Aula 16: Método Monte Carlo em Inferência

Prof. Dr. Eder Angelo Milani e Érika Soares Machado

28/06/2023

Estimação do Erro Quadrático Médio

Métodos de Monte Carlo podem ser aplicados para estimar o EQM de um estimador.

Recordando que o EQM de um estimador $\hat{\theta}$ para o parâmetro θ é definido por

$$EQM(\hat{\theta}) = E[(\hat{\theta} - \theta)^2].$$

Se m amostras aleatórios $x^{(1)}, ..., x^{(m)}$ são geradas da distribuição de X, então uma estimativa do EQM de $\theta = \theta(x_1, ..., x_n)$ é

$$\widehat{EQM} = \frac{1}{m}\sum_{j=1}^{m}(\hat{\theta}^{(j)} - \theta)^2$$

sendo $\hat{\theta}^{(j)} = \hat{\theta}(x^{(j)}) = \hat{\theta}(x^{(j)}, ..., x_n^{(j)})$.

Observação O EQM de um estimador $\hat{\theta}$ pode ser escrito como

$$EQM(\hat{\theta}) = Var(\hat{\theta}) + [V(\hat{\theta})]^2,$$

sendo que $Var(\hat{\theta})$ é a variância do estimador e $V(\hat{\theta})$ é o viés do estimador. Quando o estimador é não viesado, o EQM é igual a variância do estimador.

Exemplo: (Estimado o EQM de uma média aparada)

Uma média aparada é às vezes aplicada para estimar o centro de uma distribuição simétrica contínua que não é necessariamente normal. Neste exemplo, calculamos uma estimativa do EQM de uma média aparada. A vantagem da média aparada é o fato dela não ser influenciada por valores extremos, a ideia é combinar as vantagens da mediana e da média.

Suponha que $(X_1, ..., X_n)$ é uma amostra aleatória e $X_{(1)}, ..., X_{(n)}$ é a amostra ordenada correspondente. O k-ésimo nível da média aparada amostral é definida por

$$\bar{X}_{[-k]} = \frac{1}{n-2k}\sum_{i=k+1}^{n-k} X_{(i)}$$

Obter uma estimativa de Monte Carlo do $EQM(\bar{X}_{[-1]})$, ou seja, da média aparada de 1º nível, supondo que a distribuição da amostra é normal padrão.

Neste exemplo, o centro da distribuição é 0 e o parâmetro de interesse é $\theta = E[\bar{X}] = E[\bar{X}_{[-1]}] = 0$. Vamos denotar o primeiro nível da média aparada amostral por T. Uma estimativa de Monte Carlo de EQM(T) baseado em m réplicas pode ser obtido por:

1. Gerar as réplicas $T^j, j = 1, ..., m$, repetindo

a. gerar $x_1^{(j)}, ..., x_n^{(j)}$, iid da distribuição de X

b. ordenar $x_1^{(j)}, ..., x_n^{(j)}$ em ordem crescente e obter $x_{(1)}^{(j)} \leq x_{(2)}^{(j)} \leq ... \leq x_{(n)}^{(j)}$.

c. Calcular

$$T^{(j)} = \frac{1}{n-2} \sum_{i=2}^{n-1} x_{(i)}^{(j)}$$

2. Calcular

$$\widehat{EQM}(T) = \frac{1}{m} \sum_{j=1}^{m} (T^{(j)} - \theta)^2 = \frac{1}{m} \sum_{j=1}^{m} (T^{(j)})^2$$

Então, $T^{(1)}, \ldots, T^{(m)}$ são independentes e identicamente distribuídos de acordo com a distribuição amostral da média aparada de nível 1, para uma distribuição normal padrão. Calculamos a estimativa $E\hat{Q}M(T)$ do $EQM(T)$. Este procedimento pode ser calculado usando a seguinte rotina.

```
set.seed(2022)

n=20
m=1000
media=tmedia=numeric(m)
for(i in 1:m){
  x=sort(rnorm(n))
  tmedia[i]=sum(x[2:(n-1)])/(n-2)
  media[i]=mean(x)
}


## Para a média aparada

eqm=mean(tmedia^2)

cat("O erro quadrático médio da média aparada é", eqm, "\n")
```

```
## O erro quadrático médio da média aparada é 0.05177824
```

```
var_=sum((tmedia-mean(tmedia))^2)/m

cat("A variância da média aparada é", var_ , "\n")
```

```
## A variância da média aparada é 0.05176255
```

```r
vies=mean(tmedia)-0

cat("O viés da média aparada é ", vies, "\n")
```

```
## O viés da média aparada é  0.003961765
```

```r
cat("O viés ao quadrado da média aparada é ", vies^2, "\n")
```

```
## O viés ao quadrado da média aparada é  1.569558e-05
```

```r
## Para a média aritmética

cat("A média das estimativas da média é ", mean(media), "\n")
```

```
## A média das estimativas da média é  0.00291465
```

```r
eqm1=mean(media^2)

cat("O erro quadrático médio da média é", eqm1, "\n")
```

```
## O erro quadrático médio da média é 0.05026089
```

```r
eqm12=sum((media-mean(media))^2)/m

cat("A variância da média é", eqm12 , "\n")
```

```
## A variância da média é 0.0502524
```

Note que a média aparada é não viesada. A média aritmética também não é viesada, logo é possível comparar o EQM de ambos os estimadores, qual é o melhor estimador?

Estudo Monte Carlo

Sabemos que nem sempre conseguimos expressões fechadas para os estimadores de máxima verossimilhança. Sendo assim, método iterativos para a obtenção da estimativa de máxima verossimilhança são empregados.

Considere uma amostra aleatória $X_1, \ldots, X_n$, de tamanho n da distribuição Normal$(\mu, \sigma^2 = 1)$. Sabemos que o estimador de máxima verossimilhança de μ é a média amostral. Vamos considerar que não conhecemos este resultado e obteremos essa estimativa por meio da maximização da função de verossimilhança.

Sabemos que a densidade da distribuição Normal$(\mu, \sigma^2 = 1)$ é dada por

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{1}{2}(x-\mu)^2 \Big),$$

logo, a função de verossimilhança é dada por

$$L(\mu) = \prod_{i=1}^{n} f(x_i) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{1}{2}(x_i-\mu)^2 \Big),$$

a função log-verossimilhança é largamente utilizada nos métodos iterativos, sendo que neste caso é dada por

$$l(\mu) = \log(L(\mu)) = \sum_{i=1}^{n} \Big(-\frac{1}{2} log(2\pi) - \frac{1}{2}(x_i - \mu)^2 \Big),$$

veja no código a seguir um exemplo da utilização do método iterativo.

```
set.seed(2022)

x=rnorm(100,mean=4,sd=1)

cat("A estimativa de máxima verossimilhança de mu é ", mean(x), "\n")
```

```
## A estimativa de máxima verossimilhança de mu é  4.138746
```

```
n=length(x)

## função log-verossimilhança

log_vero=function(a){
mu=a[1]
aux=(-n/2)*log(2*pi)-(1/2)*sum((x-mu)^2)
return(-aux)  # o defult do optim minimiza
}

# verificação da função definida

log_vero(1)
```

```
## [1] 635.9894
```

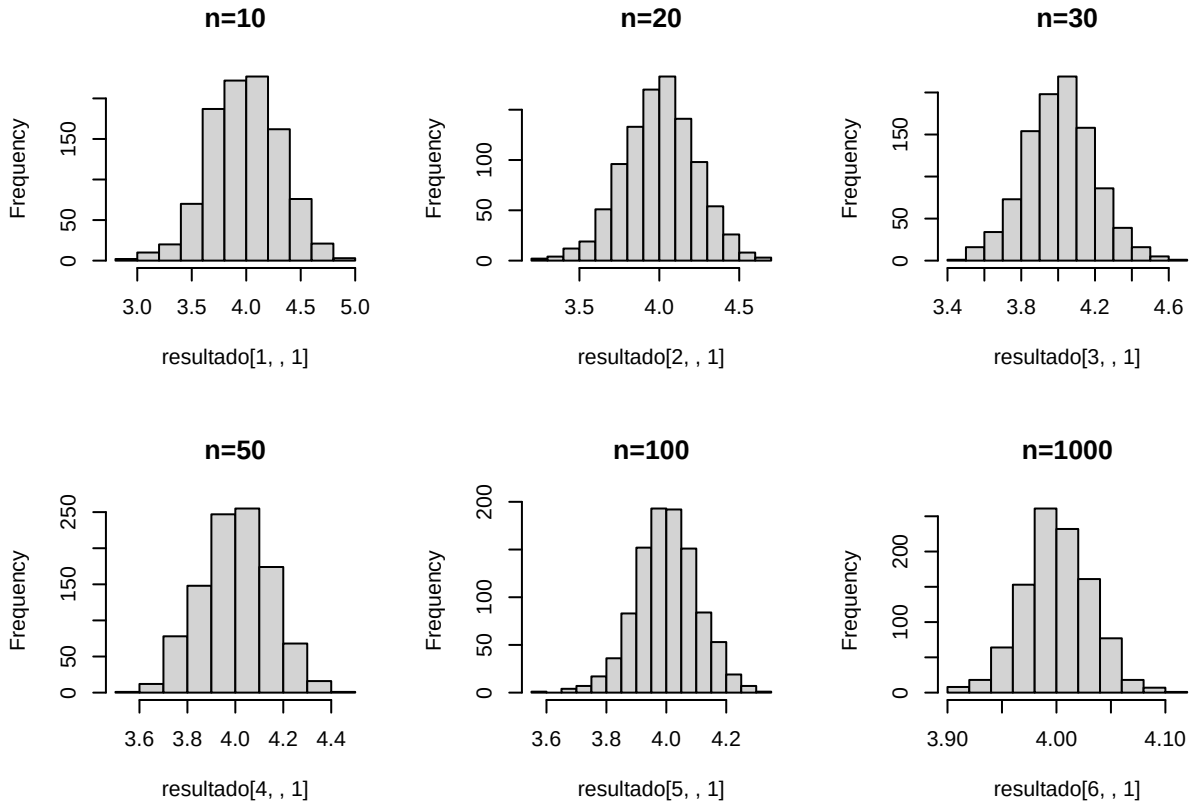
```
par=0
emv=optim(par,log_vero,method="BFGS",hessian=T)

cat("A estimativa de máxima verossimilhança de mu utilizando o método iterativo é ", emv$par, "\n")
```

```
## A estimativa de máxima verossimilhança de mu utilizando o método iterativo é  4.138746
```

A partir dos resultados acima, repetir 1000 vezes a geração da amostra e a estimação dos parâmetros, salvar a estimativa pontual. Fazer isso para tamanho amostral igual a 10, 20, 30, 50 e 100. Calcular a média, a variância e o erro quadrático médio das estimativas.

```
set.seed(2022)
n=c(10,20,30,50,100,1000)
resultado=array(NA, dim=c(6,1000,2))

log_vero=function(a,x){
mu=a[1]
n=length(x)
aux=(-n/2)*log(2*pi)-(1/2)*sum((x-mu)^2)
return(-aux)  # o defult do optim minimiza
}
par=0
```

```r
for(i in 1:6){
  for(j in 1:1000){
    x=rnorm(n[i],mean=4,sd=1)
    emv=optim(par,log_vero,x=x, method="BFGS",hessian=T)
    resultado[i,j,1]=emv$par
    resultado[i,j,2]=sqrt(solve(emv$hessian))
  }
}

# Análise para os diferentes valores de n

par(mfrow=c(2,3))

hist(resultado[1,,1], main="n=10")
par(new=F)
hist(resultado[2,,1], main="n=20")
par(new=F)
hist(resultado[3,,1], main="n=30")
par(new=F)
hist(resultado[4,,1], main="n=50")
par(new=F)
hist(resultado[5,,1], main="n=100")
par(new=F)
hist(resultado[6,,1], main="n=1000")
```



```r
# Análise da média das estimativas

mean(resultado[1,,1])
```

```
## [1] 3.991478
```

```r
mean(resultado[2,,1])
```

```
## [1] 4.003334
```

```r
mean(resultado[3,,1])
```

```
## [1] 4.008579
```

```r
mean(resultado[4,,1])
```

```
## [1] 4.003195
```

```r
mean(resultado[5,,1])
```

```
## [1] 4.001231
```

```r
mean(resultado[6,,1])
```

```
## [1] 4.000625
```

```r
# Análise da variância das estimativas


var(resultado[1,,1])
```

```
## [1] 0.1012607
```

```r
var(resultado[2,,1])
```

```
## [1] 0.0502265
```

```r
var(resultado[3,,1])
```

```
## [1] 0.03437652
```

```r
var(resultado[4,,1])
```

```
## [1] 0.02010596
```

```r
var(resultado[5,,1])
```

```
## [1] 0.01060121
```

```r
var(resultado[6,,1])
```

```
## [1] 0.0009650206
```

```r
# Análise do EQM

sum((resultado[1,,1]-mean(resultado[1,,1]))^2)/1000
```

```
## [1] 0.1011594
```

```r
sum((resultado[2,,1]-mean(resultado[2,,1]))^2)/1000
```

```
## [1] 0.05017627
```

```r
sum((resultado[3,,1]-mean(resultado[3,,1]))^2)/1000
```

```
## [1] 0.03434214
```

```r
sum((resultado[4,,1]-mean(resultado[4,,1]))^2)/1000
```

```
## [1] 0.02008586
```

```r
sum((resultado[5,,1]-mean(resultado[5,,1]))^2)/1000
```

```
## [1] 0.01059061
```

```r
sum((resultado[6,,1]-mean(resultado[6,,1]))^2)/1000
```

```
## [1] 0.0009640556
```

Exercício

1.Repetir o estudo Monte Carlos anterior, mas agora considerando amostras da distribuição

a. Normal(μ, σ^2);
b. Geométrica(p);
c. Weibull(α, β)
d. Poisson(λ)