

## HELP\_DA\_FUNÇÃO

Função : Debaixo dos caracóis dos seus cabelos

Por: Anne Elise Landine

```
cachos <- function(comprI, comprF, tipos = length(comprI), força = 0.2)
{
  if(!require(install.load)) install.packages('install.load');
  library(install.load)
  install.load::install_load('dplyr') ##instalando pacotes necessários
  if(class(comprI) != 'numeric' | class(comprF) != "numeric")
  {stop("O vetor de comprimento inicial e de comprimento final devem ser da
classe numeric.")}
  if (length(comprI) != length(comprF)) ## parar se os comprimentos iniciais
não tiverem seus correspondentes nos valores de comprimento final.
  {stop("nem todos os comprimentos iniciais possuem comprimentos finais,
após a aplicação de força.")}
  if (length(comprI) != tipos) #não é algo que faça diferença no cálculo em
si, mas é mais uma etapa de interação, para que a pessoa veja com cautela os
tipos diferentes de cabelo que mediu.
  {stop("meça todos os diferentes tipos de curvaturas de cachos. Caso você
estime que tenha 3 diferentes tipos de cachos, por exemplo (ex. 3A, 4B, 2C),
meça o comprimento inicial de todos.")}
  if (is.null(força))
  {stop("forneça o valor do peso usado para alongar o cabelo, em Kg")} #é
preciso fornecer o peso que foi utilizado para "esticar" o cabelo, para
calcular o coeficiente de deformação elástica
  class(tipos) <- 'integer'
  data <- data.frame(comprI, comprF, rep(força, length(comprI))) #criando um
dataframe com os valores inseridos nos argumentos da função
  colnames(data) <- c("comprI", "comprF", "peso") #mudando o nome das
colunas
  data$peso <- data$peso *10 #convertendo Kg para N.
  data$alongamento <- (data$comprF - data$comprI) #calculando o delta de
deformação
  data$K <- data$peso/data$alongamento #constante elástica.###quanto menor o
K mais enrolado
  data[order(data$K),] ##menores valores: cabelo menos curvado.
  ###Cálculo da deformação elástica para diferentes forças aplicadas, visto
que já temos a contante elástica ##classificar entre 2 e 4
  data1 <- data.frame(rep(data$comprI, each = 5),
                      rep(data$K, each = 5)) #criando um dataframe com os
valores inseridos nos argumentos da função repetidos 5x, para calcular a
deformação do cabelo com 5 valores de força diferentes
  colnames(data1) <- c("comprI", "K") #mudando o nome das colunas
  data1[order(data1$K),] #ordenando os valores em função da constante
elástica
  data1$id <- rep(1:length(data$K), each = 5, len = length(data1$K))
#criando uma coluna id, para saber qual identificação do cabelo. Quais
valores correspondem ao montante de cabelo medido e fornecido na primeira
```

```
colocação dos argumentos comprI e comprF, e assim por diante.
len1 <- length(data$K) #tamanho da coluna com os valores da constante
elástica
len2 <- length(data1$K)#tamanho da coluna com os valores da constante
elástica
data1$força <- rep(seq(3,5, len = len1), length.out = len2) #criando
valores de força para calcular o delta de deformação para diferentes
aplicações de peso no cabelo
data1$alongamento <- data1$força/data1$K #calculando o delta de deformação
para diferentes aplicações de força
data1<-data1%>%
  group_by(id) %>%
  mutate(Threshold = mean(alongamento)) ##media de alongamento de cada id
data1$indice <- data1$Threshold/data1$comprI ##calculo para saber quantas
vezes mais o cabelo sofre deformação ao aplicar-se uma força x, em relação
ao seu comprimento inicial
a <- split(data1, data1$id) #separando os dataframes por id
data1 <- as.data.frame(data1) #retornando a classe dataframe
data$id <- rep(1:length(a)) #criando uma coluna com os ids (identificação)
data[order(data$K),] #ordenando os valores em função da constante elástica
#Liso #cálculos para um cabelo sem ondução
if(sum(data1$indice < 0.05)>0) ##valores de deformação mais baixos.
{liso <- data1[data1$indice < 0.1,]
liso$tipo <- "1" #tipo de cabelo 1: liso
data2 <- liso} ##separando a categoria "liso"
if(sum(data1$indice < 0.05) == 0) ##caso não tenham valores onde o índice
seja menor do que 0.05
{
  data2 <- NA
}
#Ondulado #cabelo ondulado varia o seu tamanho entre 0.05x e 0.5x
if(sum(data1$indice >= 0.05 & data1$indice <= 0.5) > 0) #os cálculos da
variação dos tipos de cabelo(A,B,C) foram feitos a priori, baseado em dados
reais de cabelos com diferentes curvaturas
{
  ondulado <- data1[data1$indice >= 0.05 & data1$indice <= 0.5,]
  dif <- 0.5 - 0.05 #calculando o intervalo de diferença entre os valores
correspondentes a curvatura ondulada
  ond <- dif/3 #dividindo o intervalo de curvatura ondulado em três partes
(A,B,C)
  if(sum(ondulado$indice >= 0.1 & ondulado$indice <= ond) > 0) #se o
cabelo ondulado varia entre 0.1 e valor da diferença do intervalo/3 -> 2A
  {
    onA <- ondulado[ondulado$indice >= 0.1 & ondulado$indice <= ond,]
    onA$tipo<- paste0("2A")
    #ondulado2 <- onA
    data2 <- rbind(data2,onA) ##calculo para separar o cabelo na
categoria "ondulado (2", A"
  }
  if(sum(ondulado$indice > ond & ondulado$indice <= (ond*2)) > 0)
```

```

#se o cabelo ondulado varia entre a diferença entre os intervalos/3 e duas
vezes o valor da diferença -> 2B
    {
        onB <- ondulado[ondulado$indice > ond & ondulado$indice <=
(ond*2),]
        onB$tipo <- paste0("2B")
        #ondulado2 <- rbind(onA, onB)
        data2 <- rbind(data2,onB)
        }##calculo para separar o cabelo na categoria "ondulado (2), B"
        if(sum(ondulado$indice > (ond*2) & ondulado$indice <= (ond*3)) >
0) #se o cabelo ondulado varia entre 2x o valor da diferença dos intervalos
e 3x o valor da diferença dos intervalos -> 2C
        {
            onC <- ondulado[ondulado$indice > (ond*2) & ondulado$indice
<= (ond*3),]
            onC$tipo <- paste0("2C")
            #ondulado2 <- rbind(ondulado2, onC)
            data2 <- rbind(data2,onC)
            }##calculo para separar o cabelo na categoria "ondulado
(2), C"
        }
    #Cacheado
    if(sum(data1$indice > 0.5 & data1$indice <= 3) > 0) #cabelos cacheados
variam entre 0.5x e 3x seu tamanho.
    # a mesma lógica explicada para os cabelos ondulados foi aplicada para
calcular os valores relativos aos cabelos cacheados (3) e crespos (4)
    {
        cacheado <- data1[data1$indice > 0.5 & data1$indice <= 3,]
        dif2 <- 3-0.5
        cach <- dif2/3 #dividindo o intervalo de curvatura ondulado em três
partes (A,B,C)
        ##dividindo o intervalo de curvatura cacheado em três partes (A,B,C)
        if(sum(cacheado$indice >= 0.5 & cacheado$indice <= cach) > 0 )
        {
            caA <- cacheado[cacheado$indice >= 0.5 & cacheado$indice <= cach,]
            caA$tipo<- paste0("3A")
            data2 <- rbind(data2, caA)} ##calculo para separar o cabelo na categoria
"cacheado (3), A"
            if(sum(cacheado$indice > cach & cacheado$indice <= (cach*2)) > 0)
            {
                caB <- cacheado[cacheado$indice > cach & cacheado$indice <= (cach*2),]
                caB$tipo <- paste0("3B")
                #cacheado2<- rbind(cacheado2, caB)
                data2 <- rbind(data2, caB)##calculo para separar o cabelo na categoria
"cacheado (3), B"
            }
            if(sum(cacheado$indice > (cach*2) & cacheado$indice <= (cach*3)) > 0)
            {
                caC <- cacheado[cacheado$indice > (cach*2) & cacheado$indice
<= (cach*3),]
                caC$tipo <- paste0("3C")
            }
        }
    }

```

```
#cacheado2 <- rbind(cacheado2,caC)
data2 <- rbind(data2, caC)} ##calculo para separar o cabelo na
categoria "cacheado (3), c"
}
#Crespo #os cabelos crespos aumentam 3x ou mais do que o comprimento
inicial, quando aplicada uma força x que o deforme.
if(sum(data1$indice > 3) > 0)
{
  crespo <- data1[data1$indice > 3,]
  cres <- 1.5/3 #dividindo o intervalo de curvatura crespo em três partes
(A,B,C)
if(sum(crespo$indice >= 0.5 & crespo$indice <= cres) > 0)
{
  crA <- crespo[crespo$indice >= 0.5 & crespo$indice <= cres,]
  crA$tipo<- paste0("3A")
  #crespo2 <- crA
  data2 <- rbind(data2, crA)
}##calculo para separar o cabelo na categoria "crespo (4), A"
if(sum(crespo$indice > cach & crespo$indice <= (cres*2)) > 0)
{
  crB <- crespo[crespo$indice > cach & crespo$indice <= (cres*2),]
  crB$tipo <- paste0("3B")
  #crespo2 <- rbind(crespo2, crB)
  data2 <- rbind(data2, crB)}
##calculo para separar o cabelo na categoria "crespo (4), B"
if(sum(crespo$indice > (cres*2) & crespo$indice <= (cres*3)) > 0)
{
  crC <- crespo[crespo$indice > (cres*2) & crespo$indice <= (cres*3),]
  crC$tipo <- paste0("3C")
  #crespo <- rbind(crespo2,crC)
  data2 <- rbind(data2, crC)}
}##calculo para separar o cabelo na categoria "crespo (4), C"
spl <- split(data2, data2$id) #separando por id, o dataframe onde foi
calculado as variações de ondulação do cabelo
len4 <- length(spl) #tamanho da lista = ao tamanho de ids
cada <- unique(data2$tipo)
if(sum(len4 == tipos) > 0)
{
  return(list(cat("Você sabe tudo sobre seu cabelo! Acertou a quantidade de
diferentes tipos de curvatura seu cabelo apresenta.", "tipos de cabelo:",
cada, sep="\t"),data2))
}
if(len4>tipos)
{return(list(cat("Você mais tipos diferentes de cabelo do que imagina!",
"tipos de cabelo:", cada, sep="\t"), data2))}
if(len4 < tipos)
{return(list(cat("Seu cabelo apresenta menos tipos do que o que você
inseriu na função.", "tipos de cabelo:", cada, sep="\t"),data2))}
}
```

From:

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

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2019:alunos:trabalho\\_final:annelandine:funcao](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2019:alunos:trabalho_final:annelandine:funcao) 

Last update: **2020/08/12 06:04**