

Escore de Fisher

Distribuição gama

2026

O método do escore de Fisher é aplicado à distribuição gama com parâmetro de forma $a > 0$ e parâmetro de taxa $b > 0$. Sua função densidade é

$$f(x; \boldsymbol{\theta}) = \begin{cases} \frac{b^a}{\Gamma(a)} x^{a-1} e^{-bx}, & \text{se } x > 0, \\ 0, & \text{se } x \leq 0, \end{cases} \quad (1)$$

em que $\boldsymbol{\theta} = (a, b)^\top$.

Em uma amostra aleatória de tamanho n , a função verossimilhança é dada por

$$L(\boldsymbol{\theta}) = \prod_{i=1}^n f(x_i; \boldsymbol{\theta}) = \left(\frac{b^a}{\Gamma(a)} \right)^n \prod_{i=1}^n x_i^{a-1} \exp \left(-b \sum_{i=1}^n x_i \right), \quad (2)$$

ao passo que a função log-verossimilhança é dada por

$$\ell(\boldsymbol{\theta}) = na \log(b) - n \log(\Gamma(a)) + (a-1)T_1 - bT_2, \quad (3)$$

em que $T_1 = \sum_{i=1}^n \log(x_i)$ e $T_2 = \sum_{i=1}^n x_i$.

Nota 1. T_1 e T_2 são estatísticas suficientes para $\boldsymbol{\theta}$.

```
# Separador decimal: ","  
options(OutDec = ",")
```

```
# Função log-verossimilhança  
logver <- function(teta) {  
  a <- teta[1]  
  b <- teta[2]  
  return(n * a * log(b) - n * lgamma(a) + (a - 1) * T1 - b * T2)  
}
```

Nota 2. Reescreva a função log-verossimilhança utilizando a função `dgamma`. Verifique a parametrização adotada na linguagem R.

Em seguida são apresentadas a função escore e a inversa da matriz de informação de Fisher. As derivadas primeira e segunda da função $\log(\Gamma(\cdot))$ são chamadas função digama e função trigama, respectivamente, e estão implementadas como `digamma` e `trigamma`.

```
# Escore  
U <- function(teta) {  
  a <- teta[1]  
  b <- teta[2]  
  return(c(n * (log(b) - digamma(a)) + T1, n * a / b - T2))  
}
```

A matriz de informação de Fisher pode ser obtida a partir dos resultados da Seção 7.1 do Cap. 17 em Johnson, Kotz & Balakrishnan (1994), *Continuous univariate distributions*, volume I, 2nd ed., Wiley: Hoboken. Do livro obtemos $\mathbf{I}_F(\boldsymbol{\eta})$, que corresponde a $\boldsymbol{\eta} = (a, 1/b)^\top$. Tomando $\boldsymbol{\theta} = (a, b)^\top$, com a mudança de parametrização, a matriz passa a ser $\mathbf{J}(\boldsymbol{\theta})^\top \mathbf{I}_F(\boldsymbol{\theta}) \mathbf{J}(\boldsymbol{\theta})$, em que $\mathbf{J}(\boldsymbol{\eta}) = \partial \boldsymbol{\theta}(\boldsymbol{\eta}) / \partial \boldsymbol{\eta}$.

```
# Inversa da matriz de informação de Fisher (amostra de tamanho n)
IF1 <- function(teta) {
  a <- teta[1]
  b <- teta[2]
  det1 <- (trigamma(a) * a - 1) / b^2
  return(matrix(c(a / b^2, 1 / b, 1 / b, trigamma(a)), ncol = 2) /
    (n * det1))
}
```

O critério de convergência (ou parada) é dado por

$$\max \left(\left| \frac{\hat{a}^{(k+1)} - \hat{a}^{(k)}}{\hat{a}^{(k)}} \right|, \left| \frac{\hat{b}^{(k+1)} - \hat{b}^{(k)}}{\hat{b}^{(k)}} \right| \right) < \epsilon, \quad (4)$$

em que k é o contador de iterações e ϵ é fixado em um valor “pequeno”.

```
# Tolerância para o critério de parada
eps <- 1e-6

# Dados e estatísticas suficientes
dados <- c(49.8, 64.3, 43.4, 40.1, 80.5, 76.6, 57.4,
  133.3, 45.4, 40.3, 59.8, 24.3, 58.9, 29.7, 65.2,
  107.1, 76.1, 37.2, 61.3, 54.0, 81.5, 76.4, 45.1,
  60.2, 53.9, 38.0, 58.5, 26.8, 20.2, 37.6, 45.4,
  78.4, 45.3, 40.8, 27.8, 60.7, 34.1, 39.0, 30.2,
  107.8, 61.1, 32.0, 31.0, 46.2, 22.2, 44.7, 47.0)

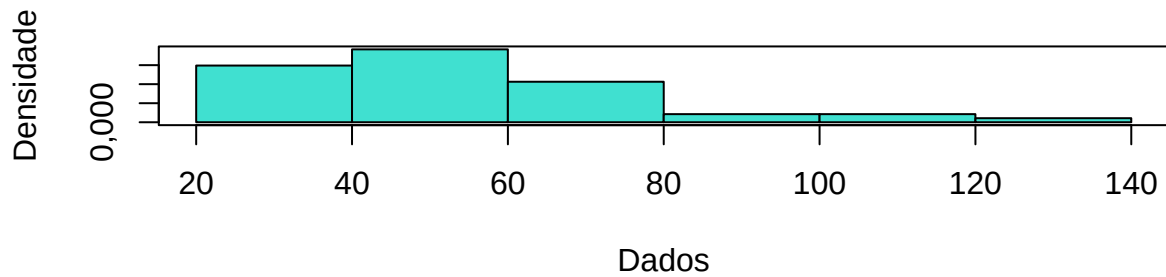
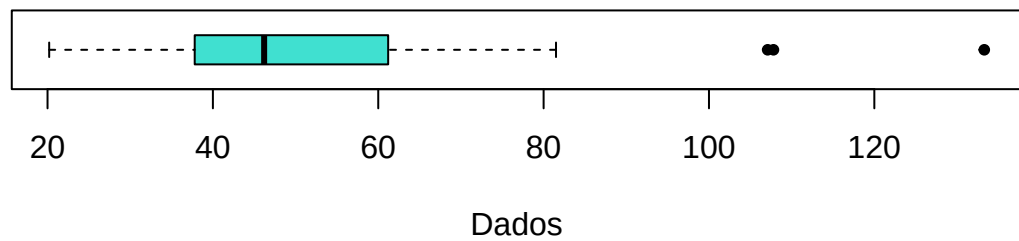
n <- length(dados)
cat("\n n =", n)

##
##  n = 47

T1 <- sum(log(dados))
T2 <- sum(dados)
```

Os gráficos abaixo indicam uma distribuição com assimetria à direita.

```
par(mfrow = c(2, 1))
hist(dados, main = "", freq = FALSE, xlab = "Dados", ylab = "Densidade",
  col = "turquoise")
box()
boxplot(dados, xlab = "Dados", horizontal = TRUE, pch = 20, col = "turquoise")
```



As estimativas iniciais são estimativas de momentos (lembrando que os estimadores do método dos momentos são consistentes).

```
# Estimativas iniciais (método dos momentos)
xb <- mean(dados)
beta0 <- xb / var(dados)
alfa0 <- xb * beta0
teta0 <- c(alfa0, beta0)

# Escore de Fisher
niter <- 0
dif <- 1 # critério de parada > epsilon
while (dif > eps) {
  niter <- niter + 1
  passo <- IF1(teta0) %*% U(teta0)
  while(logver(teta0 + passo) <= logver(teta0)) {
    passo <- 0.5 * passo
  }
  tetac <- teta0 + passo

  # Critério de parada
  dif <- max(abs(passo / teta0))
  teta0 <- tetac
}

# Resultados
cat("\n Estimativas iniciais (alfa, beta):", c(alfa0, beta0))

##
## Estimativas iniciais (alfa, beta): 5,135667 0,09668202
cat("\n Função log-verossimilhança:", logver(c(alfa0, beta0)))

##
```

```

## Função log-verossimilhança: -209,0712
cat("\n Tolerância:", eps)

##
## Tolerância: 1e-06
cat("\n Número de iterações:", niter)

##
## Número de iterações: 4
cat("\n Diferença relativa máxima:", dif)

##
## Diferença relativa máxima: 1,195081e-07
cat("\n EMV:", tetac)

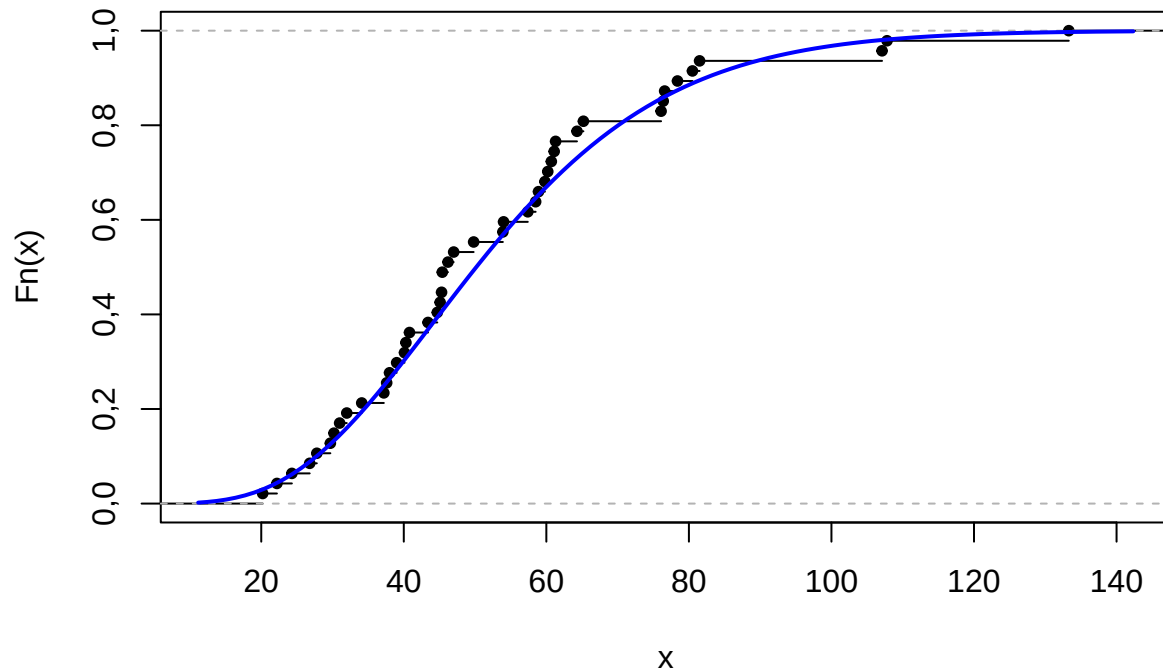
##
## EMV: 5,925721 0,1115553
cat("\n Função log-verossimilhança:", logver(tetac))

##
## Função log-verossimilhança: -208,8282
cat("\n Escore:", U(tetac))

##
## Escore: 8,526513e-14 4,547474e-13
cat("\n Erros padrão da EMV:", sqrt(diag(IF1(tetac))))

##
## Erros padrão da EMV: 1,189565 0,02337063
# Distribuição empírica e modelo ajustado
plot(ecdf(dados), main = "", pch = 20)
curve(pgamma(x, shape = tetac[1], rate = tetac[2]), col = "blue",
      add = TRUE, lwd = 2)

```



Nota 3. Pode ser afirmado que a distribuição gama faz um bom ajuste aos dados?

Nota 4. Modifique o critério de parada de modo que o algoritmo seja encerrado após um número máximo de iterações ter sido executado, mesmo que a diferença máxima `dif` ainda seja maior do que `eps`.

Nota 5. Compare com os resultados obtidos com a função `fitdistr` do pacote `MASS`.

Nota 6. Refaça o exemplo partindo de diferentes estimativas iniciais.

Nota 7. Apresente estimativas *bootstrap* dos erros padrão dos estimadores.