



DEPARTAMENTO DE ESTATÍSTICA

09 junho 2023

## **Lista 2**

Prof. Dr. Donald Matthew Pianto

Aluno: Bruno Gondim Toledo

Matrícula: 15/0167636

Estatística Computacional

1º/2023

```

if (!require("pacman")) install.packages("pacman")

## Carregando pacotes exigidos: pacman
p_load(knitr,tidyverse,doParallel,furrr,tictoc,ggpubr,beep)
cores <- detectCores()
dados <- read_rds("dados/dados.rds")

```

## Questão 1

a)

```

# "Bootstrap"?

# "Analítica"
m1 <- dados %>%
  filter(Rota == "Disappointment Cleaver") %>%
  summarise(lambda = mean(Sucessos)) %>%
  pull()

m2 <- dados %>%
  filter(Rota != "Disappointment Cleaver") %>%
  summarise(lambda = mean(Sucessos)) %>%
  pull()

dif_a <- m1-m2

dados %>%
  filter(Rota == "Disappointment Cleaver") %>%
  tally()

## # A tibble: 1 x 1
##       n
##   <int>
## 1   178

# Simulação:
rotas <- dados %>%
  filter(Rota != "Disappointment Cleaver") %>%
  select(Sucessos)

DC <- dados %>%
  filter(Rota == "Disappointment Cleaver") %>%
  select(Sucessos)

N <- 10000
dif <- numeric(N)
for (i in 1:N){
  media1 <- sample(DC$Sucessos,size=10,replace=T)
  media2 <- sample(rotas$Sucessos,size=10,replace=T)

  media1 <- mean(media1)
  media2 <- mean(media2)

  dif[i] <- abs(media1 - media2)
}
beep(2)
# Média:

```

```
mean(dif)

## [1] 16.15016
# Probabilidade das médias serem iguais:
mean(dif)/N

## [1] 0.001615016
```

b)

```
# B)

lv <- function(papb,ti,ns){
  l <- exp(papb[1]+papb[2]*ti)
  sum(dpois(ns,l,log=T))
}

parametros <- optim(c(0,0),
                    function(.) -lv(., ti=dados$Temperatura, ns=dados$Sucessos))$`par`

# Estimadores de máxima verossimilhança de alpha e beta; respectivamente:
parametros

## [1] 1.98850522 0.08127694
```

c)

```
# c) ti=15 ~ poisson (l=)

lv <- function(papb,ti,ns){
  l <- exp(papb[1]+papb[2]*ti)
  sum(dpois(ns,l,log=T))
}

parametros <- optim(c(0,0),
                    function(.) -lv(., ti=15, ns=dados$Sucessos))$`par`

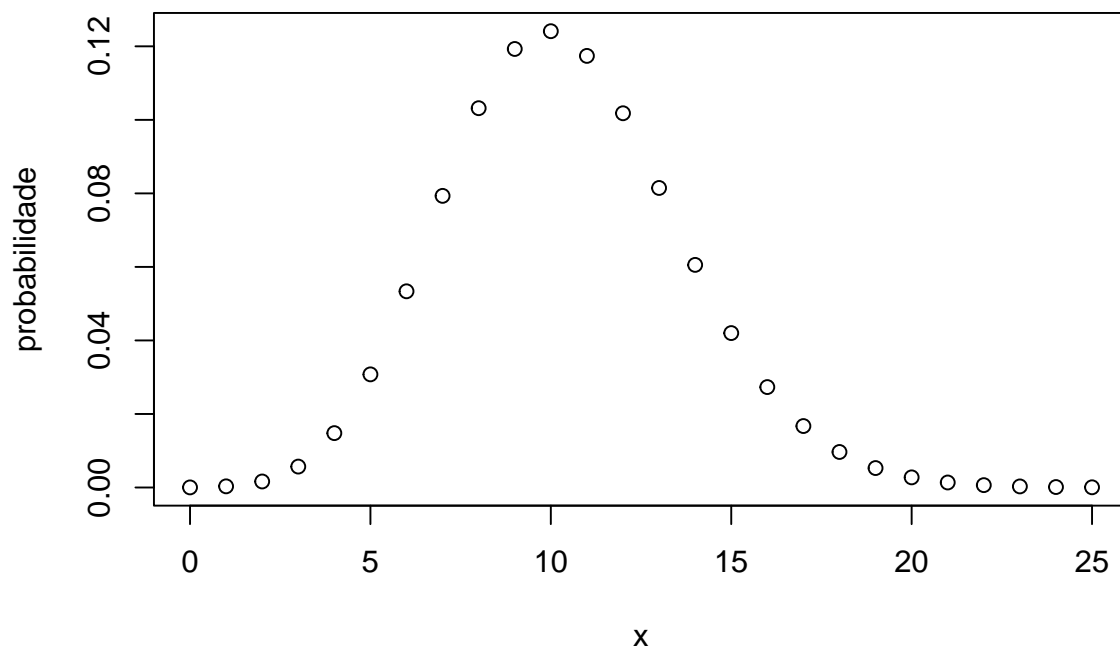
# Estimadores de máxima verossimilhança de alpha e beta; respectivamente:
parametros

## [1] 0.1370870 0.1470131

lambda <- exp(parametros[1]+parametros[2]*15) # Parâmetro lambda

# estimativa da distribuição de probabilidade para os valores de x de 0 a 25
x <- 0:25
probabilidade <- dpois(x, lambda)

plot(x, probabilidade)
```



d)

e)

2)

```
# Definindo a função que verifica se um ponto está dentro da elipsoide
pontos_dentro <- function(x, y, z) {
  (x^2/2) + (y^2/3) + (z^2/4) <= 1
}

# definindo n° de pontos
N <- 10000000

# gerando os valores de x,y e z no limite da elipsoide
x <- runif(N, min = -1, max = 1)
y <- runif(N, min = -1, max = 1)
z <- runif(N, min = -1, max = 1)

# Contando quantos pontos estão dentro da elipsoide
dentro <- sum(pontos_dentro(x, y, z))

# Calculando a proporção de pontos dentro da elipsoide em relação ao total gerado
prop <- dentro / N

volume_elipsoide <- prop * 8 # Volume do cubo que circunscreve a elipsoide (2 * 2 * 2)
volume_elipsoide

## [1] 7.997513
```