

*Bootstrap*  
(correção de viés e intervalos de confiança)

2023

## Concordância

Apresentamos mais resultados sobre o **coeficiente de correlação de concordância**, complementando o exemplo do item “13. Erro padrão *bootstrap*.”.

## Concordância

Apresentamos mais resultados sobre o **coeficiente de correlação de concordância**, complementando o exemplo do item “13. Erro padrão *bootstrap*.”.

Aplicando o método *bootstrap*, obtemos uma estimativa do coeficiente de correlação de concordância ( $\rho_c$ ) com correção de viés e também intervalos de confiança para  $\rho_c$ .

## Solução

```
## Bootstrap iterado
rca <- rc(x, y) # Estimativa
set.seed(5150)
B <- 5000 # Número de amostras (nível 1)
B2 <- 500 # Número de amostras (nível 2)
te <- c() # tb*
system.time(
  # Nível 1
  rce <- c()
  for (b in 1:B) {
    # Índices da amostra bootstrap
    indb <- sample(n, n, replace = TRUE)
    # Estimativa (*)
    rce[b] <- rc(x, y, indb)
    # Nível 2
    rcee <- c()
    for (j in 1:B2) {
      # Estimativa (**)
      rcee[j] <- rc(x[indb], y[indb], sample(indb, n, replace = TRUE))
    }
    te[b] <- (rce[b] - rca) / sd(rcee)
  }
)
```

## Solução

Com a função `system.time` obtemos o tempo de processamento (em s) da parte iterada, que é cerca de 3 min (em um computador com processador Intel Quad-Core Core i7 de 2,2 GHz, memória RAM de 16 GB e sistema operacional macOS 12.6.5).

```
user  system elapsed
175.833  0.894 176.870
```

## Solução

```
# Distribuição de t*  
hist(te, freq = FALSE, main = "", xlab = expression(t^"*"),  
      ylab = "Densidade", col = "lightgreen")  
lines(density(te), col = "blue", lty = 2, lwd = 2)  
box()
```

## Solução

```
# Distribuição de t*  
hist(te, freq = FALSE, main = "", xlab = expression(t~"*"),  
      ylab = "Densidade", col = "lightgreen")  
lines(density(te), col = "blue", lty = 2, lwd = 2)  
box()
```

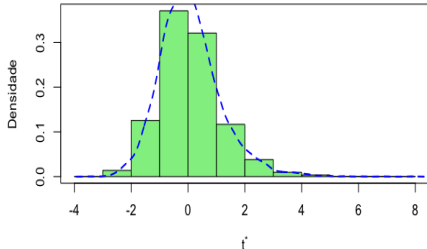


Figura 1: Distribuição de  $t^*$ .

## Solução

```
## Resultados
epe <- sd(rce) # Erro padrão bootstrap
rcae <- 2 * rca - mean(rce) # Correção de viés
cat("\n B:", B, "\n e.p. bootstrap:", epe, "\n Estimativa:", rca,
    "\n Estimativa bootstrap (sem correção):", mean(rce),
    "\n Estimativa bootstrap (com correção):", rcae)
```



## Solução

```
## Resultados
epe <- sd(rce) # Erro padrão bootstrap
rcae <- 2 * rca - mean(rce) # Correção de viés
cat("\n B:", B, "\n e.p. bootstrap:", epe, "\n Estimativa:", rca,
    "\n Estimativa bootstrap (sem correção):", mean(rce),
    "\n Estimativa bootstrap (com correção):", rcae)
```

B: 5000

e.p. bootstrap: 0.0431247

Estimativa: 0.8098709

Estimativa bootstrap (sem correção): 0.8046007

Estimativa bootstrap (com correção): 0.8151411

## Solução

```
# Intervalos de confiança
conf <- 0.95
problu <- c(1 - conf, 1 + conf) / 2
cat("\n Intervalo percentil de", 100 * conf, "%: [",
    quantile(rce, probs = problu, type = 6), ""])
```

## Solução

```
# Intervalos de confiança
conf <- 0.95
problu <- c(1 - conf, 1 + conf) / 2
cat("\n Intervalo percentil de", 100 * conf, "%: [",
    quantile(rce, probs = problu, type = 6), ""])
```

Intervalo percentil de 95 %: [ 0.7105734 0.8778015 ]

## Solução

```
# Intervalos de confiança
conf <- 0.95
problu <- c(1 - conf, 1 + conf) / 2
cat("\n Intervalo percentil de", 100 * conf, "%: [",
    quantile(rce, probs = problu, type = 6), "])")
```

Intervalo percentil de 95 %: [ 0.7105734 0.8778015 ]

```
q12 <- quantile(te, probs = problu, type = 6)
ictb <- rca - q12[2:1] * epe
cat("\n B2:", B2, "\n Intervalo t bootstrap de", 100 * conf,
    "%: [", ictb, "])")
```

## Solução

```
# Intervalos de confiança
```

```
conf <- 0.95
```

```
problu <- c(1 - conf, 1 + conf) / 2
```

```
cat("\n Intervalo percentil de", 100 * conf, "%: [",  
    quantile(rce, probs = problu, type = 6), "])")
```

```
Intervalo percentil de 95 %: [ 0.7105734 0.8778015 ]
```

```
q12 <- quantile(te, probs = problu, type = 6)
```

```
ictb <- rca - q12[2:1] * epe
```

```
cat("\n B2:", B2, "\n Intervalo t bootstrap de", 100 * conf,  
    "%: [", ictb, "])")
```

```
B2: 500
```

```
Intervalo t bootstrap de 95 %: [ 0.6975508 0.8857594 ]
```

## Solução

```
# Distribuição e intervalo t bootstrap
par(mai = c(1.2, 1.2, 0.1, 0.3))
hist(rce, freq = FALSE, main = "",
     xlab = expression(r[c]^"*"),
     ylab = "Densidade", cex.axis = 1.5, cex.lab = 1.5)
lines(density(rce), col = "blue", lwd = 2)
arrows(ictb[1], 0, ictb[2], 0, col = "red", code = 3,
      angle = 90, lwd = 2, length = 0.1)
points(rcae, 0, pch = 16, col = "green", cex = 1.5)
box()
```

## Solução

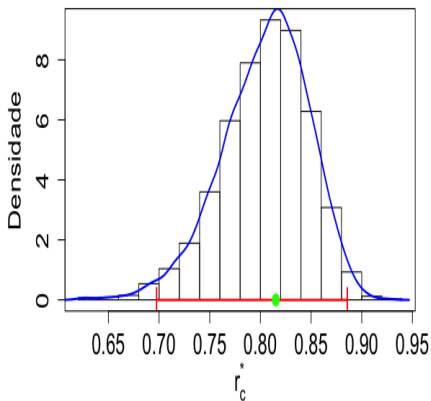


Figura 2: Histograma e densidade estimada de  $r_c^*$ , estimativa corrigida e intervalo de confiança *t bootstrap* de 95%.

- 1 Escreva um código sem utilizar laços “for” (folha 3).



- 1 Escreva um código sem utilizar laços “for” (folha 3). Compare as diferentes soluções em termos de tempo de processamento.

- 1 Escreva um código sem utilizar laços “for” (folha 3). Compare as diferentes soluções em termos de tempo de processamento.
- 2 Vários pacotes em R implementam computação de alto desempenho, como pode ser visto na página “CRAN Task View: High-Performance and Parallel Computing with R” (<https://cran.r-project.org/web/views/HighPerformanceComputing.html>).