

**Disciplina: Estatística Computacional II****Lista 1: exercícios 5 - 10**

|                          |                                   |
|--------------------------|-----------------------------------|
| Data da lista:           | 27/08 - 10/09                     |
| Preceptor(a):            | Zamir Jose Rodriguez Ortega       |
| Curso(s)<br>atendido(s): | Estatística                       |
| Orientador(a):           | Marcos Vinicius de Oliveira Peres |

**Exercício 5.**

```
# Parâmetros
n <- 10000
c <- 2
k <- 3

# Semente para reprodutibilidade
set.seed(123)

# Gerando  $U \sim \text{Uniforme}(0,1)$ 
U <- runif(n)

# Método da inversa
X <- ((1 - U)^(-1/k) - 1)^(1/c)

# Função densidade teórica da Burr XII
f_burr <- function(x, c, k) {
  c * k * x^(c - 1) * (1 + x^c)^(-(k + 1))
}

# Histograma da amostra
hist(X,
      probability = TRUE,
      breaks = 40,
      main = "Distribuição Burr XII",
      xlab = "X",
      ylab = "Densidade",
      col = "lightgray",
      border = "white")

# Curva da densidade teórica
curve(f_burr(x, c, k),
      from = 0,
      to = max(X),
```

```
add = TRUE,  
lwd = 2)
```

### **Exercício 6.**

```
# Parâmetros  
n <- 10000  
a <- 2  
b <- 5  
  
# Semente para reprodutibilidade  
set.seed(123)  
  
# Gerando  $U \sim \text{Uniforme}(0,1)$   
U <- runif(n)  
  
# Método da inversa  
X <- (1 - (1 - U)^(1/b))^(1/a)  
  
# Função densidade teórica da Kumaraswamy  
f_kumaraswamy <- function(x, a, b) {  
  a * b * x^(a - 1) * (1 - x^a)^(b - 1)  
}  
  
# Histograma da amostra  
hist(X,  
      probability = TRUE,  
      breaks = 40,  
      main = "Distribuição Kumaraswamy",  
      xlab = "X",  
      ylab = "Densidade",  
      col = "lightgray",  
      border = "white",  
      xlim = c(0, 1))  
  
# Curva da densidade teórica  
curve(f_kumaraswamy(x, a, b),  
      from = 0,  
      to = 1,  
      add = TRUE,  
      lwd = 2)
```

### **Exercício 7.**

```
# Parâmetros  
n <- 1000  
alpha <- 4  
beta <- 1.5  
  
# Semente para reprodutibilidade  
set.seed(123)
```

```

# Gerando U ~ Uniforme(0,1)
U <- runif(n)

# Método da inversa
X <- alpha - beta * log(-log(U))

# Função densidade teórica da Gumbel
f_gumbel <- function(x, alpha,
  beta) {
  z <- (x - alpha) / beta

  (1 / beta) * exp(-z) * exp(-exp(-z))
}

# Histograma da amostra
hist(X,
  probability = TRUE,
  breaks = 30,
  main = "Distribuição de Gumbel",
  xlab = "X",
  ylab = "Densidade",
  col = "lightgray",
  border = "white")

# Curva da densidade teórica
curve(f_gumbel(x, alpha, beta),
  from = min(X),
  to = max(X),
  add = TRUE,
  lwd = 2)

```

### Exercício 8.

```

# Parâmetros
n <- 1000
theta <- 2

# Semente para reprodutibilidade
set.seed(123)

# Função CDF da distribuição de Lindley
F_lindley <- function(x, theta) {
  1 - (1 + (theta * x) / (1 + theta)) * exp(-theta * x)
}

# Função para gerar uma observação pelo método da inversa
r_lindley <- function(n, theta) {

  # Gerando U ~ Uniforme(0,1)
  U <- runif(n)

  # Vetor para armazenar as observações
  X <- numeric(n)

```

```

# Inversão numérica para cada U
for (i in 1:n) {

  X[i] <- uniroot(
    function(x) F_lindley(x, theta) - U[i],
    interval = c(0, 100)
  )$root

}

return(X)
}

# Gerando a amostra
X <- r_lindley(n, theta)

# Função densidade teórica da distribuição de Lindley
f_lindley <- function(x, theta) {
  (theta^2 / (1 + theta)) *
  (1 + x) * exp(-theta * x)
}

# Histograma da amostra
hist(X,
  probability = TRUE,
  breaks = 30,
  main = "Distribuição de Lindley",
  xlab = "X",
  ylab = "Densidade",
  col = "lightgray",
  border = "white")

# Curva da densidade teórica
curve(f_lindley(x, theta),
  from = 0,
  to = max(X),
  add = TRUE,
  lwd = 2)

```

### **Exercício 9.**

```

# Parâmetros
n <- 1000
alpha <- 2
beta <- 2

# Semente para reprodutibilidade
set.seed(123)

# Função densidade da distribuição Gama
f_gamma <- function(x, alpha, beta) {
  (1 / (gamma(alpha) * beta^alpha)) *

```

```

    x^(alpha - 1) *
    exp(-x / beta)
}

# Função CDF calculada numericamente
F_gamma <- function(x, alpha, beta) {

  integrate(
    f_gamma,
    lower = 0,
    upper = x,
    alpha = alpha,
    beta = beta
  )$value
}

# Função para gerar uma amostra pelo método da inversa
r_gamma_inversa <- function(n, alpha, beta) {

  # Gerando U ~ Uniforme(0,1)
  U <- runif(n)

  # Vetor para armazenar as observações
  X <- numeric(n)

  # Inversão numérica
  for (i in 1:n) {

    X[i] <- uniroot(
      function(x) F_gamma(x, alpha, beta) - U[i],
      interval = c(0, 100)
    )$root

  }

  return(X)
}

# Gerando a amostra
X <- r_gamma_inversa(n, alpha, beta)

# Histograma da amostra
hist(X,
      probability = TRUE,
      breaks = 30,
      main = "Distribuição Gama",
      xlab = "X",
      ylab = "Densidade",
      col = "lightgray",
      border = "white")

# Curva da densidade teórica
curve(f_gamma(x, alpha, beta),

```

```

from = 0,
to = max(X),
add = TRUE,
lwd = 2)

```

### Exercício 10.

```

# Função CDF da Normal Generalizada
F_generalizada <- function(x, mu, alpha, beta) {

  # Criando um vetor para armazenar os resultados
  F <- numeric(length(x))

  # Caso x < mu
  ind1 <- x < mu

  F[ind1] <- 0.5 * (
    1 - pgamma(
      ((mu - x[ind1]) / alpha)^beta,
      shape = 1 / beta
    )
  )

  # Caso x >= mu
  ind2 <- x >= mu

  F[ind2] <- 0.5 + 0.5 * pgamma(
    ((x[ind2] - mu) / alpha)^beta,
    shape = 1 / beta
  )

  return(F)
}

```

Podemos testar a CDF no ponto  $\mu$ . Como  $F(\mu)=0.5$   $F(\mu)=0.5$ :

```

# Parâmetros
mu <- 0
alpha <- 2
beta <- 3

# Verificando F(mu)
F_generalizada(mu, mu, alpha, beta)

```

O resultado deve ser:

```
[1] 0.5
```

### Método uniroot

```

# Função para gerar observações pelo método da inversa
r_generalizada <- function(n, mu, alpha, beta) {

```

```

# Gerando U ~ Uniforme(0,1)
U <- runif(n)

# Vetor para armazenar as observações
X <- numeric(n)

# Inversão numérica
for (i in 1:n) {

  X[i] <- uniroot(
    function(x) {
      F_generalizada(x, mu, alpha, beta) - U[i]
    },
    interval = c(mu - 100 * alpha, mu + 100 * alpha)
  )$root

}

return(X)
}

```

### **Considerando uma mostra com n=1000**

```

# Parâmetros
n <- 1000
mu <- 0
alpha <- 2
beta <- 3

# Semente para reprodutibilidade
set.seed(123)

# Gerando a amostra
X <- r_generalizada(n, mu, alpha, beta)

# Função densidade teórica
f_generalizada <- function(x, mu, alpha, beta) {

  (beta / (2 * alpha * gamma(1 / beta))) *
  exp(
    -((abs(x - mu) / alpha)^beta)
  )
}

# Histograma da amostra
hist(X,
  probability = TRUE,
  breaks = 30,
  main = "Distribuição Normal Generalizada",
  xlab = "X",
  ylab = "Densidade",

```

```
col = "lightgray",
border = "white")

# Curva da densidade teórica
curve(
  f_generalizada(x, mu, alpha, beta),
  from = min(X),
  to = max(X),
  add = TRUE,
  lwd = 2
)
```