



# Carlos Henrique Botelho

Mestrando em Economia da Saúde - USP.

Projeto de mestrado:

“Avaliação econômica para tratamento de pacientes com câncer”

## MEUS EXERCÍCIOS

[1.1\\_a\\_1.3.txt](#)

[aula\\_4\\_-\\_exercicios.docx](#)

[4.2\\_a\\_4.5.txt](#)

## PROPOSTA DE TRABALHO FINAL - 22/04/2016

### PROPOSTAS A e B

Por favor coloque as propostas diretamente no wiki para que possamos discutir aqui mesmo tendo um histórico de mudanças.

Suas 4 primeiras propostas são praticamente a mesma, e não creio que sejam coisas bacanas de ficarem em uma única função. A ideia de uma função não é ser um script completo que automatiza completamente uma análise, mas uma peça identificável da análise, que eventualmente possa ser usada em vários contextos, ou no mínimo várias vezes dentro de uma mesma análise. Testar normalidade, fazer gráficos diagnósticos, escolher um modelo, rodar o modelo, diagnosticar o modelo e comparar modelos é coisa demais pra uma função só. Sugiro que vc foque num aspecto só do processo de modelagem, e não tente automatizar tudo. Frequentemente tem muita subjetividade na escolha de um modelo, e principalmente muita dependência na pergunta que vc está interessada em responder. É muito difícil automatizar isso.

A última é um péssimo jeito de montar um modelo linear. Vc deveria incluir variáveis que façam sentido e que possam ser interpretadas afim de responder uma pergunta, e não só tentar montar o conjunto que tem maior  $R^2$ .

Acho que vale a pena pensar mais em uma peça repetível e interessante das análises para transformar em função.

Valeu,

—Ogro

Veja as indicações sobre como postar propostas e o trabalho no wiki. Lembre-se que temos cerca de 70 alunos para avaliar e se não seguirem as recomendações fica difícil de operacionalizar o curso. Veja se consegue ir pelo caminho que o Diogo apontou. Se não vislumbrar, sugiro que solicite ajuda no forum e se possível vá conversar pessoalmente com algum monitor disponível. Não demore a fazê-lo pois o tempo para a execução fica curto. — [Alexandre Adalardo de Oliveira](#) 2016/04/29 11:04

Boa tarde, prof. Alexandre e Diogo,

Obrigado pelo retorno sobre o trabalho final.

Tentei reformular, segue abaixo.

## PROPOSTA DE TRABALHO FINAL - 29/04/2016

[PROPOSTAS - A - REFORMULADA e C](#)

## FUNÇÃO MEDIDAS ANTROPOMÉTRICAS

### Página de ajuda

antropometricas  
Documentation

package:unknown

R

### Função Medidas Antropométricas

#### Descrição:

Calcula o índice de massa corporal(imc) de adultos e idosos, classifica o índice de massa corporal(imc) de adultos e idosos e classifica circunferência abdominal (ca) de adultos.

#### Uso:

`antropometricas(x)`

#### Argumentos:

`x`= data frame contendo as colunas com as variáveis nomeadas e na respectiva ordem: nome, sexo, idade, altura, peso e ca.

#### Detalhes:

Os parâmetros de classificação do imc e circunferência abdominal(ca) seguem

as normas da Organização Mundial da Saúde (OMS).

sexo foi classificado como F (feminino) e M (masculino)

idade medida em anos

altura medida em metro

peso medido em kg

circunferência abdominal(ca) medida em cm

imc= medido em kg/m<sup>2</sup>

Classificação do imc:

Faixa etária de 19 a 65 anos:

imc < 18,5 (Baixo Peso)

18,5 =< imc < 25 (Peso Normal)

25 =< imc < 30 (Sobrepeso)

30 =< imc < 35 (Obesidade classe 1)

35 =< imc < 40 (Obesidade classe 2)

imc >= 40 (Obesidade classe 3)

Faixa etária acima de 65 anos:

imc < 22 (Baixo Peso)

22 =< imc =< 27 (Peso Normal)

imc > 27 (Excesso de Peso)

Classificação da circunferência abdominal(ca):

Faixa etária de 19 a 65 anos:

Masculino:

Aumentada >= 94 cm e < 102 cm

Substancialmente aumentada >= 102 cm

Feminino:

Aumentada >= 80 cm e < 88 cm

Substancialmente aumentada >= 88

Valores:

Número de NAs excluídos

Número de pacientes por faixa etária (menores de 10 anos, entre 10 e 19 anos, entre 19 e 65 anos e maiores de 65 anos).

Número de pacientes no data frame gerado, após eliminação dos NAs e classificação do imc e circunferência adominal(ca).

Número de pacientes fora dos parâmetros do imc.

Número de pacientes fora dos parâmetros da circunferência abdominal.

Retorna um data frame contendo as colunas do data frame original mais as colunas de classificação e imc com as variáveis nomeadas e na respectiva ordem: nome, sexo, idade, altura, peso, circunferencia abdominal, classificacao etaria, imc, classificacao imc e classificacao circunferencia abdominal.

Exporta um arquivo com data para o diretório do usuário nomeado da seguinte forma: dados.exportados.(ano/mes/dia).csv

#### Avisos:

O data frame de entrada deve ter as variáveis de cada coluna nomeadas e na respectiva ordem mencionada. Veja o exemplo que segue abaixo!

Esta função não classifica circunferência abdominal de idosos, pois este não é um parâmetro adotado para vigilância nutricional, segundo recomendações da OMS e Ministério da Saúde.

Esta função não calcula e classifica o imc e circunferencia abdominal de crianças e jovens, pois estes possuem outros parâmetros utilizados no cálculo e classificação.

Caso o data frame contenha algum dos casos será classificado como "Nao se aplica" e apresentado no console do R a mensagem de aviso.

#### Autor:

Carlos Henrique Botelho  
botelho13@usp.br

#### References:

<http://www2.einstein.br/Blog/paginas/post.aspx?post=1260>  
[http://www.abeso.org.br/pdf/diretrizes\\_brasileiras\\_obesidade\\_2009\\_2010\\_1.pdf](http://www.abeso.org.br/pdf/diretrizes_brasileiras_obesidade_2009_2010_1.pdf)  
[http://189.28.128.100/dab/docs/portaldab/publicacoes/orientacoes\\_coleta\\_analise\\_dados\\_antropometricos.pdf](http://189.28.128.100/dab/docs/portaldab/publicacoes/orientacoes_coleta_analise_dados_antropometricos.pdf)

#### Exemplos:

nome	sexo	idade	altura	peso	ca
Carla	F	39	1.6	65.5	75.45
Joana	F	45	1.7	50	70

Karina	F	35	1.7	120.4	100
Manoela	F	46	1.65	95	86.25
Maria	F	43	1.75	85.34	78
Rita	F	37	NA	90.345	90
Marília	NA	50	1.63	98.35	92.35
Joaquim	M	45	1.81	80	NA
NA	M	NA	1.75	100.25	100
Pedro	M	60	1.73	NA	110.23

Obsevação: Junto com a função será disponibilizado um arquivo que foi criado para testar a função.

## Código da função

```
antropometricas <- function(x) {
  pacientes <- na.omit(x) # RETIRA TODOS OS PACIENTES QUE TIVEREM ALGUM NA
  NOS DADOS!

  # Mostra na tela os avisos:
  n.NA=length(x$idade)-length(pacientes$idade) # Mostra na tela quantos
  NAs foram excluidos do data frame.
  if(n.NA != 0) # Se algum NA foi excluido
  será exibido no console do R a mensagem do números de valores excluídos.
  {cat("\t", n.NA," valores NA excluídos\n")} # A funcao cat mostra no
  console a mensagem criada de aviso.
  # Arredondar de variaveis com a funcao round

  pacientes$idade <- round( pacientes$idade, 0 ) # Idade (anos) foi
  arredondada sem casas decimais.
  pacientes$altura <- round( pacientes$altura, 2 ) # Altura (m) foi
  arredondada para duas casas decimais.
  pacientes$ca <- round( pacientes$ca, 2 ) # ca = circunferencia
  abdominal (cm) foi arredondada para duas casas decimais.
  pacientes$peso <- round( pacientes$peso, 2 ) # peso (kg) # foi
  arredondado para duas casas decimais.
  # Classificacao etaria
  # criando o vetor class.idade, para depois classificar as idades.

  pacientes$class.idade <- ifelse( pacientes$idade < 10, "Crianca", #
  ifelse foi usado para classificar as faixas etarias, dentro das faixas
  estabelecidas.
  ifelse( pacientes$idade < 19, "Jovem",
  ifelse( pacientes$idade <= 65, "Adulto",
  "Idoso")))
  # Calcular IMC - arredondado para duas casas decimais

  pacientes$imc <- round( pacientes$peso/( pacientes$altura )^2, 2) #
  criando o vetor imc a partir da formula imc=peso/m² e arredondando para duas
  casas decimais.
  # Classificacao de IMC conforme OMS (1997)
```

```
pacientes$class.imc <- rep( "Nao se aplica", nrow(pacientes) ) #
criando o vetor class.imc com o tamanho dos demais vetores para depois
classificar a partir dos valores do imc.
# classificando a coluna class.imc com o uso do which para identificar
as linhas que possuem a condicao necessaria para "Adultos".

pacientes[which( pacientes$imc < 18.5 &  pacientes$class.idade ==
"Adulto"),"class.imc" ] <- "Baixo Peso"
pacientes[which( pacientes$imc >= 18.5 &  pacientes$imc< 25 &
pacientes$class.idade == "Adulto"),"class.imc" ] <- "Peso Normal"
pacientes[which(pacientes$imc >= 25 & pacientes$imc < 30 &
pacientes$class.idade == "Adulto"),"class.imc" ] <- "Sobrepeso"
pacientes[which(pacientes$imc >= 30 & pacientes$imc < 35 &
pacientes$class.idade == "Adulto"),"class.imc" ] <- "Obesidade classe 1"
pacientes[which(pacientes$imc >= 35 & pacientes$imc < 40 &
pacientes$class.idade == "Adulto"),"class.imc" ] <- "Obesidade classe 2"
pacientes[which(pacientes$imc >= 40 & pacientes$class.idade ==
"Adulto"),"class.imc" ] <- "Obesidade classe 3"
# classificando a coluna class.imc com o uso do which para identificar
as linhas que possuem a condicao necessaria para "Idosos".

pacientes[which(pacientes$imc < 22 & pacientes$class.idade ==
"Idoso"),"class.imc"] <- "Baixo Peso"
pacientes[which(pacientes$imc >= 22 & pacientes$imc< 27 &
pacientes$class.idade == "Idoso"),"class.imc"] <- "Peso Normal"
pacientes[which(pacientes$imc >= 27 & pacientes$class.idade ==
"Idoso"),"class.imc"] <- "Excesso de Peso"

# Classificacao da ca= circunferencia abdominal conforme Diretrizes
Brasileiras de Obesidade (2010) - Os parametros sao diferentes para sexo
masculino (M) e feminino (F).

# Se for do sexo masculino (M) classificar como: ca < 94 como "Normal", ca
>= 94 e < 102 como "Aumentada" e ca >= 102 como "Aumentada
substancialmente".
pacientes$class.ca <- rep( "Nao se aplica", nrow(pacientes) ) # criando
o vetor class.ca com o tamanho dos demais vetores para para depois
classificar a partir dos valores de ca.
# classificando a coluna class.ca com o uso do which para identificar as
linhas que possuem a condicao necessaria para o sexo masculino (M).
pacientes[which(pacientes$ca < 94 & pacientes$sexo == "M" &
pacientes$class.idade == "Adulto" ), "class.ca"] <- "Normal"
pacientes[which(pacientes$ca >= 94 & pacientes$ca < 102 &
pacientes$sexo == "M" & pacientes$class.idade == "Adulto" ), "class.ca"] <-
"Aumentada"
pacientes[which(pacientes$ca >= 102 & pacientes$sexo == "M" &
pacientes$class.idade == "Adulto" ), "class.ca"] <- "Aumentada
substancialmente"
# classificando a coluna class.ca com o uso do which para identificar as
linhas que possuem a condicao necessaria para o sexo feminino (F).
```

```
# Se do sexo feminino (F) classificar como: ca < 80 como "Normal", ca >=
80 e < 88 como "Aumentada" e ca >= 88 como "Aumentada substancialmente".
pacientes[which(pacientes$ca < 80 & pacientes$sexo == "F" &
pacientes$class.idade == "Adulto" ), "class.ca"] <- "Normal"
pacientes[which(pacientes$ca >= 80 & pacientes$ca < 88 & pacientes$sexo
== "F" & pacientes$class.idade == "Adulto" ), "class.ca"] <- "Aumentada"
pacientes[which(pacientes$ca >= 88 & pacientes$sexo == "F" &
pacientes$class.idade == "Adulto" ), "class.ca"] <- "Aumentada
substancialmente"
# Contar o numero de pacientes por faixa etaria
menor10 <- sum( pacientes$class.idade == "Crianca" )
entrel0e19 <- sum( pacientes$class.idade == "Jovem" )
entre19e65 <- sum( pacientes$class.idade == "Adulto" )
maior65 <- sum( pacientes$class.idade == "Idoso" )
# Mostra na tela os avisos:

# Foi usado o paste para agrupar strings e criar o aviso no console do R
com o número de pacientes em cada faixa etária.
MENSAGEM_menor10 = paste("Ha", menor10 , "pacientes menores de 10 anos.
Para estes pacientes esta funcao nao se aplica.\n")
MENSAGEM_entrel0e19 = paste("Ha", entrel0e19, "pacientes entre 10 e 19
anos.
Para estes pacientes esta funcao nao se aplica.\n")
MENSAGEM_entre19e65 = paste("Ha", entre19e65, "pacientes entre 19 e 65
anos.
Para estes pacientes utiliza-se a classificao da circunferencia
abdominal e imc.\n")
MENSAGEM_maior65 = paste("Ha", maior65, "pacientes maiores de 65 anos.
Para estes pacientes a classificao da circunferencia abdominal nao se
aplica.\n")

# If foi usado para mostrar no console do R somente se o valor da soma
da faixa etaria for diferente de zero.

if(menor10 != 0)
{cat("\t", MENSAGEM_menor10)} # cat para mostrar na tela do R a
mensagem.
if(entrel0e19 != 0)
{cat("\t", MENSAGEM_entrel0e19)}
if(entre19e65 != 0)
{cat("\t", MENSAGEM_entre19e65)}
if(maior65 != 0)
{cat("\t", MENSAGEM_maior65)}

# Transformando em fator, para contar a frequencia de cada classificao
do imc e ca.
pacientes$class.imc <- as.factor( pacientes$class.imc )
pacientes$class.ca <- as.factor( pacientes$class.ca )
```

```
total <- length(pacientes$class.imc) # contando o número total de
pacientes que ha no data frame depois da exclusao dos NAs.

# somando o numero de pacientes que estao fora dos parametros de imc, ou
seja, todos aqueles que nao sao classificados como "Peso Normal".

risco.imc <- sum( pacientes$class.imc == "Baixo Peso" |
pacientes$class.imc == "Sobrepeso" | pacientes$class.imc == "Obesidade
classe 1" | pacientes$class.imc == "Obesidade classe 2" | pacientes$class.imc
== "Obesidade classe 3" | pacientes$class.imc == "Excesso de peso")
# somando o numero de pacientes que estao fora dos parametros de ca, ou
seja, todos aqueles que nao sao classificados como "Normal".
risco.ca <- sum( pacientes$class.ca == "Aumentada" | pacientes$class.ca
== "Aumentada substancialmente")
# Mostra na tela os avisos:
MENSAGEM_total = paste("Ha", total , "pacientes no data frame
gerado.\n") # Foi usado a funcao paste para juntar strings.
cat("\t", MENSAGEM_total)
# e depois cat para apresentar o aviso no console do R.
MENSAGEM_risco.imc = paste("Ha", risco.imc , "pacientes fora dos
parametros de imc.\n")
cat("\t", MENSAGEM_risco.imc)
MENSAGEM_risco.ca = paste("Ha", risco.ca , "pacientes fora dos
parametros de circunferencia abdominal.\n")
cat("\t", MENSAGEM_risco.ca)

# Renomeia as colunas do dataframe com os nomes por extenso.

names( pacientes ) <- c(
"nome","sexo","idade","altura","peso","circunferencia
abdominal","classificacao etaria","imc","classificacao imc","classificacao
circunferencia abdominal")
# Referencia para formato de data e hora em
https://stat.ethz.ch/R-manual/R-devel/library/base/html/strptime.html
# Usei somente ano, mes e dia, assim o usuario tem seus arquivos
organizados.
arquivo <- paste( "dados.exportados.", format( Sys.time(), "%Y%m%d" ),
".csv", sep="" )

# A ideia e ter os dados organizados de imc e ca a cada intervalo (mes,
ciclo de medicacao...) e depois poder fazer alguma analise com os dados.
# Exportando o data frame gerado para um arquivo .csv que será salvo no
diretorio que o usuario esta utilizando.
dados.exportados <- write.table( x = pacientes, file = arquivo,
col.names = T, row.names = F, sep = "," ) # write.table para salvar data
frame como arquivo.
return (pacientes) # retorna o data frame com as colunas do data frame
original mais as novas colunas de classificacao etaria, imc,
classificacao de imc e classificacao da circunferencia abdmoninal.
}
```



## Arquivo da função

[antropometricas.r](#)

[help.antropometricas.txt](#)

## Arquivo para testar a função

[teste.antropometricas.csv](#)

From:

<http://labtrop.ib.usp.br/> - Laboratório de Ecologia de Florestas Tropicais

Permanent link:

[http://labtrop.ib.usp.br/doku.php?id=cursos:ecor:05\\_curso\\_antigo:r2016:alunos:trabalho\\_final:botelho13:start](http://labtrop.ib.usp.br/doku.php?id=cursos:ecor:05_curso_antigo:r2016:alunos:trabalho_final:botelho13:start) 

Last update: **2020/07/27 18:47**