

**Função final:****Help**

biomassa.bd                      package: nenhum                      R Documentation

Cálculo da biomassa de árvores da floresta atlântica

**Descrição:**

Calcula a equação de biomassa de árvores de floresta de mata atlântica, a partir das circunferências e alturas dos indivíduos vegetais localizados em um data.frame e difencia esses resultados para cada tipo de estágio de floresta (inicial e intermediária-tardia). Segundo as necessidades do usuário, também gera de forma opcional um gráfico básico de diagnóstico (boxplot) e um modelo de regressão linear.

**Uso:**

```
biomassa.bd<-function(x,y,z,grafico=T)
```

**Argumentos:**

x: variável de classe numérica. Faz referência a uma coluna com dados contida em um data.frame, os quais correspondem a medidas de circunferência das árvores em formato centímetro (cm).

y: variável de classe numérica. Faz referência a uma coluna com dados contida em um data.frame, os quais correspondem a medidas de altura das árvores em formato metros (m).

z: variável de classe factor. Faz referência a outra coluna contida em um data.frame, cujos valores correspondem a um tipo de estágio de floresta.

grafico: base para geração de gráfico. No caso (TRUE) gerará um boxplot e o modelo de regressão linear, se (FALSE) não gerará os gráficos.

**Detalhes:**

Para que os cálculos das equações e a geração dos gráficos sejam realizados

corretamente, o preenchimento dos dados no data.frame deverão seguir as seguintes pautas:

-O data.frame deve apresentar valores em todas as linhas. Se houvesse presença de NAs, a função não se efetua.

-A coluna com os valores que apresentem as categorias dos estagios sucessionais terá que estar nomeada como "estagios" na planilha que será importada ao software R. Como parte do resultado, a função retornará um data.frame com as variáveis na seguinte ordem:  
"Estagios", "Circunferencia", "Altura", "DAP", "Biomassa".

Valor:

Se grafico=T, a função gera um gráfico boxplot que analisa a biomassa para cada tipo de estágio de floresta e um modelo de regressão linear que relaciona o DAP calculado com a altura para cada indivíduo vegetativo e retorna um data.frame com duas colunas adicionadas, resultado dos cálculos das equações. Se gráfico=F, a função retornará apenas o data.frame. As duas colunas a ser adicionadas correspondem a:

DAP                      resultado da equação do DAP na unidade (cm):  $DAP = \text{circunferência}/\pi$

Biomassa                resultado da equação proposta por Burger & Delitti (2008) nas unidades (Mg/hectare) para calcular a biomassa de árvores na floresta de mata

$$\text{atlântica: } \text{LnDW} = -3.676 + 0.951 \text{Ln}d^2h$$

Avisos:

A função é interrompida quando os argumentos "x" e "y" não são da classe numérica e/ou o argumento "z" da classe factor. Quando a função gera o data.frame, retornará o aviso de se os gráficos são ou não gerados, dependendo da opção do argumento "gráfico" que o usuario escolheu.

Os cálculos das equações DAP e Biomassa são realizados independentemente da quantidade de parâmetros disponíveis no data.frame, mas o usuário deve adequar as colunas que corresponderiam aos argumentos "x", "y" e "z" na planilha antes de importá-la para o software R.

Para que o cálculo seja realizado corretamente, o argumento "x" terá que corresponder na planilha à coluna dos dados referente à circunferência em formato cm, o argumento "y" corresponderá à coluna da altura em metros e, o argumento "z", à coluna das categorias dos estágios sucessionais da floresta em estudo.

A função realizará a transformação da escala métrica do resultado da Biomassa de kg por metro quadrado diretamente a Mg por hectare, como recomendado por Burger & Delitti (2008).

Autora:

Susana López Caracena - Instituto de Biociências USP  
selc@usp.br

Referências:

-Burger, D.M. & Delitti, W.B.C. 2008. Modelos alométricos para estimativa de fitomassa de Floresta Atlântica secundária no sudeste do Brasil. *Biota Neotrop.*, 8(4)

Exemplo 1:

```
dados<-read.csv("dados_fun.csv", sep=";",dec=",") #Importa-se a planilha exemplo "dados"
x<-dados$circunferencia #Objeto x recebe os valores que correspondem à
coluna dos dados da circunferência
(cm) da planilha exemplo "dados"
y<-dados$altura #Objeto y recebe os valores que correspondem à coluna dos
dados da altura (m) da planilha
z<-dados$estagios #Objeto z recebe as categorias que correspondem à coluna
dos estágios sucessionais da
floresta em estudo da planilha
biomassa.bd(dados$circunferencia,dados$altura, dados$estagios, grafico="T")
#Testa-se a função, indicando
a opção de retorno do fataframe com gráficos
```

Exemplo 2:

```
dados<-read.csv("dados_fun.csv", sep=";",dec=",")
x<-dados$circunferencia
y<-dados$altura
z<-dados$estagios
biomassa.bd(dados$circunferencia,dados$altura, dados$estagios, grafico="F")
#Testa-se a função, indicando
a opção de retorno do fataframe sem gráficos.
```

## Código

Para testar a função, por favor salve o arquivo: [dados\\_fun.csv](#)

```
biomassa.bd<-function(x,y,z,grafico=F)
{
  if(class(x)!="numeric") stop ("x precisa ser da classe numérica") #A
função avisará se a variável "x" não é de classe numérica, neste caso a
função dará o seguinte aviso: "x precisa ser da classe numérica"
  if(class(y)!="numeric") stop ("y precisa ser da classe numérica") #A
função avisará se a variável "y" não é de classe numérica, neste caso a
função dará o seguinte aviso: "y precisa ser da classe numérica"
  if(class(z)!="factor") stop ("z precisa ser da classe factor")      #A
função avisará se a variável "z" não é de classe numérica, neste caso a
função dará o seguinte aviso: "z precisa ser da classe factor"
  {                                                                    #Abro
chave onde será escrita a função
  if(grafico=="T")                                                    #Utilizo o
comando if com o opcional else para executar códigos de maneira condicional.
Dentro do parênteses conterà uma condição lógica, que terá como resultado
TRUE para que a função gere os gráficos boxplot e o modelo de regressão
linear
  {                                                                    #Abro chave onde
será escrito o bloco de códigos que será executado se a condição do
parênteses for TRUE
    cat("\t Os gráficos são gerados\n")                               #No caso de que
na função seja escrito grafico=T, o resultado da função dará o seguinte
aviso: "Os gráficos são gerados"
    dap <- (x/pi)                                                      #Equação para
calcular o DAP (diâmetro à altura do peito) a partir da circunferência "x"
    biomas <- exp(-3.676+0.951*log(dap^2*y))                          #Cálculo da
equação de biomassa (Burger&Delitti,2008), com as informações de DAP e
altura "y". Equação:  $\ln DW = -3.676 + 0.951 \ln D^2 h$ 
    biomassa <- biomas*10                                              #O resultado da
equação da biomassa dado em kg/m2 é transformado nas unidades de Mg/hectare
    resultado <- data.frame(z, x, y, dap, biomassa)                  #Incluimos que
sejam adicionadas ao dataframe existente as novas informações calculadas
(DAP e biomassa)
    colnames(resultado) <- c("Estagios","Circunferencia", "Altura",
"DAP","Biomassa")           #Nomeamos novamente os objetos do dataframe
    par(mfrow=c(1,2))                                                #Utilizo a função
par para construir uma figura onde apareça o gráfico boxplot e o modelo de
regressão linear
    par(bty="l", family="serif", tcl=0.4)
    boxplot(resultado$Biomassa~resultado$Estagios, ylab = "Biomassa", names
= (levels(dados$estagios)))    #Gero o gráfico boxplot para apresentar a
biomassa por categorias de estágio de floresta (inicial e intermediária-
tardia)
    plot(dap~y, bty="l", tcl=0.3, xlab = "Altura" ,ylab = "DAP")
#Plota "DAP" em função da altura "y"
```

```

    m.lin<-lm(dap~y)                                #Gera o modelo de
regressão linear através da função lm
    abline(m.lin, col="red", lwd = 2)                #Com a função
abline gera a reta da função
    return(resultado)                                #Gera o dataframe
com as novas colunas calculadas (DAP e biomassa) e cria os dois gráficos
    }else                                            #0 opcional else
executará os comandos que não gerarão gráficos no caso de que a condição seja
FALSE
    {                                                #Abro chave onde
será escrito o bloco de códigos que serão executados se a condição do
parênteses for FALSE
    cat("\tOs gráficos não são gerados\n")          #No caso de que
na função seja escrito grafico=F, o resultado da função dará o seguinte
aviso: "Os gráficos não são gerados"
    dap <- (x/pi)                                    #Equação para
calcular o DAP
    biomas <- exp(-3.676+0.951*log(dap^2*y))          #Cálculo da
equação de biomassa (Burger&Delitti,2008)
    biomassa <- biomas*10                            #0 resultado da
equação é transformado nas unidades de Mg/hectare
    resultado <- data.frame(z, x, y, dap, biomassa)   #Incluimos que
sejam adicionadas ao dataframe existente as novas informações calculadas
(DAP e biomassa)
    colnames(resultado) <- c("Estagios","Circunferencia", "Altura",
"DAP","Biomassa")    #Nomeamos novamente os objetos do dataframe
    return(resultado)                                #Fecha a função
biomassa.bd() mostrando o dataframe com as novas coluna na tabela
    }}}

```

From:

<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2017:alunos:trabalho\\_final:selc:funcao\\_final:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2017:alunos:trabalho_final:selc:funcao_final:start)Last update: **2020/08/12 06:04**