

# Atividade 6 - Geração de Dados de Modelo de Regressão Linear

Jaqueline Lamas da Silva

2024

## Sumário

<b>Simulação</b>	<b>2</b>
<b>Distribuições dos coeficientes estimados</b>	<b>3</b>
<b>Intervalos empíricos e cobertura:</b>	<b>5</b>

## Simulação

$$Y_i = 150 - 4x_{i1} + 2.5x_{i2} + \varepsilon_i$$
$$\varepsilon \sim N(0, (16)^2)$$

```
x1.val<-seq(36,86,length.out=100)
x2.val<-seq(-56,456, length.out=100)

x1<-sample(x1.val, size=50, replace = F)
x2<-sample(x2.val, size=50, replace = F)
X<-matrix(c(rep(1,50),x1,x2), byrow = F, ncol=3)
C<-solve(t(X)%*%X)
c.mm<-diag(C)
beta.verdadeiro<-c(150,-4,2.5)

simulaRegressao<-function(x,x1,x2, b0=150,b1=-4,b2=2.5)
{
  erro<-rnorm(50,mean = 0, sd=16)
  y<-b0+b1*x1+b2*x2 + erro
  ml1<-lm(y~x1+x2)
  X<-matrix(c(rep(1,50),x1,x2), byrow = F, ncol=3)
  C<-solve(t(X)%*%X)
  c.mm<-diag(C)
  b=ml1$coefficients
  beta.verdadeiro<-c(150,-4,2.5)
  # indicar se cada valor verdadeiro dos coeficientes
  # pertence a seu respectivo intervalo de confiança
  Ind<-numeric(4)
  for(i in 1:3)
  {
    li<-b[i]-qt(0.975,50-3)*sqrt((summary(ml1)$sigma^2)*c.mm[i])
    ls<-b[i]+qt(0.975,50-3)*sqrt((summary(ml1)$sigma^2)*c.mm[i])
    Ind[i]<- beta.verdadeiro[i]>=li && beta.verdadeiro[i]<=ls
  }
  li<-(47*summary(ml1)$sigma^2)/qchisq(0.975,50-3)
  ls<-(47*summary(ml1)$sigma^2)/qchisq(0.025,50-3)
  Ind[4]<- 16^2>=li && 16^2<=ls

  return(list(b=ml1$coefficients, sigma2=summary(ml1)$sigma^2,
             R=summary(ml1)$r.squared,Indicadora=Ind))
}

estimativas<-lapply(1:1000, FUN = simulaRegressao,x1=x1,x2=x2)
```

```

b<-do.call(rbind,apply(1:1000,FUN=function(x)
  {cbind.data.frame(estimativas[[x]]$b)}))
sigma2<-sapply(1:1000,FUN=function(x)
  {estimativas[[x]]$sigma2})
  
```

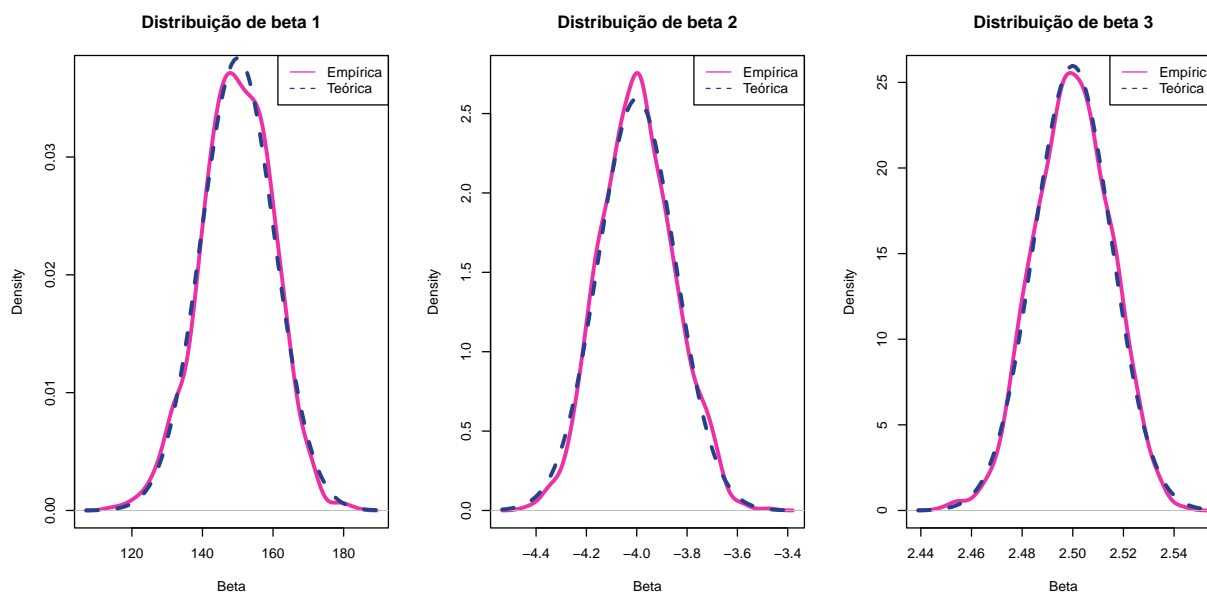
## Distribuições dos coeficientes estimados

```

par(mfrow=c(1,3))

for(i in 1:3)
{
  plot(density(b[,i]), col="maroon2", lwd=3,
       main=paste0("Distribuição de beta ",i), xlab="Beta")
  curve(dnorm(x,mean = beta.verdadeiro[i],
              sd=16*sqrt(c.mm[i])), from=min(density(b[,i])$x),
              to=max(density(b[,i])$x), add=T, col="royalblue4",lwd=3,lty=2)

  legend("topright",legend=c("Empírica","Teórica"),
        col = c("maroon2", "royalblue4"), lty = c(1,2))
}
  
```

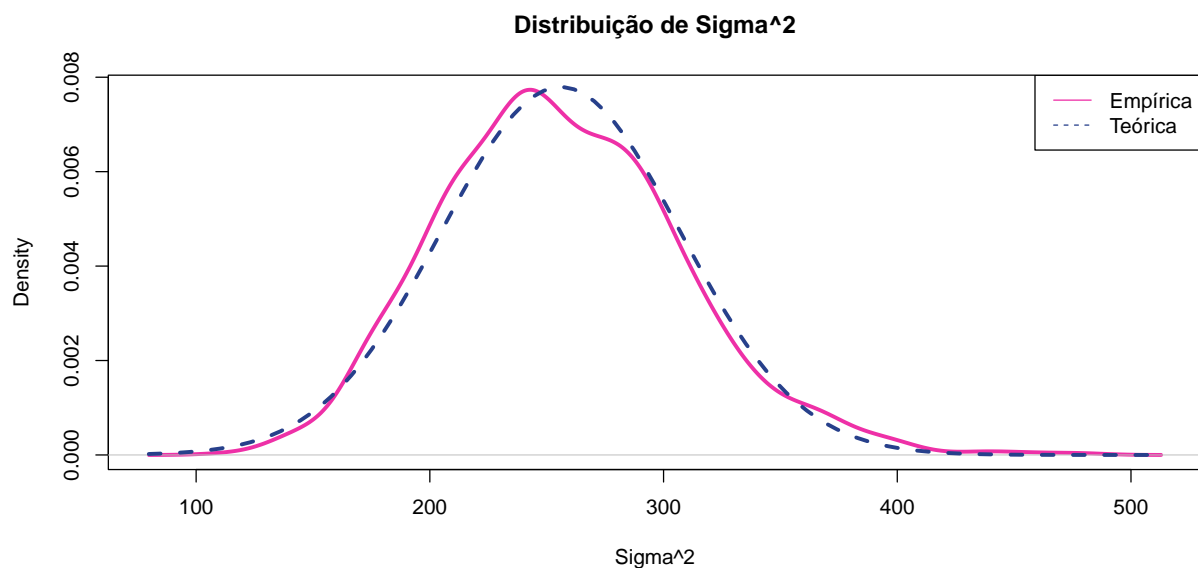


```
par(mfrow=c(1,1))

plot(density(sigma2), col="maroon2", lwd=3,
     main=paste0("Distribuição de Sigma^2"), xlab="Sigma^2")

curve(dnorm(x, mean=16^2, sd=sqrt((2*(16^2)^2)/50)),
      from=min(density(sigma2)$x), to=max(density(sigma2)$x),
      add=T, col="royalblue4", lwd=3, lty=2)

legend("topright", legend=c("Empírica", "Teórica"),
      col = c("maroon2", "royalblue4"), lty = c(1,2))
```



## Intervalos empíricos e cobertura:

```
# Neste objeto foi armazenado se o intervalo teorico inclui o verdadeiro parametro  
indicadora<-do.call(rbind,sapply(1:1000,FUN=function(x){  
  cbind.data.frame(estimativas[[x]]$Indicadora)))  
quantile(b[,1], probs = c(0.025,0.975))
```

```
##      2.5%      97.5%  
## 129.5725 168.5860
```

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

```
## [1] 0.957
```

- $IC(\beta_1, 95\%) = [129.573, 168.586]$ , cobertura do intervalo teórico foi 0.957.
- $IC(\beta_2, 95\%) = [-4.269, -3.705]$ , cobertura do intervalo teórico foi 0.964.
- $IC(\beta_3, 95\%) = [2.472, 2.528]$ , cobertura do intervalo teórico foi 0.956.
- $IC(\sigma^2, 95\%) = [167.076, 367.346]$ , cobertura do intervalo teórico foi 0.96.