Aula 10: Método de Newton Raphson

Prof. Dr. Eder Angelo Milani

07/06/2023

O método de Newton-Rapson

O procedimento de Newton-Rapson baseia-se na aproximação da função que se deseja maximizar por uma função quadrática. Para maximizar a log-verossimilhança, $l(\theta|x)$, consideremos a expansão de Taylor de segunda ordem ao redor do máximo $\hat{\theta}$

$$l(\theta|x) \approx l(\hat{\theta}|x) + (\theta - \hat{\theta})^T \frac{\partial l(\theta|x)}{\partial \theta}\Big|_{\theta=\hat{\theta}} + \frac{1}{2}(\theta - \hat{\theta})^T \frac{\partial^2 l(\theta|x)}{\partial \theta \partial \theta^T}\Big|_{\theta=\hat{\theta}}(\theta - \hat{\theta})$$

Então, para θ numa vizinhança de $\hat{\theta}$,

$$\frac{\partial l(\theta|x)}{\partial \theta} \approx \frac{\partial l(\theta|x)}{\partial \theta}\Big|_{\theta=\hat{\theta}} + \frac{\partial^2 l(\theta|x)}{\partial \theta \partial \theta^T}\Big|_{\theta=\hat{\theta}}(\theta - \hat{\theta}) = 0$$

e como o primeiro termo do segundo membro é igual a zero, obtemos

$$\hat{\theta} \approx \theta - \Big[\frac{\partial^2 l(\theta|x)}{\partial \theta \partial \theta^T}\Big|_{\theta=\hat{\theta}}\Big]^{-1} \frac{\partial l(\theta|x)}{\partial \theta}\Big|_{\theta=\hat{\theta}}$$

De modo geral podemos escrever

$$\theta^{(i+1)} \approx \theta^{(i)} - [H(\theta^{(i)})]^{-1} g(\theta^{(i)}),$$

em que $\theta^{(i)}$ é a aproximação do máximo na i-ésima iteração.

A sequência de iterações convirgirá para um ponto de máximo se $H(\overline{\theta}) < 0$, que acontecerá se a função a maximizar for convexa, o que pode não valer em geral. O procedimento não convirgirá se o hessiano calculado no ponto de máximo for singular.

Exemplo 1

Consideremos a função

$$f(\theta) = \theta^3 - 3\theta^2 + 1,$$

que tem um ponto de máximo na origem e um ponto de mínimo em $\theta = 2$. Nesse caso,

i) $g(\theta) = 3\theta(\theta - 2)$

ii) $H(\theta) = 6(\theta - 1)$

O valor máximo é 1 e o valor mínimo é -3. Inicializemos o algoritmo com $\theta^{(0)} = 1,5$, para determinar o ponto de mínimo. Então, $g(1,5) = -2,25$ e $H(1,5) = 3$, de modo que na primeira iteração,

$$\theta^1 \approx 1,5 + \frac{2,25}{3} = 2,25,$$

continuando as iterações, obtemos $\theta^{(2)} = 2,025$, $\theta^{(3)} = 2,0003$, indicando a convergência para 2.

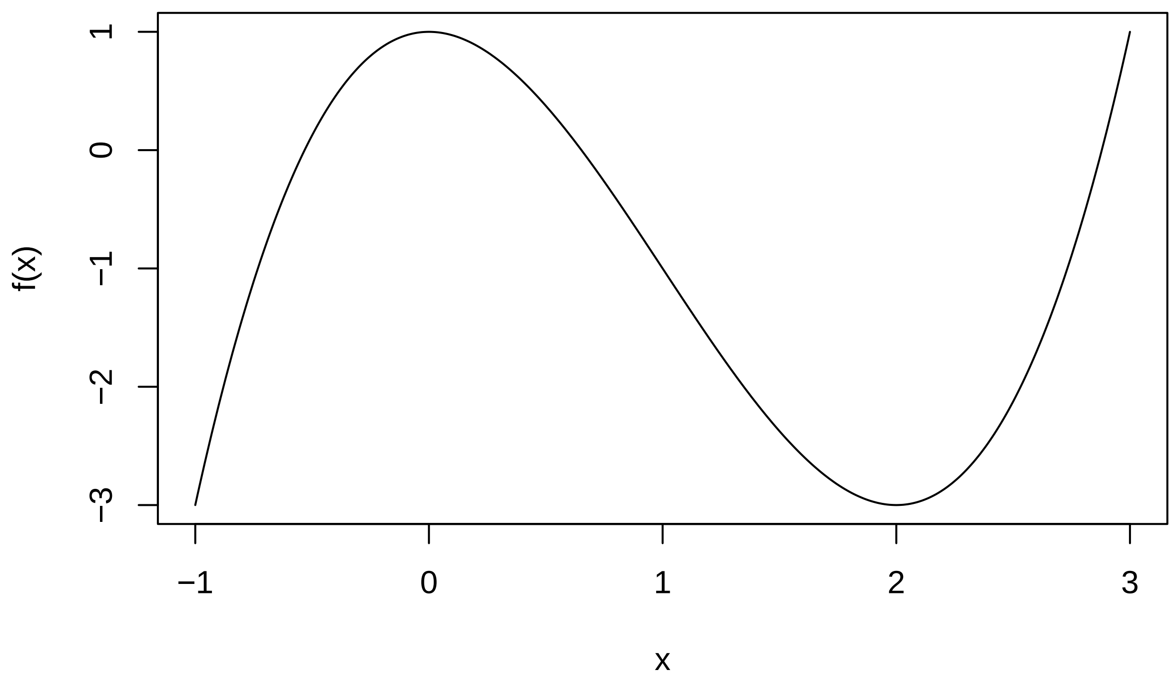
Se começarmos com $\theta^0 = 0,5$, na primeira iteração obtemos $\theta^{(1)} = -0,25$, mostrando que, como $H(0,5) < 0$, a primeira iteração direciona o estimador para o ponto de máximo.

```
f <- function(theta) theta^3 - 3*theta^2 +1

g <-function(theta) 3*theta^2 - 6*theta

H <- function(theta) 6*theta - 6


x <- seq(-1, 3, 0.01)
plot(x, f(x), type="l")
```




```
theta <- numeric()
dif_ <- 1
```

```r
epsilon <- 0.001
cont <- 1
#theta[1] <- 1.5
theta[1] <- 0.5
while(abs(dif_)>= epsilon){
  cat("cont+1=", cont+1, "\n")
  theta[cont+1] <- theta[cont] - ((H(theta[cont]))^-1)*g(theta[cont])
  cat("theta[", cont+1, "]=", theta[cont+1], "\n")
  dif_ <- theta[cont+1]-theta[cont]
  cat("dif=", dif_, "\n")
  cont <- cont+1
}
```

```
## cont+1= 2
## theta[ 2 ]= -0.25
## dif= -0.75
## cont+1= 3
## theta[ 3 ]= -0.025
## dif= 0.225
## cont+1= 4
## theta[ 4 ]= -0.000304878
## dif= 0.02469512
## cont+1= 5
## theta[ 5 ]= -4.646115e-08
## dif= 0.0003048316
```

```r
theta
```

```
## [1]  5.000000e-01 -2.500000e-01 -2.500000e-02 -3.048780e-04 -4.646115e-08
```

Exemplo 2

Agora vamos utilizar o processo de otimização em inferência. Para isso, vamos inicialmente gerar uma amostra aleatória da distribuição Weibull(α, γ) e depois realizar a otimização numérica da função log-verossimilhança para obter as estimativas de máxima verossimilhança.

A função densidade da distribuição Weibull(α, γ) é dada por

$$f(x) = \frac{\gamma}{\alpha^\gamma} x^{\gamma-1} \exp\Big[-\Big(\frac{x}{\alpha}\Big)^\gamma\Big],$$

sendo que $\alpha > 0, \gamma > 0$ e $x > 0$.

Com tal parametrização, tem-se que $E(X) = \alpha\Gamma(1 + 1/\gamma)$.

Para realizar o exemplo, vamos considerar $\alpha = 2$ e $\gamma = 0.5$.

```r
set.seed(2023)
n <- 1000
alpha <- 2
gama <- 0.5

x<- rweibull(n, shape =gama , scale = alpha)

summary(x)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##    0.0000    0.1802    1.1010    3.9462    4.1123 125.5789
```

```
# o valor esperado e
alpha*gamma(1+1/gama)
```

```
## [1] 4
```

Para obter a estimativa para α e γ, precisamos seguir os seguintes passos

1- Obter a função de verossimilhança;

2- Calcular a função log-verossmilhança;

3- Calcular o vetor escore;

4- Obter a matriz de informação observada;

5- Implementar o método de Newton-Raphson.

Para uma amostra de tamanho n da distribuição Weibull(α, γ), temos que

$$L(\alpha,\gamma) = \prod_{i=1}^{n} f(x_i) = \prod_{i=1}^{n} \frac{\gamma}{\alpha^{\gamma}} x_i^{\gamma-1} \exp\Big\{ -\Big(\frac{x_i}{\alpha}\Big)^{\gamma}\Big\},$$

a função log-verossimilhança é dada por

$$l(\alpha,\gamma) = log(L(\alpha,\gamma)) = nlog(\gamma) - n\gamma\log(\alpha) + (\gamma-1)\sum log(x_i) - \sum\Big(\frac{x_i}{\alpha}\Big)^{\gamma},$$

as derivadas de primeira ordem são dadas por

$$\frac{\partial l(\alpha,\gamma)}{\partial\alpha} = -\frac{n\gamma}{\alpha} + \frac{\gamma}{\alpha^{\gamma+1}}\sum x_i^{\gamma}$$

$$\frac{\partial l(\alpha,\gamma)}{\partial\gamma} = \frac{n}{\gamma} - nlog(\alpha) + \sum log(x_i) - \sum\Big(\frac{x_i}{\alpha}\Big)^{\gamma} log(x_i/\alpha),$$

as derivadas de segunda ordem são dadas por

$$\frac{\partial^2 l(\alpha,\gamma)}{\partial\alpha^2} = \frac{n\gamma}{\alpha^2} + \gamma(-\gamma-1)\alpha^{-\gamma-2}\sum x_i^{\gamma},$$

$$\frac{\partial^2 l(\alpha,\gamma)}{\partial\gamma^2} = -\frac{n}{\gamma} - \sum\Big(\frac{x_i}{\alpha}\Big)^{\gamma}[\log(x_i/\alpha)]^2,$$

$$\frac{\partial l(\alpha,\gamma)}{\partial\alpha\partial\gamma} = -\frac{n}{\alpha} + \frac{\gamma}{\alpha^{\gamma+1}}\sum x_i^{\gamma} log(x_i/\alpha) + \alpha^{-\gamma-1}\sum x_i^{\gamma}.$$

Após a realização dos cálculos, vamos realizar a implementação do método de Newton-Raphson no R.

```r
# vetor score

g <- function(x, theta){
  n <- length(x)
  alpha <- theta[1]
  gama <- theta[2]
  part1 <- -(n*gama)/alpha + (gama/(alpha^(gama+1)))*sum(x^gama)
  part2 <- n/gama - n*log(alpha) + sum(log(x)) - sum(((x/alpha)^gama)*log(x/alpha))
  return( c(part1, part2))
}


g(x, c(1, 0.5))
```

```
## [1]  217.3866 -545.4308
```

```r
# matriz de informação observada

H <- function(x, theta){
  n <- length(x)
  alpha <- theta[1]
  gama <- theta[2]
  part1 <- (n*gama)/(alpha^2)+gama*(-gama-1)*alpha^(-gama-2)*sum(x^gama)
  part2 <- -n/alpha + (gama/(alpha^(gama+1)))*sum((x^gama)*log(x/alpha))+ alpha^(-gama-1)*sum(x^gama)
  part3 <- -n/(gama^2)-sum(((x/alpha)^gama)*(log(x/alpha))^2)
  h <- matrix(c(part1, part2, part2, part3), ncol=2, nrow=2, byrow=T)
  return(h)
}

H(x, c(1, 0.5))
```

```
##          [,1]       [,2]
## [1,] -576.080   1518.516
## [2,] 1518.516 -10737.908
```

```r
# metodo de newton-raphson

n.max <- 100
theta <- matrix(NA, nrow=2, ncol=n.max)
dif_ <- 1
epsilon <- 0.001
cont <- 1
theta[, 1] <- c(1, 2)

while(abs(dif_)>= epsilon & cont<=n.max){
  cat("cont+1=", cont+1, "\n")
  theta[ ,cont+1] <- theta[ ,cont] - solve(H(x, theta[ ,cont]))%*%g(x, theta[ ,cont])
  cat("theta[", cont+1, "]=", theta[ ,cont+1], "\n")
  dif_ <- sqrt(sum((theta[ ,cont+1]-theta[ ,cont])^2))
  cat("dif=", dif_, "\n")
  cont <- cont+1
}
```

```
## cont+1= 2
## theta[ 2 ]= 0.8781681 1.677699
## dif= 0.3445592
## cont+1= 3
## theta[ 3 ]= 0.7362294 1.334386
## dif= 0.3714973
## cont+1= 4
## theta[ 4 ]= 0.5547017 0.9573204
## dif= 0.4184863
## cont+1= 5
## theta[ 5 ]= 0.3563464 0.5738512
## dif= 0.4317331
## cont+1= 6
## theta[ 6 ]= 0.3312675 0.381048
## dif= 0.1944274
## cont+1= 7
## theta[ 7 ]= 0.5641077 0.4181372
## dif= 0.2357757
## cont+1= 8
## theta[ 8 ]= 0.9305036 0.4671185
## dif= 0.3696554
## cont+1= 9
## theta[ 9 ]= 1.378278 0.501388
## dif= 0.4490838
## cont+1= 10
## theta[ 10 ]= 1.790903 0.5153879
## dif= 0.412862
## cont+1= 11
## theta[ 11 ]= 2.041855 0.5178614
## dif= 0.2509642
## cont+1= 12
## theta[ 12 ]= 2.112259 0.5178338
## dif= 0.07040396
## cont+1= 13
## theta[ 13 ]= 2.116738 0.5178167
## dif= 0.004479602
## cont+1= 14
## theta[ 14 ]= 2.116755 0.5178166
## dif= 1.699994e-05
```

```
cont
```

```
## [1] 14
```

```
theta[, 1:cont]
```

```
##      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]
## [1,]    1 0.8781681 0.7362294 0.5547017 0.3563464 0.3312675 0.5641077 0.9305036
## [2,]    2 1.6776988 1.3343860 0.9573204 0.5738512 0.3810480 0.4181372 0.4671185
##          [,9]     [,10]     [,11]     [,12]     [,13]     [,14]
## [1,] 1.378278 1.7909026 2.0418546 2.1122585 2.1167381 2.1167551
## [2,] 0.501388 0.5153879 0.5178614 0.5178338 0.5178167 0.5178166
```

Exercícios

1) Gerar 100 valores da distribuição Poisson($\lambda = 3$). Utilizando o método de Newton-Rapson, como feito anteriormente, obtenha a estimativa de máxima verossimilhança do parâmetro λ.

2) Gerar 300 valores da distribuição Bernoulli($p = 0,3$). Utilizando o método de Newton-Rapson, como feito anteriormente, obtenha a estimativa de máxima verossimilhança do parâmetro p.

3) Gerar 100 valores da distribuição Exp($\lambda = 0.3$). Utilizando o método de Newton-Rapson, como feito anteriormente, obtenha a estimativa de máxima verossimilhança do parâmetro λ.

4) Gerar 100 valores da distribuição normal($\mu = 3, \sigma^2 = 4$). Utilizando o método de Newton-Rapson, como feito anteriormente, obtenha as estimativas de máxima verossimilhança dos parâmetros μ e σ^2.