, as follows:

**plot.code** Plot identifier

**tree.code** Tree identifier

**spp.name** species abbreviation as follows: PCAB=Picea abies, FASY=Fagus sylvatica, QCPE=Quercus petraea, PNSY=Pinus Sylvestris, LADC=Larix decidua

**x.coord** Cartesian position in the X-axis, in m

**y.coord** Cartesian position in the Y-axis, in m

**year** Measurement year

**dbh** diameter at breast height, in cm

## References

- Kindermann G, Kristofel F, Neumann M, Rossler G, Ledermann T & Schueler. 2018. 109 years of forest growth measurements from individual Norway spruce trees. Sci. Data 5:180077 DOI: 10.1038/sdata.2018.77.

## Examples

```
# data(spatAustria)
#head(spatAustria)
#graphics for tree by plots
#pos<-data(spatAustria)
#par(mar=c(4,4,0,0))
#bord<-data.frame(x=c(min(pos$x.coord),max(pos$x.coord),min(pos$x.coord),max(pos$x.coord)),
#                  y=c(min(pos$y.coord),min(pos$y.coord),max(pos$y.coord),min(pos$y.coord)))
#plot(bord,type="n", xlab="x [m]", ylab="y [m]", asp=1, bty='n')
#points(pos$x.coord,pos$y.coord,col=pos$plot.code,cex=0.5)
```

---

spatdistplot	<i>Produces a spatial distribution plot. If provided, a marked spatial point plot is drawn instead of point-patterns only.</i>
--------------	--

---

## Description

The main feature of this function is to produce a single figure having: a spatial distribution plot of points (point pattern) or a marked point pattern.

## Usage

```
spatdistplot(
  data = data,
  xc = xc,
  yc = yc,
  mark = NULL,
  group.var = NULL,
  main.tex = NULL,
  ylab,
  tresh.low.mark = NULL,
  tresh.greo.mark = NULL,
  col.tresh = FALSE,
  legend = TRUE,
  biom.plotting = FALSE,
  corre.fac = 0.05,
  cex = 1,
  plot.as.trees = FALSE,
  dimx = dimx,
  dimy = dimy,
  col.mark = "gray",
  col.border = "black"
)
```

## Arguments

data	a dataframe with at least two columns representing the X-axis coordinate ("xc"), the Y-axis coordinate ("yc"). Optionally, any mark can be used.
xc	the position at the X-axis coordinate.
yc	the position at the Y-axis coordinate.
mark	a character giving the column name of the response variable or variable of interest.
group.var	an optional character having the name of a column having a factor variable (e.g., species). The default value is set to NULL. Notice that a group.var can also me a mark for spatial analysis.
main.tex	The main title of the plot
ylab	Label for the Y-axis
tresh.low.mark	a number indicating a treshold value for the mark, using the logical operator "lower than". The default value is set to NULL.

tresh.greo.mark	a number indicating a treshhold value for the mark, using the logical operator "greater or equal than". The default value is set to NULL.
col.tresh	is a vector of length two, having the colors to be used for the groups being identified by the rule provided in either tresh.low.mark or tresh.greo.mark. It only should be defined if the options tresh.low.mark or tresh.greo.mark are not NULL.
legend	logical value for printing the legend. Default is set to TRUE.
biom.plotting	if TRUE, the plotting imitates the biometrics sort of plotting in papers. Default is set to FALSE. Review the reference Soto et al. (2010) for further details.
corre.fac	is a number to be used to modify the drawing of circels if the option "biom.plotting" is set to TRUE. Notice that default value for this option is set to 0.05
cex	a number to be used for affecting the size of the objects to be drawn. Default is set to 1.
plot.as.trees	if TRUE, the plotting imitates trees for each observation. Default is set to FALSE. Only should be defined to be TRUE if the option "mark" is not defined or NULL.
dimx	the length of the plot for the X-axis. Therefore, not value of 'xc' can be larger than dimx.
dimy	the length of the plot for the Y-axis. Therefore, not value of 'yc' can be larger than dimy.
col.mark	a character having the color to be used for drawing the plot. Default is set to "gray".
col.border	a character having the color to be used for drawing the border of the window plot. Default is set to "black".

### Value

This function returns a spatial distribution plot

### Note

Please, uses with caution, and run first the examples to understand it better.

### Author(s)

Christian Salas-Eljatib

### References

- Soto DP, Salas C, Donoso PJ, Uteau D. 2010. Heterogeneidad estructural y espacial de un bosque mixto dominado por *Nothofagus dombeyi* despues de un disturbio parcial. *Revista Chilena de Historia Natural* 83(3): 335–347.
- Salas C, LeMay V, Nunez P, Pacheco P, Espinosa A. 2006. Spatial patterns in an old-growth *Nothofagus obliqua* forest in south-central Chile. *Forest Ecology and Management* 231:38-46



## Examples

```
#df<-data(spatLlancahue)
#spatdistplot(df, xc="x.coord", yc="y.coord",dimx=130, dimy=70, main.tex ="Spatial - unmark")
#spatdistplot(df, xc="x.coord", yc="y.coord",dimx=130, dimy=70, cex=.7)
#spatdistplot(df, xc="x.coord", yc="y.coord",mark="dbh", dimx=130, dimy=70)
#spatdistplot(df, xc="x.coord", yc="y.coord",mark="dbh", dimx=130, dimy=70,
#   col.mark = "red", col.border="green")
#spatdistplot(df, xc="x.coord", yc="y.coord", mark="dbh", dimx=130, dimy=70,
#   biom.plotting = TRUE)
```

---

speciesList

*Names and other information of plant species (mainly trees)*

---

## Description

This data set provides names (taxonomy), of plant species. Includes codes and name abbreviations used by the Biometrics group at the Center for Ecosystem Modeling (CEM), Universidad Mayor, Santiago, Chile.

## Usage

```
data(speciesList)
```

## Format

A data frame with 63 observations on 31 variables

**nesp** Unique correlative specie number

**spp.ci.name** Species scientific name

**spp.ci.abb** Species scientific name abbreviation

**common.name** Species common name. No blank spaces, no special characters

**common.nameBlank** Species common name. With blank spaces, no special characters

**esp** Species code: code given by CEM Biometrics to identify species for different processing routines

**common.nameLatex** Species common name formatted for Latex

**nTaxon** Unique number of the taxon (i.e., species)

**kingdom** Taxonomic rank Kingdom. In this dataset, all species belong to the Kingdom Plantae

**division** Taxonomic rank division or phylum within the Kingdom

**class** Taxonomic rank Class within the Kingdom

**order** Taxonomic rank Order within the Class

**family** Taxonomic rank Family within the Order

**spp.ci.full** Full scientific name including author

**genus** Taxonomic rank Genus within the Family

**epithet** Specific epithet

**spp.Author** Species author

**subSpp** Subspecies: one of two or more populations of a species varying from one another by morphological characteristics

**subSppAuthor** Subspecies author  
**varSpp** Species variety or varietas  
**varSppAuthor** Variety author  
**formSpp** Form or forma  
**formSppAuthor** Form author  
**commonNamesList** List of common names per species, separated by commas  
**synonyms** Synonyms of the scientific name by which the species has been or is known  
**borCountries** Border countries given the species distribution range  
**habit** Habit. The general appearance, growth form, or architecture e.g., tree, shrub, grass  
**lifeCycle** Life cycle  
**statusOri** Status according to the species origin: Native or Endemic  
**regDist** Distribution range of the species, within Chile administrative regions  
**elevRange** Distribution range of the species, in terms of elevation. Meters above sea level  
**notes** Notes

### Source

Data provided from [https://investigacion.conaf.cl/repositorio/documento/ficha-repositorio.php?redo\\_id=1080946](https://investigacion.conaf.cl/repositorio/documento/ficha-repositorio.php?redo_id=1080946)

### References

Proyecto 004/2016 Lista sistematica actualizada de la flora vascular nativa de Chile, origen y distribucion geografica. VII Concurso del Fondo de Investigacion del Bosque Nativo

---

sppAbundance	<i>Contains information of abundance of plant species in the central-southern Andes of Chile.</i>
--------------	---

---

### Description

Abundance of plant species [50 total] (at parcel scale [100 m<sup>2</sup>]) in burned Araucaria-Nothofagus forests with different levels of fire severity (ie, unburned = unburned, low\_sev = low severity, mid\_sev = medium severity , high\_sev = high severity) in the China Muerta National Reserve, Andes of central-southern Chile.

### Usage

```
data(sppAbundance)
```

### Format

Contains 6 variables, as follows:

**sp.name** name of specie.  
**sp.code.name** code of specie  
**unburned** Abundance of plants unburned.  
**low.sev** Abundance of plants for low severity of burned.  
**mid.sev** Abundance of plants for middle severity of burned.  
**high.sev** Abundance of plants for high severity of burned.

## Source

The data are provided courtesy of Dr. Christian Salas at the Universidad Mayor (Santiago, Chile) and Dr. Andres Fuentes at the Universidad of La Frontera (Temuco, Chile)

## References

- Fuentes A, Salas C, Gonzalez M, Urrutia J, Arroyo P, Santibanez P. 2020. Initial response of understorey vegetation and tree regeneration to a mixed-severity fire in old-growth Araucaria-Nothofagus forests. *Applied Vegetation Science*. 23:210-222.

## Examples

```
data(sppAbundance)
head(sppAbundance)
```

---

sppTraits

*Contains information of functional traits of species.*

---

## Description

Dataset contains 48 observations about about functional trait values for each of the 48 study species, including 23 evergreen and 25 deciduous.

## Usage

```
data(sppTraits)
```

## Format

Contains 17 variables, as follows:

**sp** Abbreviated name of specie.

**sp.name** Name of specie.

**family** Family of specie.

**genus** Genus of specie.

**phyl** Type of phylogeny.

**l.hab** Type of leaf habit.

**leaf** Type of leaf.

**lt** .

**lma** Leaf mass area.

**amass** Photosynthetic capacity per unit leaf mass.

**n.mass** Leaf N content per unit mass.

**p.mass** Leaf P content per unit mass.

**l.lifespan** Leaf life span.

**l.length** Leaf length.

**sem** Seed mass.

**wd** Wood density.

**max.h** Maximum height.

**Source**

The data were provided from DRYAD repository

**References**

- Ameztegui A, Paquette A, Shipley B, Heym M, Messier C, Gravel D. 2016 . Shade tolerance and the functional trait: demography relationship in temperate and boreal forests. *Functional Ecology*, 31: 821-830. DOI:10.1111/1365-2435.12804

**Examples**

```
data(sppTraits)
head(sppTraits)
```

---

stage	<i>Mathematical expression for the Stage function for relating variable Y versus X</i>
-------	--

---

**Description**

Stage

**Usage**

```
stage(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Stage function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**Examples**

```

b0<- 70.51
b1<- 3.7672
b2<- 0.35
params<-c(b0,b1,b2)
X<- c(70)
y<-stage(params,X,intercept=1.3)
plot(y~X,type='p')

```

standVar

*Calculate stand variables for a given tree list***Description**

For each plot and time calculates the following stand variables: density, basal area, quadratic diameter, mean height, height of the dominant trees, and volume.

**Usage**

```

standVar(
  data = data,
  area = sup,
  id = plot.id,
  categories = NA,
  t = NA,
  d = dap,
  h = h.m,
  v = vtot
)

```

**Arguments**

data	data frame having the tree list
area	column name having the plot area in square meters
id	column name with the plot code/number
categories	column names of the variables used as factors to calculate the stand variables
t	column name having the time variable, if not provided the current year is assigned as the time variable
d	diameter at the breast-height, in centimeters
h	total height in meters
v	total volume in cubic meters

**Value**

This function returns a data frame with the stand variables of each plot and time. If categories are given, the result will be a data that will contain the previous one and will have the values of the stand variables for each of the categories. For the calculation of the height of the dominant trees, the TopHeight and Uestimator functions are used (Garcia, 2005).

**Note**

In order to use this function, the packages 'dplyr' and 'data.table' must be previously installed.

**Author(s)**

Christian Salas-Eljatib and Joaquin Riquelme-Alarcon

**Examples**

```
#not yet implemented
```

---

stat.rss

*Statistics for a Stratified Random Sample without replacement*

---

**Description**

Statistics for a Stratified Random Sample without replacement

**Usage**

```
stat.rss(n, y, conf)
```

**Arguments**

n	is the sample size
y	vector with the target variable
conf	is the confidence level in percentage for building the confidence intervals

**Value**

Statistics for a Stratified Random Sample without replacement

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#not yet implemented
```

---

`stat.st`*Statistics for a Stratified Random Sample without replacement*

---

**Description**

Statistics for a Stratified Random Sample without replacement

**Usage**

```
stat.st(tau.yh, n, var.yh, conf, total, L)
```

**Arguments**

<code>tau.yh</code>	is the total value of the variable at the h-th stratum
<code>n</code>	is the sample size
<code>var.yh</code>	is the variance of the variable at the h-th stratum
<code>conf</code>	is the confidence level in (percentage) for building the confidence intervals
<code>total</code>	put 0 when computing at the stratum level, and 1 at the TOTAL level
<code>L</code>	is the total numbers of stratum used

**Value**

Statistics for a Stratified Random Sample without replacement

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#not yet implemented
```

---

`stat.sys`*Statistics for a systematic sample*

---

**Description**

Statistics for a systematic sample

**Usage**

```
stat.sys(y, N, conf, suc.var.est)
```

**Arguments**

y	vector with the target variable
N	population size
conf	is the confidence level in percentage for building the confidence intervals
suc.var.est	is an indicator number 1 for estimating the variance of the sampling based on successive differences and 0 otherwise

**Value**

Statistics for a systematic sample

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#not yet implemented
```

---

statVar

*A statistic summary of tree-level, separated per categories*

---

**Description**

function of summary statistic of variables individuals and separated by categories.

**Usage**

```
statVar(data = data, dap = NA, y = dap, id = plot.id, t = NA, categories = NA)
```

**Arguments**

data	data frame having the tree list
dap	column name with the diameter at breast heigth, if not given can not calculate the statistics for g
y	column names of the y variables
id	column name with the plot codenumber
t	column name having the time variable, if not provided the current year is assigned as the time variable
categories	column names of the variables used as factors

**Value**

This function returns a data frame with the statistical summary of each plot and time for the y variables. If categories are given, the result will be a data that will contain the previous one and will have the values for each of the categories.



**Note**

In order to use this function, the package 'dplyr' must be previously installed. See 'skew.coef' and 'kurt.coef'

**Author(s)**

Christian Salas-Eljatib and Joaquin Riquelme-Alarcon

**Examples**

```
#not yet implemented
```

---

timeSerPlot	<i>Produces a time series plot, of variable Y as a function of X, by an observational unit factor or id.</i>
-------------	--

---

**Description**

Produces a time series plot, of variable Y as a function of X by an observational unit factor or id. Both Y and X must be numeric variables, and the column representing the observational unit, must be a factor. This factor identifies the longitudinal context of the data, for instance, a student being measured on time. Besides, two more factors can be added to the plotting details, in order to represent the potential variability among them.

**Usage**

```
timeSerPlot(
  data = data,
  y = y,
  x = x,
  obs.unit = obs.unit,
  factor1 = NA,
  factor2 = NA,
  only.lines = FALSE,
  ylab = NA,
  xlab = NA,
  linetype.lab = NA,
  factor2.line = TRUE,
  factor2.col = FALSE,
  max.y.all = NA,
  levels.i.want = FALSE,
  col.lev.i.want = FALSE,
  col.lines = FALSE
)
```

**Arguments**

data	a dataframe with at least tree columns representing the response variable ("y"), the main predictor variable ("x"), and a variable indicating the observational unit ("obs.unit").
------	--

y	a character giving the column name of the response variable or variable of interest.
x	a character giving the column name of the main predictor variable. Generally this variable is time.
obs.unit	a character giving the column name containing the info of the observational unit.
factor1	an optional character having the name of a column having a factor variable (e.g., treatment). The default value is set to NULL.
factor2	an optional character having the name of a column having another factor variable (e.g., species). The default value is set to NULL.
only.lines	a logic value if only lines, but not including dots, are going to be drawn in the plot. The default value is set to FALSE.
ylab	Label for the Y-axis
xlab	Label for the X-axis
linetype.lab	is an optional string to be used as the title of the factor being represented by lines. It is only needed if factor1 and factor2 are defined. See example.
factor2.line	a logic value if the second factor, factor2, is going to be segregated according to the type of lines. The default value is set to TRUE.
factor2.col	a logic value if the second factor, factor2, is going to be segregated according to the color of the lines only. The default value is set to FALSE.
max.y.all	A number representing the maximum level of Y-axis for all classes
levels.i.want	A vector having the levels for the factor under study
col.lev.i.want	A vector having the colors to be used for the factor under study

### Value

This function returns a time series plot

### Note

Please, uses with caution, and run first the examples to understand it better.

### Author(s)

Christian Salas-Eljatib

### Examples

```
#data(orange)
# df <- orange
# df$Site <- "Southern site"
# df[df$tree.id==1 | df$tree.id==3,"Site"] <- "Northern site"
# df$Species <- "Douglas-fir"
# df[df$tree.id==2,"Species"] <- "Beech"
# df[df$tree.id==1,"Species"] <- "Radiata pine"
#
# timeSerPlot(df, y="dbh", x="time", obs.unit = "tree.id")
# timeSerPlot(df, y="dbh", x="time", obs.unit = "tree.id", only.lines = T)
#
# # dos opciones de lo solicitado
# timeSerPlot(df, y="dbh", x="time", obs.unit = "tree.id", col.lines = T, only.lines = T)
```

```
# timeSerPlot(df, y="dbh", x="time", obs.unit = "tree.id", col.lines = T, only.lines = F)
#
# timeSerPlot(df, y="dbh", x="time", obs.unit = "tree.id", factor1="Site")
# timeSerPlot(df, y="dbh", x="time", obs.unit = "tree.id", factor1="Site", factor2= "Species")
# timeSerPlot(df, y="dbh", x="time", obs.unit = "tree.id", factor1="Site", factor2= "Species",
#             factor2.col = T, only.lines = T)
```

---

topHeight

---

*Calculate the height of the dominant trees in a stand*


---

## Description

Computes dominant height of a sample plot

## Usage

```
topHeight(height, area)
```

## Arguments

height	vector or list of tree heights, sorted by increasing dbh
area	column name having the plot area in ares (1 hectare = 100 ares)

## Value

Calculate top height using the interpolated U-estimator. Reference: Garcia,O. and Batho,A. Western Journal of Applied forestry 20(1), 64-68. 2005.

## Note

This function need the 'Uestimator' function

## Author(s)

Garcia, O. and Batho, A.

## Examples

```
N <- 760
A <- 0.03
n <- round(N * A)
heights <- rnorm(n, 15, 4)
m <- round(n / (100 * A))
choose(n, m) # subsets
Uestimator(sort(heights), m)
# find the largest in each subset...
h <- combn(heights, m, FUN=max)
# the mean is the top height U-estimate:
mean(h)
topHeight(sort(heights), 100 * A)
```

---

topHeight.tradi	<i>Calculate the height of the dominant trees in a stand</i>
-----------------	--

---

**Description**

Conventional top height estimate

**Usage**

```
topHeight.tradi(heights, ares)
```

**Arguments**

heights	vector or list of tree heights, sorted by increasing dbh
ares	column name having the plot area in ares (1 hectare = 100 ares)

**Value**

Calculate top height using the conventional top height estimate Reference: Garcia,O. and Batho,A. Western Journal of Applied forestry 20(1), 64-68. 2005.

**Note**

This function need the 'Uestimator' function

**Author(s)**

Christian Salas-Eljatib

**Examples**

```
#not yet implemented
```

---

trailCameraTrap	<i>Contains information of Camera trap data on medium to large terrestrial mammals collected at 54 camera stations in Ruaha National Park, southern Tanzania.</i>
-----------------	---

---

**Description**

Dataset contains 14604 observations and sampling was carried out for two months during the dry season of 2013 and two months during the wet season of 2014. Each camera station is associated with a randomly placed camera and a trail-based camer, with the aim of comparing communities resulting from the two camera trap placement strategies.

**Usage**

```
data(trailCameraTrap)
```

### Format

Contains 6 variables, as follows:

**reference** Number of observation od datasets.

**placement** Type of "placement" placed in each station (random or trail).

**season** Season where were made the samplings.

**station** Station where were collected the data.

**specie** Name of specie medium to large terrestrial mammals.

**date.time** The date and time of each photographic event is also given.

### Source

The data are provided courtesy of Dr. Jeremy Cusack at the Universidad Mayor (Santiago, Chile)

### References

- Cusack J, Dickman A, Rowcliffe M, Carbone C, Macdonald D, Coulson T. 2016 . Random versus game trail-based camera trap placement strategy for monitoring terrestrial mammal communities. PLoS ONE 10(5): e0126373.

### Examples

```
data(trailCameraTrap)
head(trailCameraTrap)
```

---

traits	<i>Contains information of functional traits of vegetative species in Chile.</i>
--------	--

---

### Description

Functional traits of vegetative species in Chile. Includes column with codified name (esp)

### Usage

```
data(traits)
```

### Format

**esp** species codified name

**shadeTolerance** indicates the species tolerance to shade. There are three main classes: shade-tolerant, shade-midtolerant and shade-intolerant

**spp.ci.name** Scientific name.

**spp.ci.abb.** .

**wd** wood density in kg per cubic meters.

### Source

Some of the information on shade tolerance can be found in Soto et al 2010. Heterogeneidad estructural y espacial de un bosque mixto dominado por *Nothofagus dombeyi* despues de un disturbio parcial. Revista Chilena de Historia Natural 83: 335-347, 2010

---

treegrowth	<i>Contains information of.</i>
------------	---------------------------------

---

**Description**

Data contains.

**Usage**

```
data(treegrowth)
```

**Format**

Contains 7 variables, as follows:

- tree.id** .
- forest** .
- habitat** .
- tree.code** .
- age** .
- dbh** .
- htot** .

**Source**

The data were provided.

**References**

not yet

**Examples**

```
data(treegrowth)
head(treegrowth)
```

---

treevol	<i>Diameter, height and volume for Black Cherry Trees</i>
---------	---

---

**Description**

This data set provides measurements of the diameter, height and volume of timber in 31 felled black cherry trees. This dataframe is a slight modification to the original dataframe "trees" from the datasets R package.

**Usage**

```
data(treevol)
```

**Format**

A data frame with 31 observations on three variables

**dbh** diameter at breast height, in cm

**htot** total height, in m

**volume** volume of timber, in cubic meters

**Source**

Ryan, T. A., Joiner, B. L. and Ryan, B. F. (1976) The Minitab Student Handbook. Duxbury Press.

**Examples**

```
#pairs(treevol, panel = panel.smooth, main = "treevol dataframe")
#plot(volume ~ dbh, data = treevol, log = "xy")
#coplot(log(volume) ~ log(dbh) | htot, data = treevol,
#       panel = panel.smooth)
#summary(m1 <- lm(log(volume) ~ log(dbh), data = treevol))
#summary(m2 <- update(m1, ~ . + log(htot), data = treevol))
#anova(m1,m2)
```

---

treevolroble	<i>Contains tree-level variables for roble (Nothofagus obliqua) in the Rucamanque experimental forest, near Temuco, Chile.</i>
--------------	--

---

**Description**

These are tree-level measurement data of sample trees in the Rucamanque experimental forest, near Temuco, in the Araucania region in south-central Chile, measured in 1999. The data are the same as in the dataframe "treevolruca", but only having observations for the species roble (*Nothofagus obliqua*).

**Usage**

```
data(treevolroble)
```

**Format**

Contains tree-level variables, as follows:

**tree.no** Tree id

**spp** Species

**dbh** Diameter at breast height, in cm

**htot** Total height (m)

**d6** Upper-stem diameter at 6 m, in cm

**vtot** Tree gross volume, in m<sup>3</sup> with bark.

**Source**

The data are provided courtesy of Dr. Christian Salas at the Universidad Mayor (Santiago, Chile).

## References

Salas C. 2002. Ajuste y validacion de ecuaciones de volumen para un relicto del bosque de Roble-Laurel-Lingue [Fitness and validation of volume equations for a relict forest of Roble-Laurel-Lingue]. Bosque 23(2): 81-92.

## Examples

```
data(treevolroble)
head(treevolroble)
```

---

treevolruca	<i>Contains tree-level variables of several species in the Rucamanque experimental forest, near Temuco, Chile.</i>
-------------	--

---

## Description

These are tree-level measurement data of sample trees in the Rucamanque experimental forest, near Temuco, in the Araucania region in south-central Chile, measured in 1999. The following species are part of the data: laurel (*laurelia sempervirens*), lingue (*Persea lingue*), olivillo (*Aextocicon punctatum*), roble (*Nothofagus obliqua*), tepa (*Laurelissis philippiana*), y tineo (*Weinmannia trichosperma*).

## Usage

```
data(treevolruca)
```

## Format

Contains tree-level variables, as follows:

**tree.no** Tree id.  
**spp** Species.  
**dbh** Diameter at breast height, in cm.  
**htot** Total height, in m.  
**d6** Upper-stem diameter at 6 m, in cm.  
**vtot** Tree gross volume, in m<sup>3</sup> with bark.

## Source

The data are provided courtesy of Dr. Christian Salas of the Universidad Mayor (Santiago, Chile). The data were used in the study of Salas (2002).

## References

Salas C. 2002. Ajuste y validacion de ecuaciones de volumen para un relicto del bosque de Roble-Laurel-Lingue [Fitness and validation of volume equations for a relict forest of Roble-Laurel-Lingue]. Bosque 23(2): 81-92.

## Examples

```
data(treevolruca)
head(treevolruca)
```



---

Uestimator

---

*Calculate the height of the dominant trees in a stand*


---

**Description**

Calculate the U estimator

Calculate the U estimator

**Usage**

```
Uestimator(heights, trees.per.are)
```

**Arguments**

heights                vector or list of tree heights, sorted by increasing dbh

trees.per.are        length(heights) / (plot area in ares). If not integer, it is rounded to the nearest integer, with a warning.

**Value**

Calculate U-estimator, for integer trees/are

**Note**

see topHeight()

**Author(s)**

Christian Salas-Eljatib and Joaquin Riquelme-Alarcon

**References**

Garcia,O. and Batho,A. Western Journal of Applied Forestry 20(1), 64-68. 2005.

**Examples**

```
N <- 760
A <- 0.03
n <- round(N * A)
heights <- sort(rnorm(n, 15, 4))
m <- round(n / (100 * A))
trees.per.are<-m
choose(n, m)        # subsets
Uestimator(heights, trees.per.are)
```

---

`Uestimator.alt`*Alternative Uestimator*

---

**Description**

Alternative implementation of Uestimator, useful for translation to C, VisualBasic, or other procedural languages.

**Usage**

```
Uestimator.alt(height, trees.per.are)
```

**Arguments**

<code>height</code>	vector or list of tree heights, sorted by increasing dbh
<code>trees.per.are</code>	<code>length(heights) / (plot area in ares)</code> . If not integer, it is rounded to the nearest integer, with a warning.

**Details**

Description function

**Value**

Calculate U-estimator, for integer trees/are

**Note**

see `topHeight()`

**Author(s)**

Christian Salas-Eljatib

**References**

Garcia,O. and Batho,A. Western Journal of Applied Forestry 20(1), 64-68. 2005.

**Examples**

```
#not yet implemented
```

---

valesta	<i>Computes validation statistics</i>
---------	---------------------------------------

---

### Description

This function computes validation statistics, as the RMSD, AD and AAD of any response vector 'y.pred' using as real values the vector 'y.obs'. The computed statistics are: root mean square differences (RMSD); aggregated difference (AD); and aggregated of the absolute value differences (AAD)

### Usage

```
valesta(y.obs = y.obs, y.pred = y.pred)
```

### Arguments

y.obs	observed values of the variable of interest
y.pred	predicted values of the variable of interest

### Details

Be sure that both predicted and observed values are expressed in the same measurement unit. Be careful if the model uses a transformation of the response variable.

### Value

This function returns the following statistics as a vector: (RMSD,RMSD.p,AD,AD.p,AAD,AAD.p); where RMSD.p stands for RMSD expressed as a percentage, and the same applied to AD.p and AAD.p.

### Note

Please check the references for further details.

### Author(s)

Christian Salas-Eljatib

### References

Salas C, Ene L, Gregoire TG, Nasset E, Gobakken T. 2010. Modelling tree diameter from airborne laser scanning derived variables: a comparison of spatial statistical models. Remote Sensing of Environment 114 (6):7-1285.

### Examples

```
#creating a fake dataframe
#set.seed(1234)
#df <- as.data.frame(cbind(Y=rnorm(30, 30,9), X=rnorm(30, 450,133)))
#df
#fitting a candidate model
#mod1 <- lm(Y~X, data=df)
```

```
#using the function
#valesta(y.obs=df$Y,y.pred=fitted(mod1))
```

---

valida.gr	<i>Computes validation statistics for an already fitted annualized growth model</i>
-----------	---

---

## Description

Computing validation statistics for a growth model being already fitted with annualized coefficients, as in Weiskittel et al (2007). The computation is based on predicted the periodic annual increment (pai)

## Usage

```
valida.gr(mod = mod, data, vary = vary, dt = dt)
```

## Arguments

mod	is an already annualized fitted growth model
data	is a dataframe
vary	is the response variable
dt	is the period length

## Value

This function returns the following statistics as a vector: (RMSD,RMSD.p,AD,AD.p,AAD,AAD.p); where RMSD.p stands for RMSD expressed as a percentage, and the same applied to AD.p and AAD.p.

## Note

It uses the function valesta(). Please check the references for further details.

## References

Weiskittel AR, Garber SM, Johnson GP, Maguire DA, Monserud RA. 2007. Annualized diameter and height growth equations for Pacific Northwest plantation-grown Douglas-fir, western hemlock, and red alder. Forest Ecology and Management. 250: 266-278

## Examples

```
#Not yet implemented
```

---

weibull	<i>Mathematical expression for the Weibull function for relating variable Y versus X</i>
---------	--

---

**Description**

Weibull

**Usage**

```
weibull(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Weibull function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Yang RC, A Kozak & JH Smith (1978) The potential of Weibull type functions as flexible growth curves. Canadian Journal of Forest Research 8(2), 424 431.

**Examples**

```
b0<- 36.62
b1<- 0.09
b2<- 0.7
params<-c(b0,b1,b2)
X <- c(70)
y<-weibull(params,X,intercept=1.3)
plot(y~X,type='p')
```

---

wykoff	<i>Mathematical expression for the Wykoff function for relating variable Y versus X</i>
--------	---

---

**Description**

Wykoff

**Usage**

```
wykoff(params, X, intercept = NA)
```

**Arguments**

params	Vector with value of parameters (The values must be added in an orderly way, see example)
X	Vector containing the values for the predictor variable
intercept	Value of intercept of function. Default value is 0.

**Details**

Mathematical expression for the Wykoff function for relating variable Y versus X

**Value**

The value of the function for every value of the vector X and the parameters.

**Author(s)**

Christian Salas.

**References**

- Wykoff WR, NL Crookston & AR Stage (1982) Users guide to the Stand Prognosis Model. USDA For. Serv. Gen. Tech. Rep. INT 133, USA. 112 p.

**Examples**

```
b0<- 3.4883
b1<- 10.44
params<-c(b0,b1)
X <- c(70)
y<-wykoff(params,X,intercept=1.3)
plot(y~X,type='p')
```

# Index

## \*Topic **datasets**

- airquality, [6](#)
- annualppCities, [6](#)
- araucaria, [7](#)
- baiTreelines, [14](#)
- bears, [18](#)
- bears2, [19](#)
- biomass, [21](#)
- carbohydrateTreelines, [21](#)
- chicksw, [23](#)
- demograph, [28](#)
- election, [30](#)
- eucaleaf, [31](#)
- fertilizaexpe, [38](#)
- fishgrowth, [39](#)
- floraChile, [40](#)
- forestHawaiian, [41](#)
- hawaii, [46](#)
- idahohd, [49](#)
- invasivesRCI, [53](#)
- lleuque, [55](#)
- orange, [65](#)
- pinaster, [66](#)
- pinusContorta, [67](#)
- pinusSpp, [68](#)
- plantsHawaiian, [69](#)
- plotLleuque, [70](#)
- presenceIce, [72](#)
- pspllancahue, [74](#)
- ptaeda, [75](#)
- radiatapl, [77](#)
- radiosCopa, [78](#)
- regeneraNothofagus, [81](#)
- simula, [86](#)
- sludge, [92](#)
- snaspeChile, [93](#)
- spatAustria, [94](#)
- speciesList, [97](#)
- sppAbundance, [98](#)
- sppTraits, [99](#)
- trailCameraTrap, [108](#)
- traits, [109](#)
- treegrowth, [110](#)

- treevol, [110](#)
- treevolroble, [111](#)
- treevolruca, [112](#)

## \*Topic **package**

- biometria-package, [4](#)

- a.cdf, [4](#)
- a.contrast, [5](#)
- airquality, [6](#)
- annualppCities, [6](#)
- araucaria, [7](#)
- assignDomi, [8](#)
- assignLPclassi, [9](#)
- assignspp, [10](#)
- assignsTol, [11](#)
- asympt, [12](#)
- asymptOri, [13](#)
- baiTreelines, [14](#)
- bal, [15](#)
- balmodi, [16](#)
- bat, [17](#)
- bears, [18](#)
- bears2, [19](#)
- beta.rich.si, [20](#)
- biomass, [21](#)
- biometria (biometria-package), [4](#)
- biometria-package, [4](#)
- carbohydrateTreelines, [21](#)
- chicksw, [23](#)
- ClarkE, [23](#)
- comparCorreFunc, [24](#)
- creaIncrData, [25](#)
- curtis, [27](#)
- deleteRight, [28](#)
- demograph, [28](#)
- descstat, [29](#)
- election, [30](#)
- eucaleaf, [31](#)
- expMod, [32](#)
- expModV1, [33](#)
- expModV2, [34](#)

- extractRight, 35
- fancypairs.panel, 36
- fertilizaexpe, 38
- findColumn.byname, 38
- fishgrowth, 39
- floraChile, 40
- forestHawaiian, 41
- geomean, 42
- gompertz, 43
- gompertzMod, 44
- gtree, 45
- hawaii, 46
- hossfeld, 47
- huang, 48
- iccMeff, 49
- idahohd, 49
- inequality, 50
- interp.a, 52
- interp.b, 52
- invasivesRCI, 53
- inverse, 54
- kurt.coef, 55
- lleuque, 55
- loaded.pack, 56
- loessfit, 56
- logistic, 57
- logisticMod, 58
- lrt, 59
- meyer, 60
- mmenten, 61
- moda, 62
- naslund, 62
- nele.list, 63
- nndist, 64
- orange, 65
- pinaster, 66
- pinusContorta, 67
- pinusSpp, 68
- plantsHawaiian, 69
- plotLleuque, 70
- poder, 70
- poissonWeibull, 71
- presenceIce, 72
- press (geomean), 42
- prodan, 73
- pspLlancahue, 74
- ptaeda, 75
- pvalue.t, 75
- pvalue.z, 76
- qmd, 77
- radiatap1, 77
- radiosCopa, 78
- ranking2, 79
- ratkowsky, 79
- rdRoraco, 80
- regeneraNothofagus, 81
- richards, 82
- schnute, 83
- schumacher, 84
- sibbesen, 85
- simula, 86
- sizeDist.byGroupAllPlot, 87
- sizeDistPlot, 89
- skew.coef, 91
- sludge, 92
- snaspeChile, 93
- spatAustria, 94
- spatdistplot, 95
- speciesList, 97
- sppAbundance, 98
- sppTraits, 99
- stage, 100
- standVar, 101
- stat.rss, 102
- stat.st, 103
- stat.sys, 103
- statVar, 104
- timeSerPlot, 105
- topHeight, 107
- topHeight.tradi, 108
- trailCameraTrap, 108
- traits, 109
- treegrowth, 110
- treevol, 110
- treevolroble, 111
- treevolruca, 112
- Uestimator, 113
- Uestimator.alt, 114
- valesta, 115
- valida.gr, 116
- weibull, 117
- wykoff, 118