Estatística Computacional

Simulação de Monte Carlo

Prof. Paulo Cerqueira Jr
Faculdade de Estatística - FAEST
Programa de Pós-graduação em Matemática e Estatística - PPGME

https://github.com/paulocerqueirajr

Introdução

Introdução

- A simulação de Monte Carlo (SMC) é uma técnica amplamente utilizada para entender o comportamento de modelos estatísticos sob diferentes cenários de incerteza.
- Na estatística, as simulações de Monte Carlo são úteis para explorar distribuições de variáveis aleatórias, estimar incertezas, e validar modelos em cenários onde soluções analíticas são complexas ou impossíveis.
- Elas têm aplicações em diversas áreas, incluindo ciências sociais, economia, finanças e estudos ambientais.

Introdução

- A metodologia da simulação de Monte Carlo consiste em quatro etapas principais:
  1. Definir o modelo estatístico: O modelo é formulado para descrever a relação entre variáveis de interesse.
  2. Gerar amostras aleatórias: Utilizando distribuições de probabilidade apropriadas, geramos amostras de variáveis aleatórias que representam os inputs do modelo.
  3. Calcular o resultado para cada amostra: Aplicamos o modelo às amostras geradas, obtendo uma distribuição dos resultados.
  4. Analisar os resultados: Com base nos resultados, estimamos estatísticas, intervalos de confiança e outras métricas que representam a incerteza do modelo.

Modelo Normal

Modelo Normal

- Suponha que X seja uma variável aleatória que segue uma distribuição normal, $X \sim N(\mu, \sigma^2)$, onde:
  - μ é a média da população (desconhecida),
  - σ^2 é a variância da população (também desconhecida).
- Dado um conjunto de n observações independentes $X_1, X_2, \ldots, X_n$, nossos estimadores de μ e σ^2 são definidos como:

- **Média Amostral ($\hat{\mu}$)**:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

- **Variância Amostral ($\hat{\sigma}^2$)**:

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} (X_i - \hat{\mu})^2$$

- Esses estimadores são não-viesados, o que significa que em média eles convergem para os valores reais de μ e σ^2 quando n é suficientemente grande.

Passos para a Simulação de Monte Carlo

1. **Definir os Parâmetros Verdadeiros**: Especificar os valores de μ e σ^2 para o modelo populacional.
2. **Especificar o Tamanho da Amostra e o Número de Simulações**: Definir o tamanho de cada amostra (n) e o número total de simulações.
3. **Gerar Amostras Aleatórias**: Para cada simulação, gerar uma amostra aleatória de tamanho n de uma distribuição normal com média μ e desvio padrão σ.
4. **Calcular as Estimativas para Cada Amostra**: Calcular $\hat{\mu}$ e $\hat{\sigma}^2$ para cada amostra gerada.
5. **Analisar os Resultados**: Examinar a distribuição das estimativas de $\hat{\mu}$ e $\hat{\sigma}^2$ e calcular as médias e os desvios padrão dessas distribuições.

Resultados

Vamos implementar a metodologia acima em R para realizar uma simulação de Monte Carlo, estimando os parâmetros μ e σ^2 de uma distribuição normal.

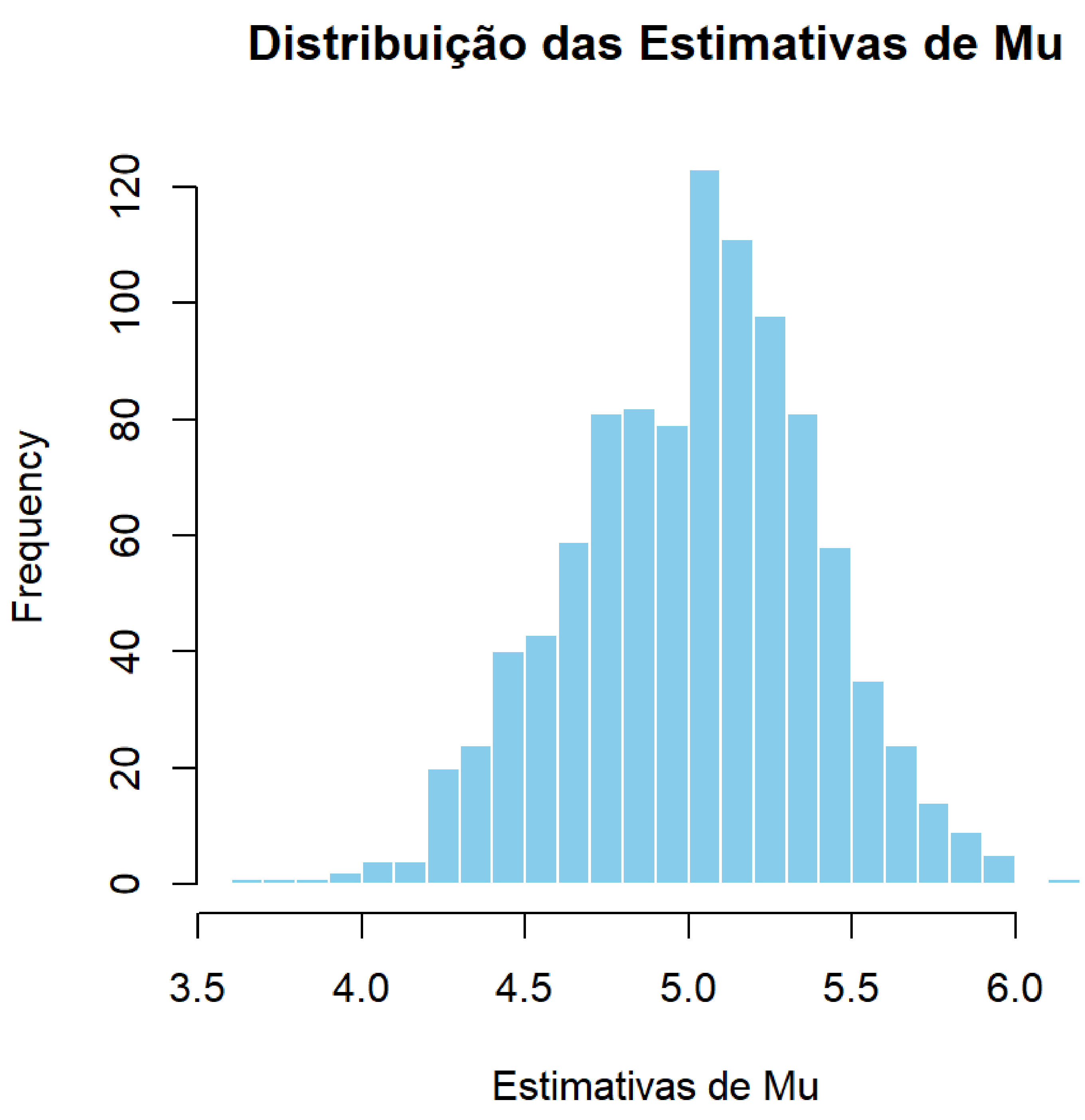
```
# Definindo os Parâmetros Verdadei
set.seed(123456789)  # Para replic
true_mu <- 5         # Média verdade
true_sigma <- 2      # Desvio padrão
n <- 30              # Tamanho da am
n_simulations <- 1000  # Número de

# Inicializando vetores para armaz
mu_estimates <- numeric(n_simulati
sigma2_estimates <- numeric(n_simu

# Simulação de Monte Carlo
for (i in 1:n_simulations) {
  # Gerando uma amostra da distrib
```

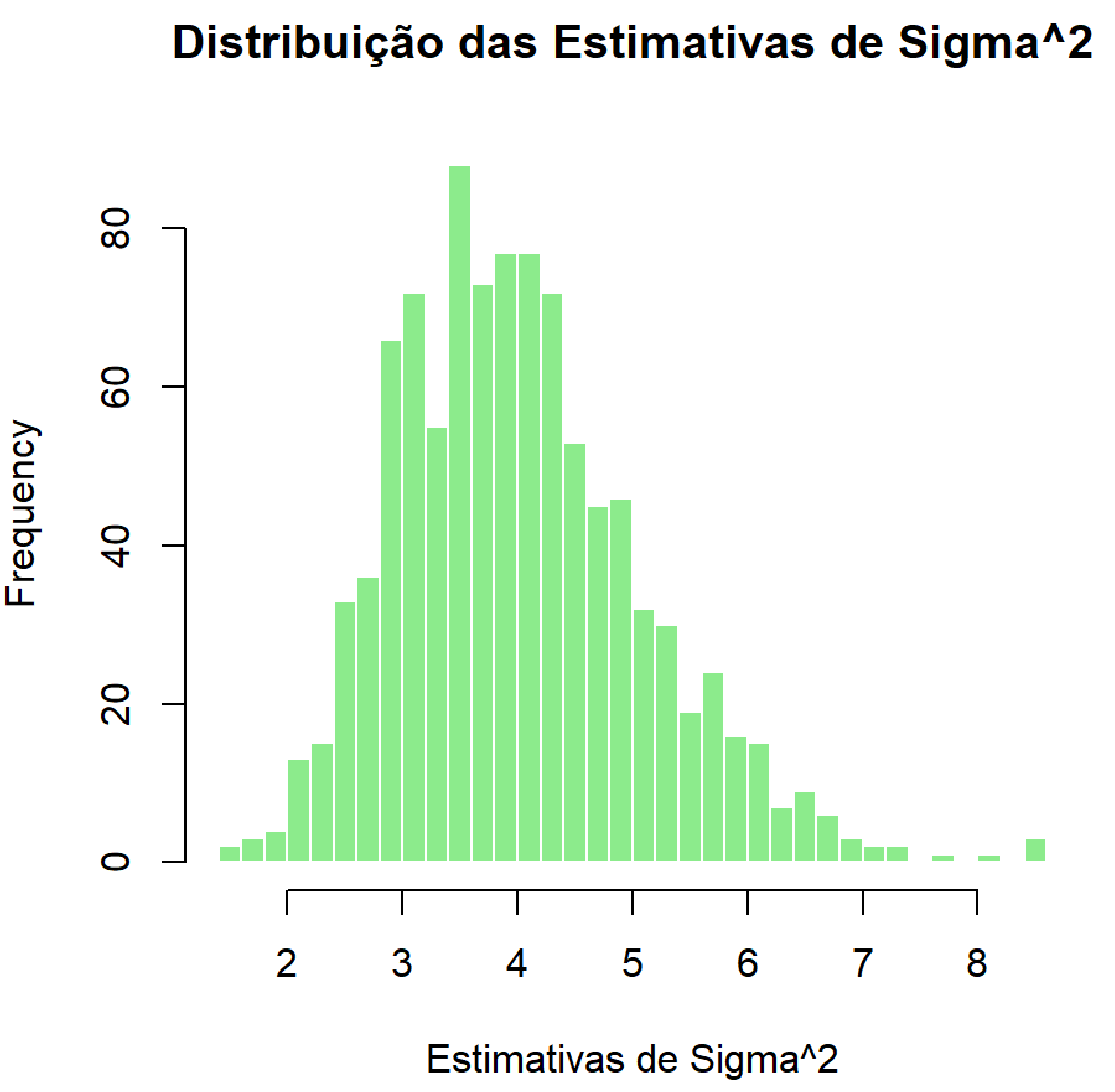
```
# Análise dos Resultados
mean_mu_estimate <- mean(mu_estima
mean_sigma2_estimate <- mean(sigma
sd_mu_estimate <- sd(mu_estimates)
sd_sigma2_estimate <- sd(sigma2_es

cat("Estimativa da média (mu):", m
cat("Erro padrão da média estimada
cat("Estimativa da variância (sigm
cat("Erro padrão da variância esti

# Visualização das Distribuições d
par(mfrow = c(1, 2))  # Dividir a
hist(mu_estimates, breaks = 30, ma
```

https://github.com/paulocerqueirajr

Resultados

```
Estimativa da média (mu): 5.016872
Erro padrão da média estimada: 0.3745315
Estimativa da variância (sigma^2): 4.016886
Erro padrão da variância estimada: 1.079607
```





Interpretação dos resultados

- Após a simulação de Monte Carlo, obtemos distribuições das estimativas $\hat{\mu}$ e $\hat{\sigma}^2$.
- Essas distribuições fornecem uma ideia da variabilidade esperada das estimativas de μ e σ^2 para o tamanho amostral dado.
- A média das estimativas de $\hat{\mu}$ deve ser próxima da média verdadeira μ.
- A média das estimativas de $\hat{\sigma}^2$ deve ser próxima da variância verdadeira σ^2.
- Os desvios padrão das distribuições de $\hat{\mu}$ e $\hat{\sigma}^2$ fornecem uma medida da precisão das estimativas para o tamanho amostral utilizado.

Caso mais geral

- Seja $(X_1, X_2, \ldots, X_n)$ uma amostra aleatória da variável X.
- A função de verossimilhança de $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)$, correspondente a amostra observada $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ e será denotada por

$$L(\boldsymbol{\theta} \mid \mathbf{x}) = \prod_{i=1}^{n} f(x_i \mid \boldsymbol{\theta})$$

- A função de log-verossimilhança é dada por

$$\ell(\boldsymbol{\theta} \mid \mathbf{x}) = \sum_{i=1}^{n} \log f(x_i \mid \boldsymbol{\theta})$$

Estimativa da variância

- $I(\boldsymbol{\theta})$ é a informação de Fisher de $\boldsymbol{\theta}$ e é definida como:

$$I_F(\boldsymbol{\theta}) = E\left[\left(\frac{\partial \log L(\theta \mid \mathbf{X})}{\partial \theta}\right)^2\right] = -E\left[\frac{\partial^2 \log L(\theta \mid \mathbf{X})}{\partial \theta^2}\right] = -E\left[\sum_{i=1}^{n} \frac{\partial^2 \log f(X_i \mid \theta)}{\partial \theta^2}\right] = = \sum_{i=1}^{n} E\left[-\frac{\partial^2 \log f(}{\partial \theta}\right.$$

- No caso multiparamétrico, temos que $I_F(\boldsymbol{\theta})$ é vista como uma matriz.
- Uma alternativa para o cálculo da Informação de Fisher é considerar o caso "observado" ao invés do "esperado".
- Informação de Fisher observada é dada por

$$\hat{I}(\boldsymbol{\theta}) = \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^2}\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}$$

Estimativa da variância

- A estimativa da variância do estimador é obtida tomando

$$Var(\hat{\boldsymbol{\theta}}) = diag\left(\hat{I}(\boldsymbol{\theta})^{-1}\right) = \left[Var(\hat{\theta}_1), Var(\hat{\theta}_2), \ldots, Var(\hat{\theta}_k)\right]$$

- Dessa forma podemos calcular o intervalo de $\gamma\%$ de confiança assintótico:

$$IC(\theta, \gamma\%) = \hat{\theta} \pm z_{\gamma/2} \frac{\sqrt{Var(\hat{\theta})}}{\sqrt{n}} = \hat{\theta} \pm z_{\gamma/2} EP(\hat{\theta})$$

Modelo Weibul

- A função de densidade de probabilidade (fdp) de Weibull é dada por:

$$f(x; s, a) = asx^{a-1}\exp(-sx^{a})$$

- A função de log-verossimilhança para uma amostra de n observações é dada por:

$$\ln L(s, a) = n\ln a + n\ln s + (a-1)\sum_{i=1}^{n}\ln x_i - s\sum_{i=1}^{n}x_i^{a}$$

Derivadas de primeira ordem

As derivadas parciais da função de log-verossimilhança em relação a a e s são:

$$\frac{\partial \ln L}{\partial a} = \frac{n}{a} + \sum_{i=1}^{n} \ln x_i - s \sum_{i=1}^{n} x_i^a \ln x_i$$

e

$$\frac{\partial \ln L}{\partial s} = \frac{n}{s} - \sum_{i=1}^{n} x_i^a$$

Derivadas de Segunda Ordem

As derivadas parciais de segunda ordem são:

$$\frac{\partial^2 \ln L}{\partial a^2} = -\frac{n}{a^2} - s\sum_{i=1}^{n} x_i^a (\ln x_i)^2,$$

$$\frac{\partial^2 \ln L}{\partial s^2} = -\frac{n}{s^2},$$

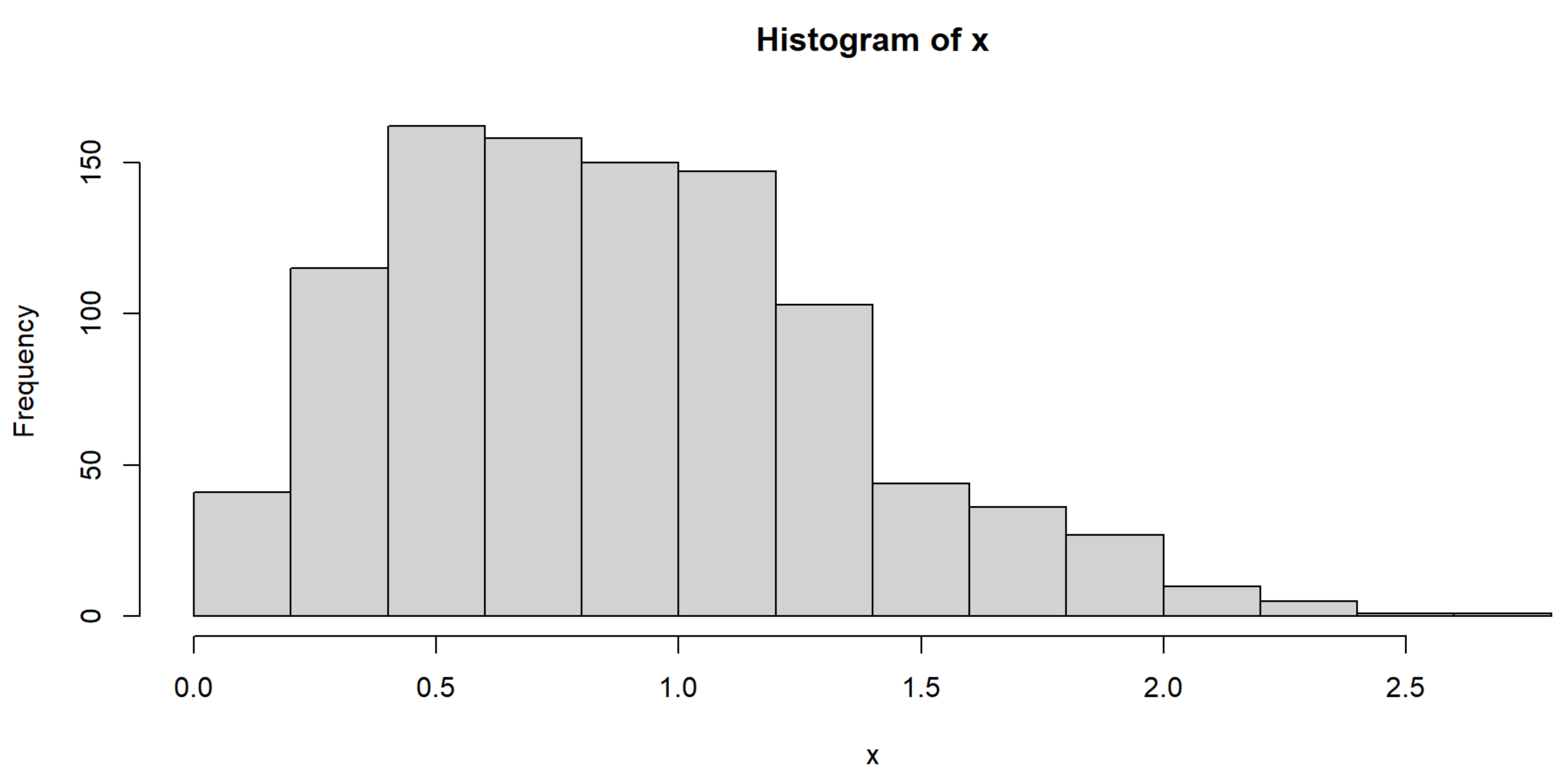
e

$$\frac{\partial^2 \ln L}{\partial s \partial a} = -\sum_{i=1}^{n} x_i^a \ln x_i.$$

Gerando os dados:

Foi gerado uma amostra de tamanho `1000` de uma distribuição de Weibull com parâmetro de forma $a = 2$ e escala $s = 1$.

```r
## Modelo Weibull:

set.seed(1234567890)

n <- 1000 # tamanho da amostra
a.p <- 2   # forma
s.p <- 1   # escala
x <- rweibull(n, shape = a.p, scale = 1/(s.p^a.p))  # dados gerados
hist(x)
```



Usando a função de log-verossimilhança

Neste passo precisamos somente da função de $\log$ verossimilhança.

```r
logWeibull <- function(theta, dados){
  a <- theta[1]
  s <- theta[2]
  n <- length(dados)
  x <- dados

  l <- n*log(a)+n*log(s)+(a-1)*sum(log(x))-s*sum((x)^a)
  return(-l)
}
```

Usando a função `optim` para otimização

Dentro da função `logWeibull` o objeto `l` dará retorno negativo, pois a função `optim` determina ponto de mínimo.

```r
theta0 <- c(3, 2) # Chute inicial
est <- optim(par = theta0, fn = logWeibull, gr =NULL , method ="BFGS" ,
      hessian = TRUE, dados=x)

est$par
```

```
[1] 1.974179 1.020102
```

```r
est$hessian
```

```
         [,1]     [,2]
[1,] 459.6588 198.7491
[2,] 198.7491 960.9790
```

- Tomando a diagonal da inversa da matriz hessiana

```r
ep <- sqrt(diag(solve(est$hessian)))
ep
```

```
[1] 0.04887924 0.03380534
```