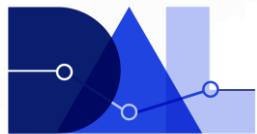




CEFET/RJ



# Apply



Eduardo Ogasawara

eduardo.ogasawara@cefet-rj.br

<https://eic.cefet-rj.br/~eogasawara>

## *Implicitly Loops – sapply, lapply*

- lapply, sapply executa uma função para cada coluna
  - O primeiro caractere define o tipo de retorno
    - l – lista, s – simples (vetor ou matriz)
  - O segundo parâmetro é a função a ser invocada
  - Os seguintes parâmetros são passados para a função invocada
- apply é a função genérica
  - O segundo parâmetro define:
    - se chama a função para cada linha (1) ou cada coluna (2)

## Exemplo

```
weight <- c(60, 72, 57, 90, 95, 72)
height <- c(1.75, 1.80, 1.65, 1.90, 1.74, 1.91)
subject <- c("A", "B", "C", "D", "E", "F")
```



## *Criando uma tabela básica*

```
d <- data.frame(weight=weight, height=height, subject=subject)
head(d)
```



```
##   weight height subject
## 1     60   1.75      A
## 2     72   1.80      B
## 3     57   1.65      C
## 4     90   1.90      D
## 5     95   1.74      E
## 6     72   1.91      F
```



## Implicitly Loops – *sapply*, *lapply*

```
lapply(d[,1:2], mean, na.rm=TRUE)
```



```
## $weight  
## [1] 74.33333  
##  
## $height  
## [1] 1.791667
```



```
sapply(d[,1:2], mean, na.rm=TRUE)
```



```
##   weight   height  
## 74.333333 1.791667
```



## Implicitly Loops – *sapply, lapply*

```
m <- as.matrix(d[,1:2])
```



```
apply(m, 1, min, na.rm=TRUE)
```



```
## [1] 1.75 1.80 1.65 1.90 1.74 1.91
```



```
apply(m, 2, min, na.rm=TRUE)
```

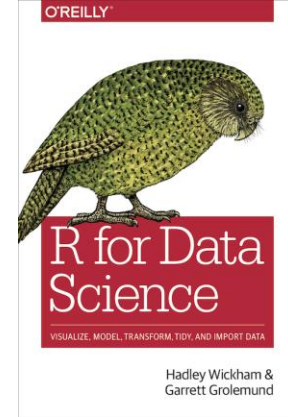
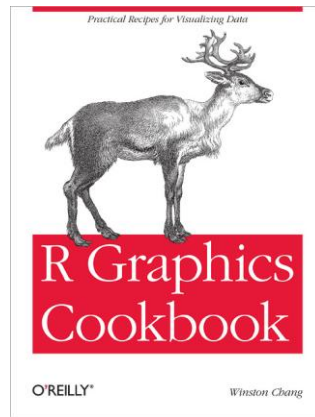
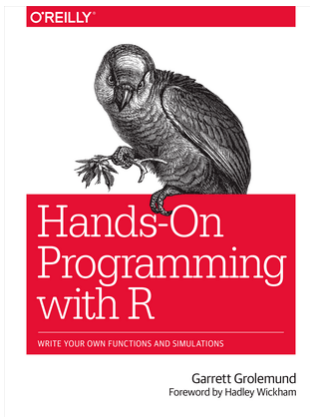


```
## weight height  
## 57.00 1.65
```



# Referências

Material: <https://eic.cefet-rj.br/~eogasawara/tutorial-r>



Hands-on Programming with R: <https://rstudio-education.github.io/hopr/index.html>

R Graphics Cookbook: <https://r-graphics.org>

R Packages: <https://r-pkgs.org/index.html>

R for Data Science: <https://r4ds.had.co.nz>

<https://rstudio-education.github.io/hopr/basics.html>