



DEPARTAMENTO DE ESTATÍSTICA

16 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,beepr)
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)
}
# Média:
mean(dif)

```

```
## [1] 16.4387
```

```
# Probabilidade das médias serem iguais:  
mean(dif)/N
```

```
## [1] 0.00164387
```

b)

```
# B)
```

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

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

```
# Estimadores de máxima verossimilhança de alpha e beta; respectivamente:  
parametross
```

```
## [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
```

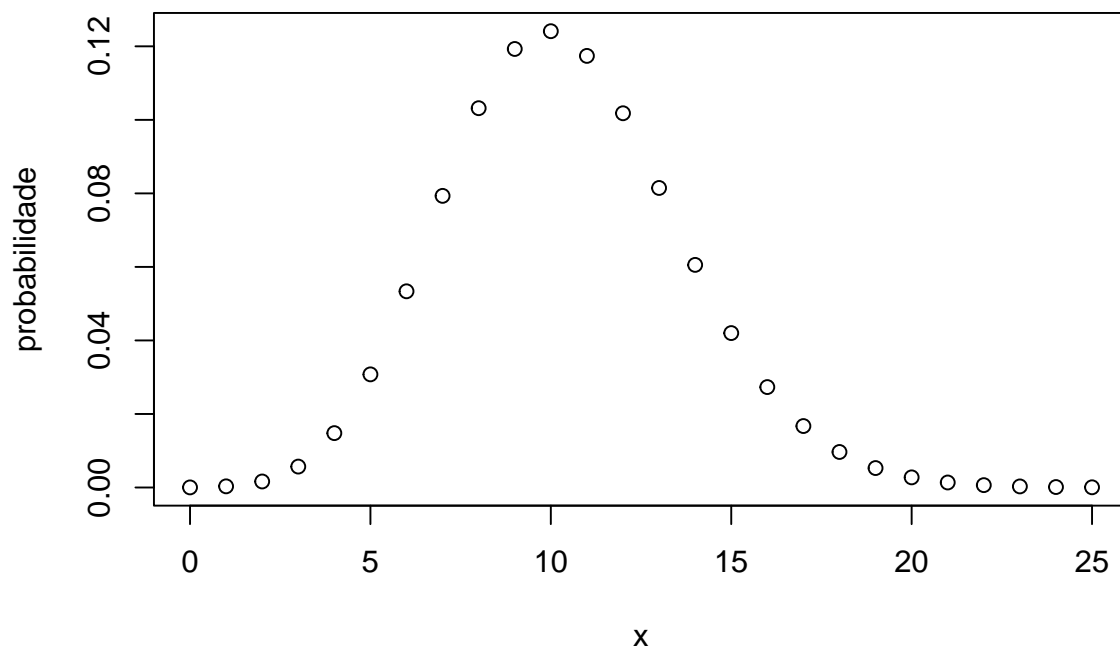
```
lambda <- exp(parametros[1]+parametros[2]*15) # Parâmetro lambda
```

```
# 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)

```
sucessos <- numeric()
beta_estim <- numeric()
MSE <- numeric()

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

tic()
for(I in 1:1000){
  for(i in 1:nrow(dados)){
    sucessos[i] <- rpois(n=1,lambda=exp(parametross[1]+parametross[2]*dados$Temperatura[i]))}

  Parametros <- optim(c(0,0),
    function(.) -lv(., ti=dados$Temperatura, ns=sucessos))$`par`
  beta_estim[I] <- Parametros[2]
  MSE[I] <- mean((sucessos-exp(parametross[1]+parametross[2]*dados$Temperatura))^2)

  sucessos <- numeric()
}
toc()
```

```
## 12.09 sec elapsed
```

```
X <- exp(beta_estim)
```

```
summary(X)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
## 1.074 1.083 1.085 1.085 1.087 1.095
```

```
# Intervalo 95% conf.
```

```
quantile(X, probs= (0.025))
```

```
## 2.5%
```

```
## 1.078898
```

```
quantile(X, probs= (0.975))
```

```
## 97.5%
```

```
## 1.090442
```

e) A magnitude do Erro quadrático médio é pequeno, o que indica que neste caso, obter o estimador iterativamente não fez tanta diferença em relação ao calculado de máxima verossimilhança.

```
mean(MSE)
```

```
## [1] 10.3933
```

2)

```
pontos_dentro <- function(x, y, z) {  
  (x^2/2) + (y^2/3) + (z^2/4) <= 1  
}
```

```
N <- 10000000
```

```
x <- runif(N, min = -1, max = 1)
```

```
y <- runif(N, min = -1, max = 1)
```

```
z <- runif(N, min = -1, max = 1)
```

```
dentro <- sum(pontos_dentro(x, y, z))
```

```
prop <- dentro / N
```

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

```
## [1] 7.997542
```