Estatística Matemática

Métodos de otimização

Prof. Paulo Cerqueira Jr - cerqueirajr@ufpa.br
Faculdade de Estatística - FAEST
Programa de Pós-Graduação em Matemática e Estatística - PPGME
Instituto de Ciências Exatas e Naturais - ICEN

https://github.com/paulocerqueirajr

Introdução

Introdução

- Nessa unidade estaremos interessados no seguinte:

> Problema 1: Seja $f : \Theta \to \mathbb{R}$. Encontre um ponto $\theta \in \Theta$ que minimiza a função f.

- É importante observar que o problema de encontrar um ponto $\theta \in \Theta$ que maximiza uma função $g : \Theta \to \mathbb{R}$, recai no problema anterior, basta ver que maximizar g é o mesmo que minimizar $f = -g$.
- Problemas de otimização (ou seja, de minimização ou maximização) ocorrem com frequência em diversas áreas das Ciências Exatas, em particular, na Estatística.

Método de Newton-Raphson

Caso unidimensional - Descrição do Método.

- O método de Newton-Raphson é um algoritmo apropriado para encontrar raízes (ou zeros) de funções.
- Formalmente, estamos interessados em encontrar um ponto $\hat{\theta}$ no domínio de uma função $h : \Theta \to \mathbb{R}$ tal que $h(\hat{\theta}) = 0$.
- Inicialmente vamos considerar o caso onde h é uma função de uma única variável.

Caso unidimensional - Descrição do Método.

- Nessa situação, o método pode ser descrito nos seguintes passos:
  1. Fixe um número real $\epsilon > 0$;
  2. Dê uma aproximação inicial θ_0 para $\hat{\theta}$;
  3. Para $k \geq 0$, faça

$$\theta_{k+1} = \theta_k - \frac{h(\theta_k)}{h'(\theta_k)}.$$

  4. Pare o processo iterativo se $|\theta_{k+1} - \theta_k| < \epsilon$. Caso contrário, volte para o passo anterior.

Caso unidimensional - Descrição do Método

- O método de Newton-Raphson possui uma interpretação geométrica simples.
- Basta ver que para todo $k > 0$:

$$h^{'}(\theta_k) = \frac{h(\theta) - 0}{\theta_k - \theta_{k+1}} \quad \Rightarrow \quad \theta_{k+1} = \theta_k - \frac{h(\theta_k)}{h^{'}(\theta_k)}$$

dado alguma aproximação inicial criteriosa θ_0.

- É possível provar que a sequência $(\theta_k)_{k \geq 0}$ converge para $\hat{\theta}$ quando $k \to \infty$, se θ_0 é escolhido próximo de $\hat{\theta}$.

Exemplo

> A função $h(\theta) = 2\theta - \cos(\theta)$ possui uma raiz real $\hat{\theta}$ isolada no intervalo $[0, \pi/4]$. Encontre um valor aproximado de $\hat{\theta}$ usando o método de Newton-Raphson.

`Solução:` Primeiramente temos sabemos que:

$$h^{'}(\theta) = 2 + \sin(\theta).$$

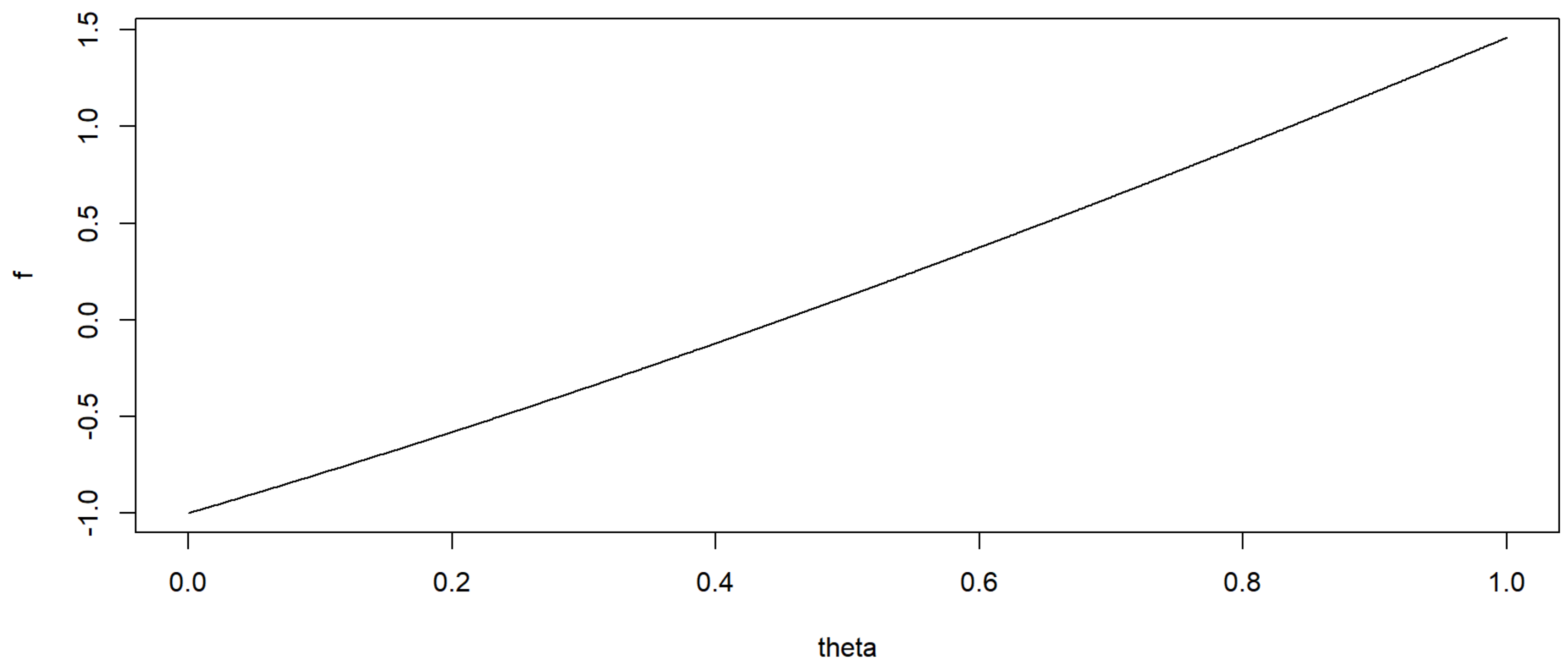
- Fixe $\epsilon = 0,0001$ e $\theta_0 = \pi/8$.
- Agora itere e para $k \geq 0$,

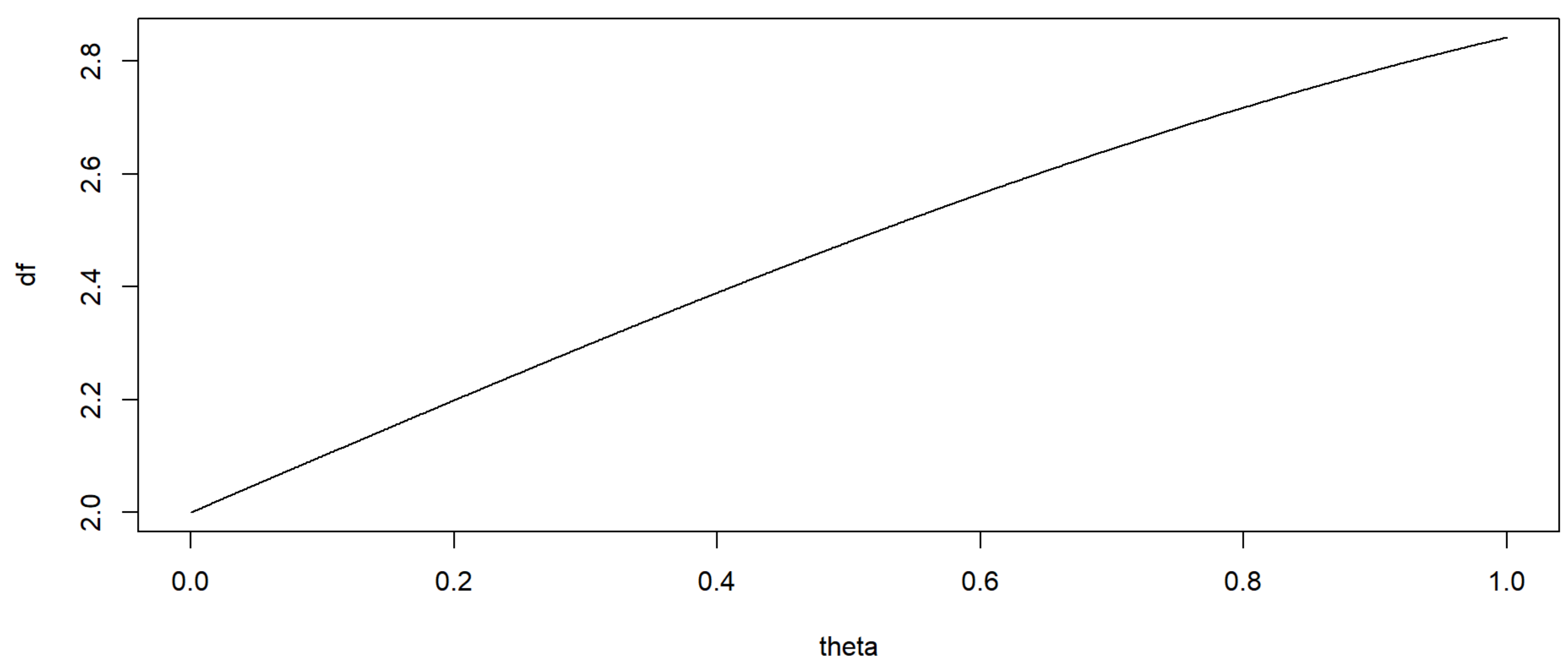
$$\theta_{k+1} = \theta_k - \frac{2\theta_k - \cos(\theta_k)}{2 + \sin(\theta_k)}$$

Exemplo

```r
theta <- seq(0, 1, length.out=1000)
f     <- (2*theta)-cos(theta)
plot(theta,f, type="l")
```

```r
theta <- seq(0, 1, length.out=1000)
df      <- 2+sin(theta)
plot(theta,df, type="l")
```





Exemplo:

- Código em R:

```r
theta.0 <- pi/8
theta.0
```

```
[1] 0.3926991
```

```r
precisao <- 0.0001
dif <- 1
while(dif > precisao){
  razao <- (2*theta.0 - cos(theta.0))/(2 +
                               sin(theta.0))
  theta.1 <- theta.0 - razao
  dif <- abs(theta.1 - theta.0)
  theta.0 <- theta.1
  cat("Valor de theta=",theta.0, "\n")
}
```

```
Valor de theta= 0.450819
Valor de theta= 0.4501837
Valor de theta= 0.4501836
```

```r
raiz <- theta.0
raiz
```

```
[1] 0.4501836
```

```r
h <- 2*raiz - cos(raiz)
h
```

```
[1] 2.553513e-15
```

Newton-Raphson para Otimização

Newton-Raphson para Otimização

- Considere o problema 1 para o caso em que f é uma função de uma única variável.
- O método de Newton-Raphson é apropriado para resolver numericamente este problema de otimização, basta encontrar as raízes de $h = f^{'}$.
- Neste caso o mínimo θ pode ser encontrado seguindo os seguintes passos:

1. Fixe um número real $\epsilon > 0$;
2. Dê uma aproximação inicial θ_0 para $\hat{\theta}$;
3. Para $k \geq 0$, faça

$$\theta_{k+1} = \theta_k - \frac{f^{'}(\theta_k)}{f^{''}(\theta_k)}.$$

4. Pare o processo iterativo se $|\theta_{k+1} - \theta_k| < \epsilon$. Caso contrário, volte para o passo anterior.

Exemplo

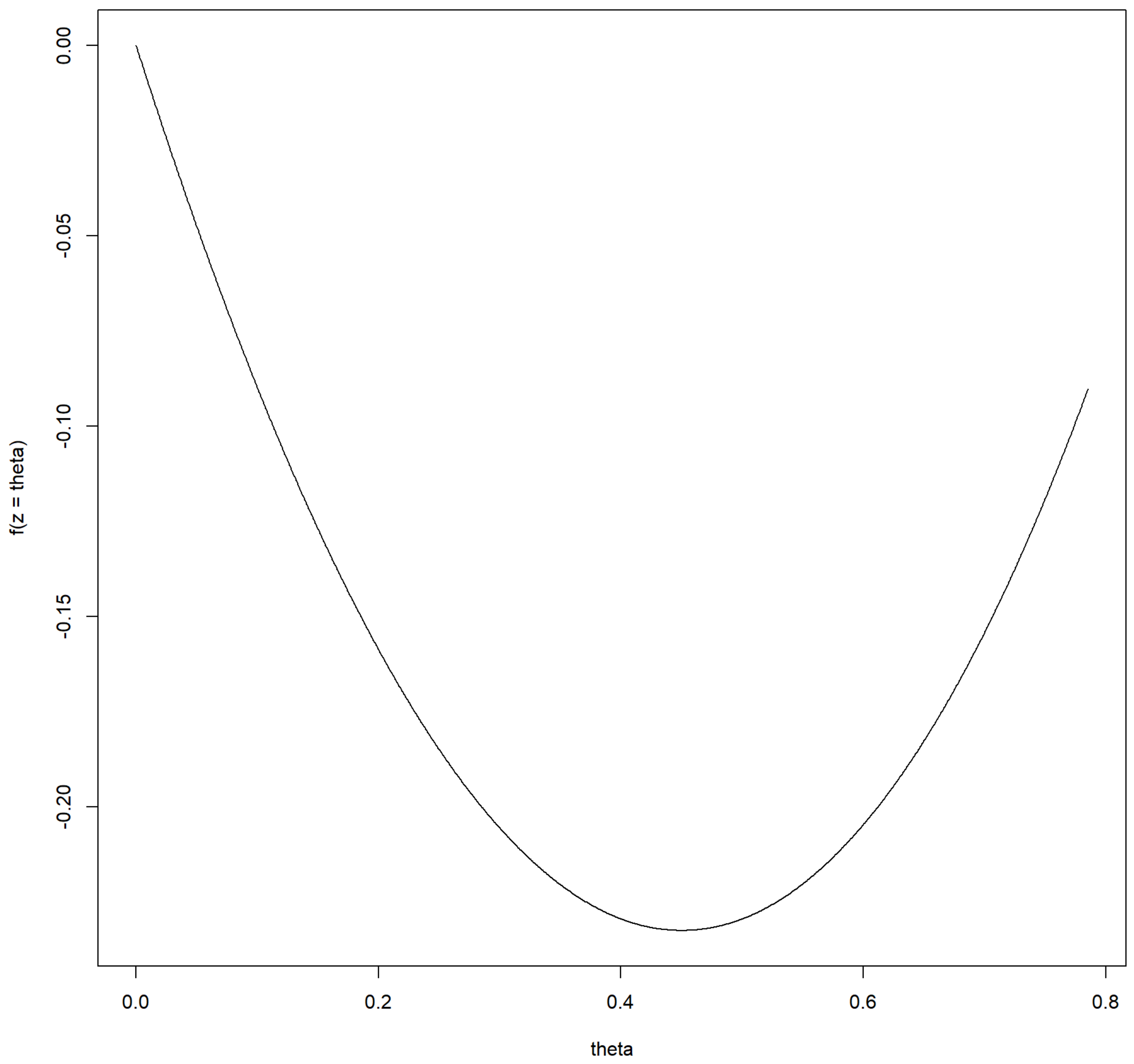
> Utilize o método de Newton-Raphson para encontrar o mínimo da função $f(\theta) = \theta^2 - \sin(\theta)$.

`Solução:`

- Fixe $\epsilon = 0,0001$ e $\theta_0 = \pi/8$, itere e para $k \geq 0$

$$\theta_{k+1} = \theta_k - \frac{2\theta_k - \cos(\theta_k)}{2 + \sin(\theta_k)}$$

Exemplo:



- O R também possui funções prontas para pesquisar, dentro de um intervalo, um ponto de mínimo (ou de máximo) de uma função.
- Veja o código abaixo aplicado para o exemplo em questão:

```
$minimum
[1] 0.4501836

$objective
[1] -0.2324656
```

Newton-Raphson em Estatística

Newton-Raphson em Estatística

- Seja $(X_1, ..., X_n)$ uma a.a. de tamanho n da distribuição de uma v.a. X com densidade $f(x;\theta)$ onde θ pertence ao espaço paramétrico Θ (por enquanto, considere que Θ é unidimensional).
- A função de verossimilhança de θ $(L : \Theta \to \mathbb{R})$ associada à a.a. observada $\mathbf{x} = (x_1, ..., x_n)$ é definida por

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

- Seja a função de log verossimilhança dada por:

$$\ell(\theta) = \ln(L(\theta)).$$

e a função escore:

$$U(\theta) = \frac{d \ln L(\theta)}{d\theta} = \frac{d\ell(\theta)}{d\theta} = \ell^{'}.$$

Newton-Raphson em Estatística

- Portanto o estimador de máxima verossimilhaça, denotado por $\hat{\theta}$, satisfaz as seguintes equações:

$$U(\hat{\theta}) = 0 \quad \Rightarrow \quad \hat{\theta} = \overset{\theta \in \Theta}{\max} \ \ell(\theta).$$

- Em alguns casos pode ser difícil obter uma solução analítica explícita para as equações.
- Nesses casos, é possível obter uma soluçãao aproximada para $\hat{\theta}$ por meio de métodos numéricos.
- Um alternativa consiste em utilizar o método de Newton-Raphson para aproximar a raiz da função escore (ou maximizar a logverossimilhança).

Newton-Raphson em Estatística

- Explicitamente, basta seguir o seguinte algoritmo:

1. Fixe um número real $\epsilon > 0$;
2. Dê uma aproximação inicial θ_0 para $\hat{\theta}$;
3. Para $k \geq 0$, faça

$$\theta_{k+1} = \theta_k - \frac{U(\theta_k)}{U'(\theta_k)} = \theta_k - \frac{\ell'(\theta_k)}{\ell''(\theta_k)}.$$

4. Pare o processo iterativo se $|\theta_{k+1} - \theta_k| < \epsilon$. Caso contrário, volte para o passo anterior.

- A sequência $(\theta_k)_{k \geq 0}$ converge para $\hat{\theta}$ quando $k \to \infty$, se θ_0 é escolhido próximo de $\hat{\theta}$

```
(Dica: um gráfico de $U(\theta)$ ou $\ell(\theta)$ pode ajudar nessa escolha
inicial).
```

Método Escore

Método Escore

- Em alguns casos, a substituição de $U^{'}(\theta_k)$ por $E(U^{'}(\theta_k))$, apresenta significativa simplificação no procedimento.
- Esse método é conhecido como método do escore e pode ser descrito assim:

1. Fixe um número real $\epsilon > 0$;
2. Dê uma aproximação inicial θ_0 para $\hat{\theta}$;
3. Para $k \geq 0$, faça

$$\theta_{k+1} = \theta_k - \frac{U(\theta_k)}{E(U^{'}(\theta_k))} = \theta_k - \frac{U(\theta_k)}{I(\theta_k)}.$$

em que $I(\theta_k)$ é a informação de Fisher de θ.

4. Pare o processo iterativo se $|\theta_{k+1} - \theta_k| < \epsilon$. Caso contrário, volte para o passo anterior.

- Novamente, a sequência $(\theta_k)_{k \geq 0}$ converge para $\hat{\theta}$ quando $k \to \infty$, se θ_0 é escolhido próximo de $\hat{\theta}$.

Exemplo

Sejam $X_1, \ldots, X_n$ uma a.a. de X, com função densidade dada por

$$f(x \mid \theta) = \frac{1}{2}(1+\theta x), \;\; -1 \leq x \leq 1, \;\; -1 \leq \theta \leq 1.$$

Determine o EMV para θ pelo método de Newton-Raphson e Escore.

Sol. Inicalmente temos que a função de verossimilhança é dada por

$$L(x \mid \theta) = \frac{1}{2^n} \prod_{i=1}^{n} (1+\theta x_i),$$

de modo que

$$U(\theta) = \ell^{'} = \sum_{i=1}^{n} \frac{x_i}{1+\theta x_i}$$

Exemplo

E dessa forma

$$U^{'}(\theta) = \ell^{''} = -\sum_{i=1}^{n} \frac{x_i^2}{(1+\theta x_i)^2}.$$

A informação de Fisher de θ é igual,

[Math Processing Error]

Gerou-se $n = 20$ valores, com $\theta = 0.4$ usando a função densidade do exemplo via método da transformação inversa, logo

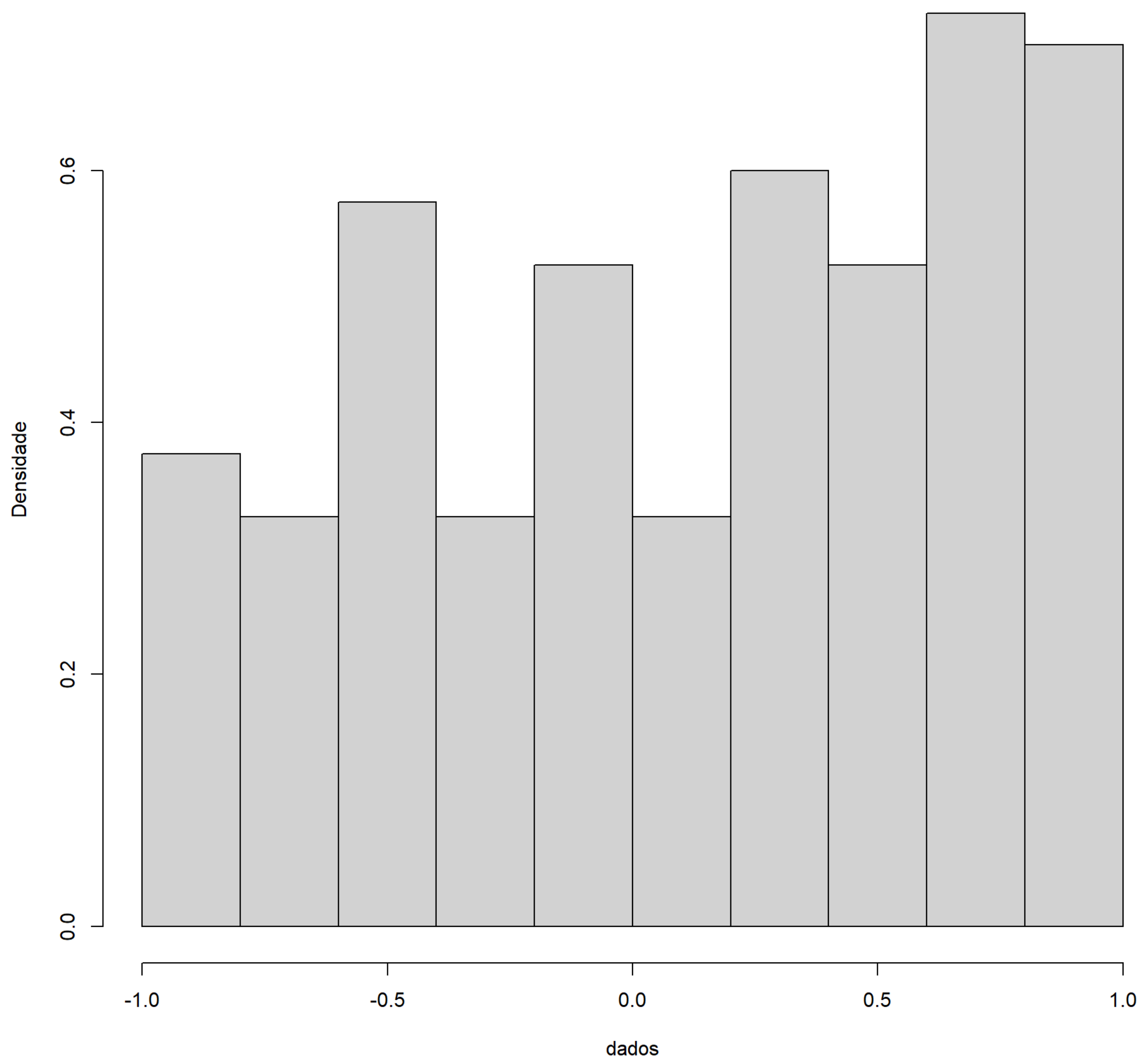
$$x = \frac{-1 + 2\sqrt{1/4 - \theta(1/2 - \theta/4 - u)}}{\theta}.$$

em que $U \sim U(0,1)$.

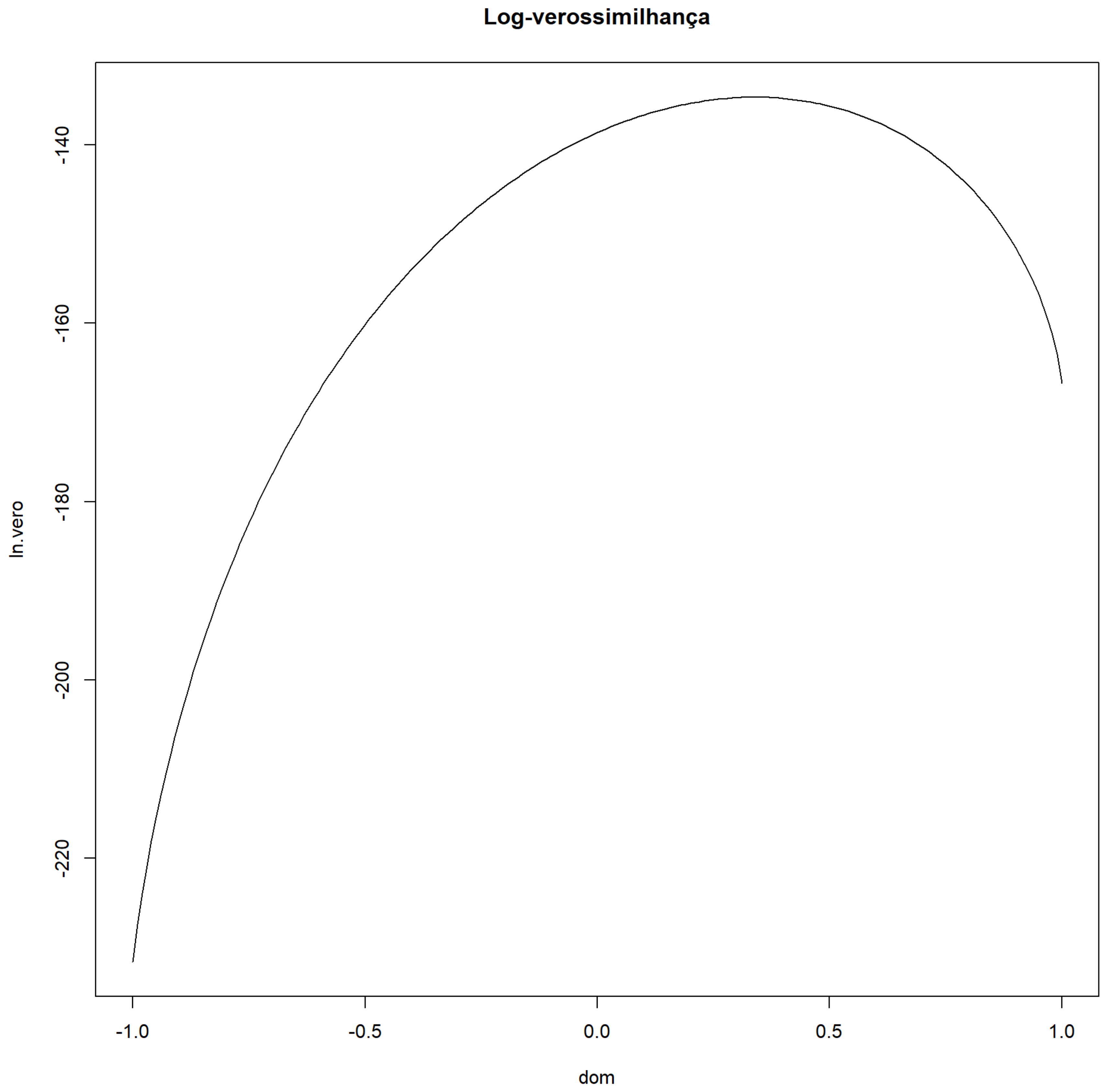
Exemplo

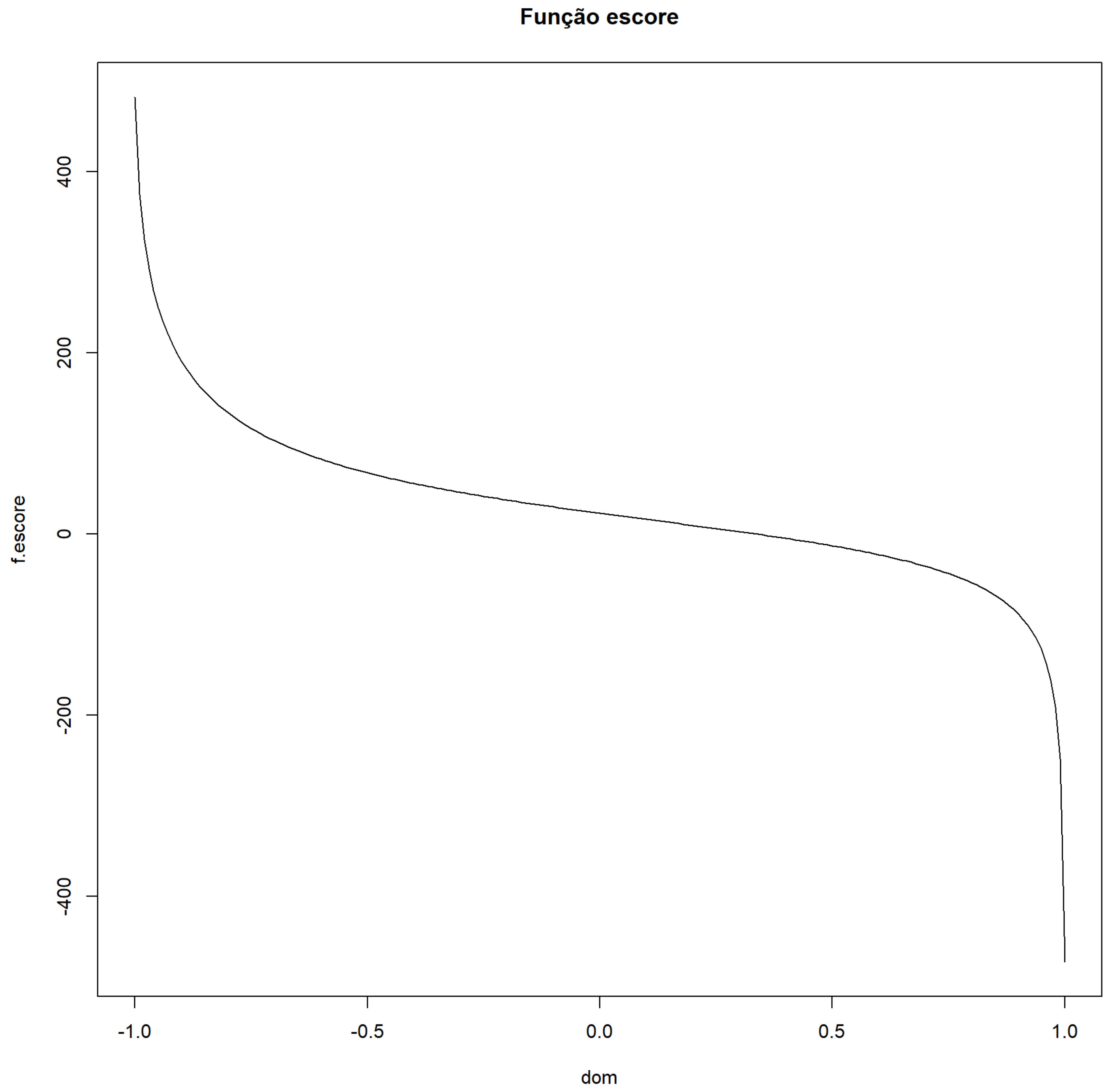
- Código em R:

```r
set.seed(123456)
n <- 200
theta <- 0.4
u <- runif(n,0,1)
raiz <- 1-(theta*(2-theta))+(4*theta*u) # t
dados <-(-1+sqrt(raiz))/(theta)
```



Exemplo


Log-verossimilhança
ln.vero
-140
-160
-180
-200
-220
-1.0
-0.5
0.0
0.5
1.0
dom


Função escore
f.escore
400
200
0
-200
-400
-1.0
-0.5
0.0
0.5
1.0
dom

Exemplo - Comparação dos métodos:

`Newton-Raphson`:

```r
theta.zero <- 0.15
precisao <- 0.000001
dif <- 1
while(dif > precisao){
 num <- S(theta.zero)
 den <- S.prime(theta.zero)
 theta.um <- theta.zero - (num/den)
 dif <- abs(theta.um - theta.zero)
 theta.zero <- theta.um
 print(theta.zero)
}
```

```
[1] 0.3459363
[1] 0.3398386
[1] 0.3398224
[1] 0.3398224
```

```r
raiz.NR <- theta.zero
raiz.NR # Método NR.
```

```
[1] 0.3398224
```

`Escore`:

```r
theta.zero <- 0.15
dif <- 1
while(dif > precisao){
 num <- S(theta.zero)
 a <- 2*theta.zero
 b <- log((1+theta.zero)/(1-theta.zero))-a
 den <- n*(1/(2*theta.zero^3))*b
 theta.um <- theta.zero + (num/den)
 dif <- abs(theta.um - theta.zero)
 theta.zero <- theta.um
 print(theta.zero)
}
```

```
[1] 0.3433802
[1] 0.3397711
[1] 0.3398231
[1] 0.3398223
```

```r
raiz.E <- theta.zero
raiz.E # Método Escore
```

```
[1] 0.3398223
```

Caso Multidimensional

Método de Newton-Raphson

- Agora considere o problema de otimização quando Θ é um espaço multidimensional.
- Antes de apresentar o método de Newton-Raphson nesse caso, vejamos alguns conceitos básicos de Cálculo.

> Noções preliminares:

Seja n um inteiro positivo e seja $D \subseteq \mathbb{R}^n$. Considere uma função g que associa a cada $\mathbf{x} = (x_1, \ldots, x_n) \in D$ um numero real $g(\mathbf{x})$, ou seja, $g : D \to \mathbb{R}$. O gradiente de g, denotado por

[Math Processing Error]

Método de Newton-Raphson

- Seja $(X_1, \ldots, X_n)$ uma a.a. de tamanho n da distribuição de uma v.a. X com densidade $f(x;\theta)$ onde $\theta = (\theta_1, \ldots, \theta_n)$ pertence ao espaço paramétrico $\boldsymbol{\Theta}$.
- A função de verossimilhança de θ $(L : \boldsymbol{\Theta} \to \mathbb{R})$ associada à a.a. observada $\mathbf{x} = (x_1, \ldots, x_n)$ é definida por

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

- Seja a função de log verossimilhança dada por:

$$\ell(\theta) = \ln(L(\theta)).$$

Método de Newton-Raphson

- O i-ésimo elemento do vetor escore, denotado por $U(\theta)$, é dado por

$$U_i(\theta) = \frac{\partial \ell(\theta)}{\partial \theta^{(i)}}.$$

- O (i, j)-elemento da matriz Hessiana, denotada por $H(\theta)$, é dado por

$$H_{ij} = \frac{\partial^2 \ell}{\partial \theta^{(i)} \partial \theta^{(j)}}.$$

Portanto o estimador de máxima verossimilhança, denotado por $\hat{\theta}$, satisfaz as seguintes equações:

$$U(\hat{\theta}) = \triangledown \ell(\hat{\theta}) = \mathbf{0} \qquad \hat{\theta} =^{\theta \in \Theta} \max \ell(\theta).$$

Newton-Raphson em Estatística

- Em alguns casos pode ser difícil obter uma solução analítica explícita para as equações.
- Nesses casos, é possível obter uma solução aproximada para $\hat{\theta}$ por meio de métodos numéricos.
- Um alternativa consiste em utilizar o método de Newton-Raphson para aproximar a raiz da função escore (ou maximizar a logverossimilhança).

Newton-Raphson em Estatística

- Explicitamente, basta seguir o seguinte algoritmo:
  1. Fixe um número real $\epsilon > 0$;
  2. Dê uma aproximação inicial θ_0 para $\hat{\theta}$;
  3. Para $k \geq 0$, faça

$$\theta_{k+1} = \theta_k - [H(\theta_k)]^{-1} U(\theta_k).$$

4. Pare o processo iterativo se $|\theta_{k+1} - \theta_k| < \epsilon$. Caso contrário, volte para o passo anterior.

- A sequência $(\theta_k)_{k \geq 0}$ converge para $\hat{\theta}$ quando $k \to \infty$, se θ_0 é escolhido próximo de $\hat{\theta}$

Método Escore

- Por vezes substituir de $H(\theta_k)$ por $E(H(\theta_k))$ pode apresentar significativa simplificação no procedimento.
- Esse método é conhecido como método do escore e pode ser descrito assim:

1. Fixe um número real $\epsilon > 0$;
2. Dê uma aproximação inicial θ_0 para $\hat{\theta}$;
3. Para $k \geq 0$, faça

$$\theta_{k+1} = \theta_k - [E(H(\theta_k))]^{-1} U(\theta_k) = \theta_k - [-I(\theta_k)]^{-1} U(\theta_k),$$

onde $I(\theta_k)$ é a matriz de informação de Fisher de θ.

4. Pare o processo iterativo se $|\theta_{k+1} - \theta_k| < \epsilon$. Caso contrário, volte para o passo anterior.

- A sequência $(\theta_k)_{k \geq 0}$ converge para $\hat{\theta}$ quando $k \to \infty$, se θ_0 é escolhido próximo de $\hat{\theta}$

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:

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

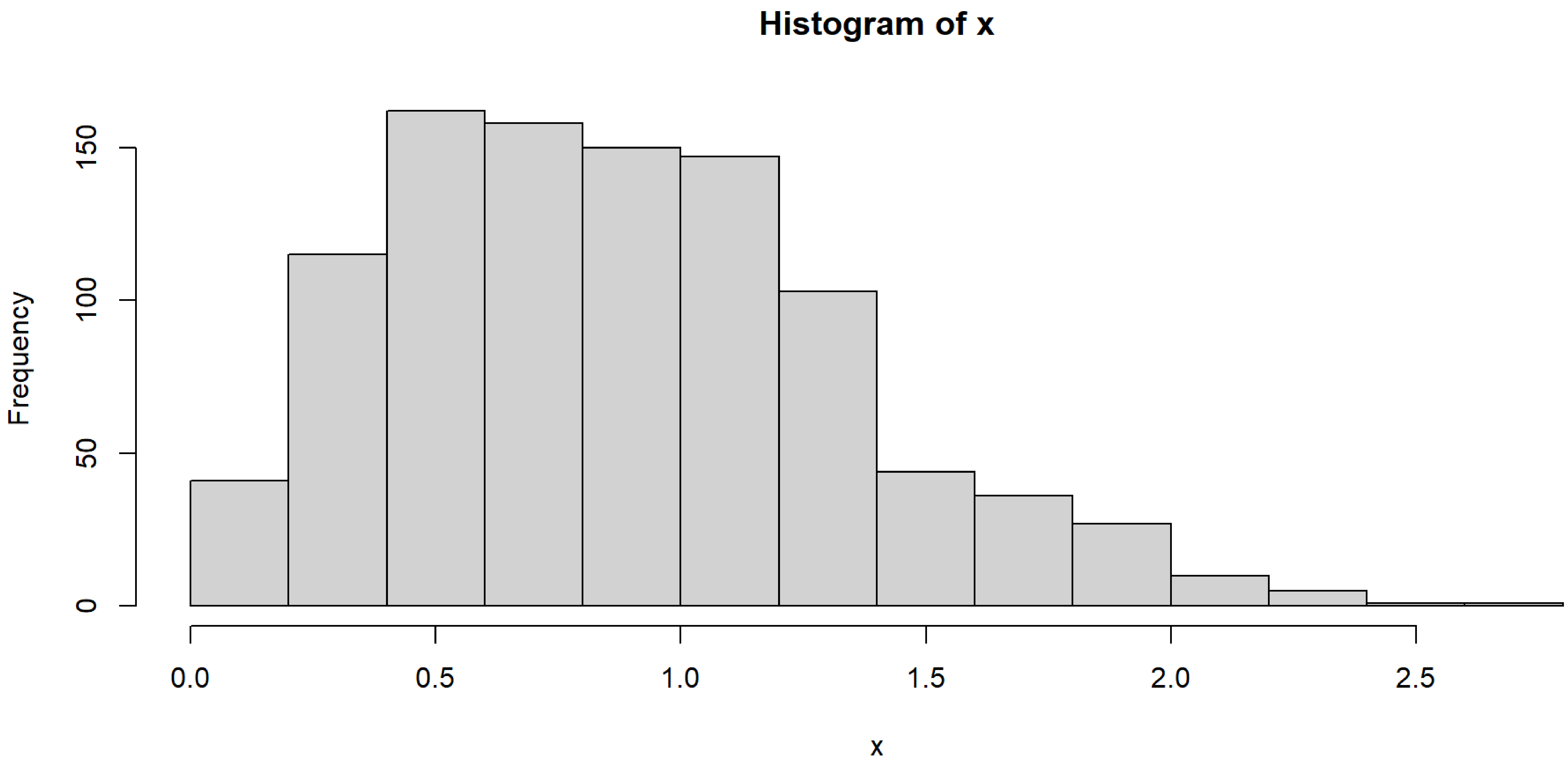
$$\frac{\partial^2 \ln L}{\partial s^2} = -\frac{n}{s^2},$$

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 $s = 1$.



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.

```
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
```

```
est$hessian
```

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

- Tomando a diagonal da inversa da matriz hessiana

```
[1] 0.04887924 0.03380534
```