# Curso R

# Reamostragem e Randomização

Alexandre Adalardo de Oliveira

Ecologia- IBUSP maio 2017

# Reamostragem e Permutação

# Introdução

Técnicas de simulação aleatórias baseadas em dados ou distribuições teóricas buscando soluções numéricas.

- teste de hipóteses
- medidas de precisão de estimativas
- otimizadores
- integração numérica
- algoritmos de amostragem

# Definições

## Monte Carlo

Técnicas de simulação buscando resultado numérico

- simulações aleatórias (numérico)
- distribuição conhecida (normal, poisson)
- MCMC "Markov Chain Monte Carlo"

## Reamostragem

Técnicas de reamostragem de dados

- precisão de estimativa (bootstrap, jackknife)
- teste de significância (reordenação, permutação)
- validação de modelos (subconjuntos)

# Definições

## Teste de Permutação

- reordenamento (rótulos) em todas as combinações possíveis (teste exato de Fisher)
- combinações possíveis de 10 valores:

```
factorial(10)
```

```
[1] 3628800
```

  - Teste de Monte Carlo: uma amostra das combinações

# Reposição

Bootstrap

Técnica de reamostragem de dados com reposição que permite a inferência sobre a precisão de uma estimativa.

Jackknife

Reamostragem de subamostra para o cálculo da precisão de uma estimativa.

# Implicações

Não assumem a distribuição de probabilidade teórica

## Vantagens

- poucas restrições (dados)
- intuitiva (conhecimento matemático)
- poucos pressupostos
- assintótica

## Desvantagens

- dificuldades computacionais
- resultado pode variar (estatítica de interesse)
- domínio de inferência restrito
- necessita cenário nulo adequado (complexo)

# Teste de Hipóteses

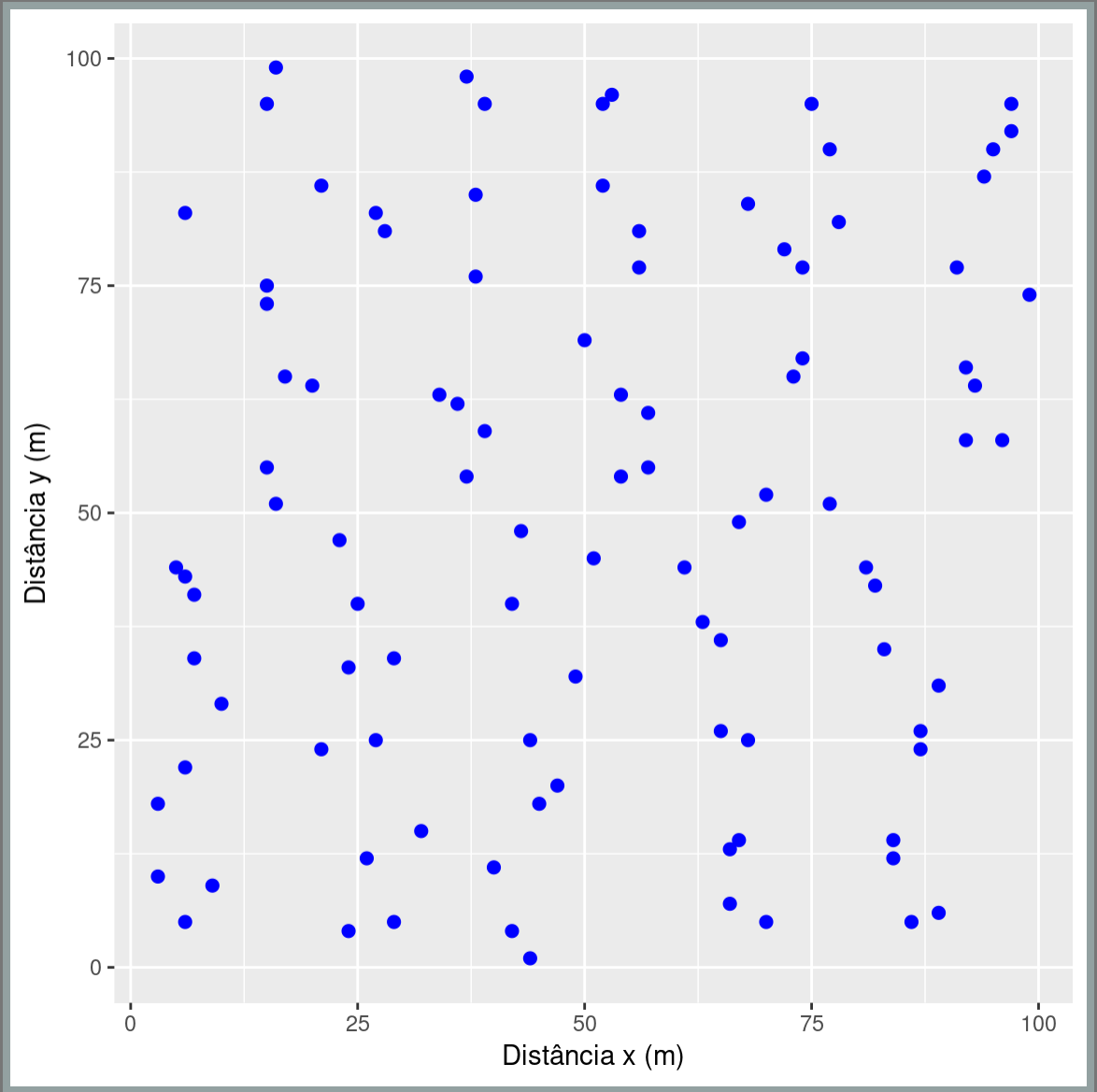
1. Definir a estatística de interesse (EI)
2. Estabelecer o cenário nulo
3. Reamostrar, reordenar ou simular o cenário nulo
4. Calcular a EI no cenário nulo
5. Criar a distribuição dos pseudovalores da EI
6. Posicionar o observado na distribuição dos pseudovalores
7. Calcular o p-valor


R

# Exemplos: teste de hipóteses

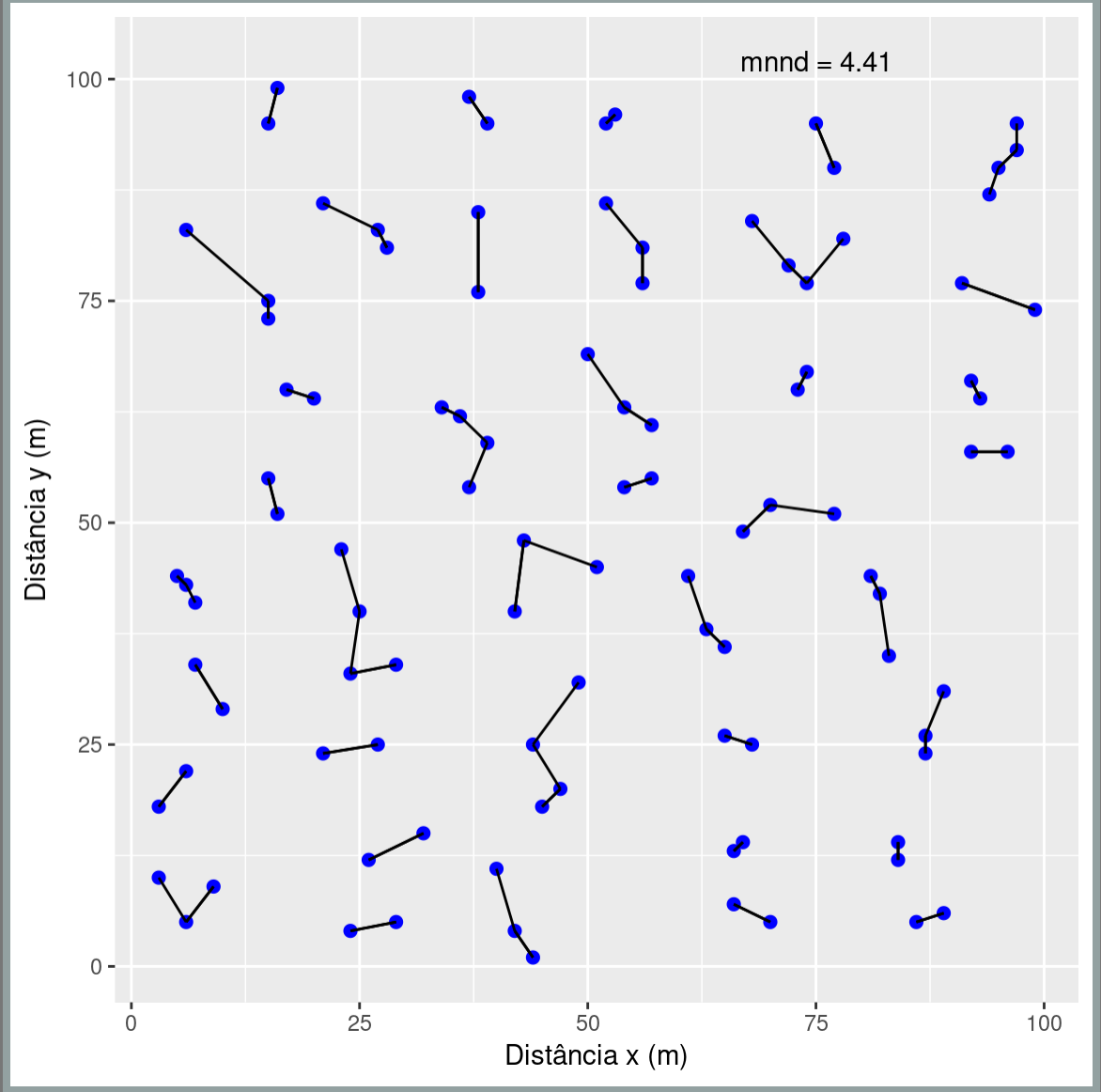
# Exemplo: Monte Carlo

Plantas estão distribuídas aleatoriamente em uma parcela?



# Estatística de interesse (EI)

- média da distância ao vizinho mais próximo



# Estatística de interesse (EI)

- média da distância ao vizinho mais próximo

```
xy[1:3,]
```

```
  xp yp
1  9  9
2  3 18
3  6 22
```

```
distmat <- as.matrix(dist(xy, diag = FAl
```

# Estatística de interesse (EI)

```
distmat[1:3, 1:3]
```

```
          1         2         3
1  0.00000 10.81665 13.34166
2 10.81665  0.00000  5.00000
3 13.34166  5.00000  0.00000
```

```
diag(distmat) <- NA
```

# Cálculo da estatística de interesse

```
diag(distmat) <- NA
nnd<- apply(distmat, 1, min, na.rm=TRUE)
tail(nnd)
```

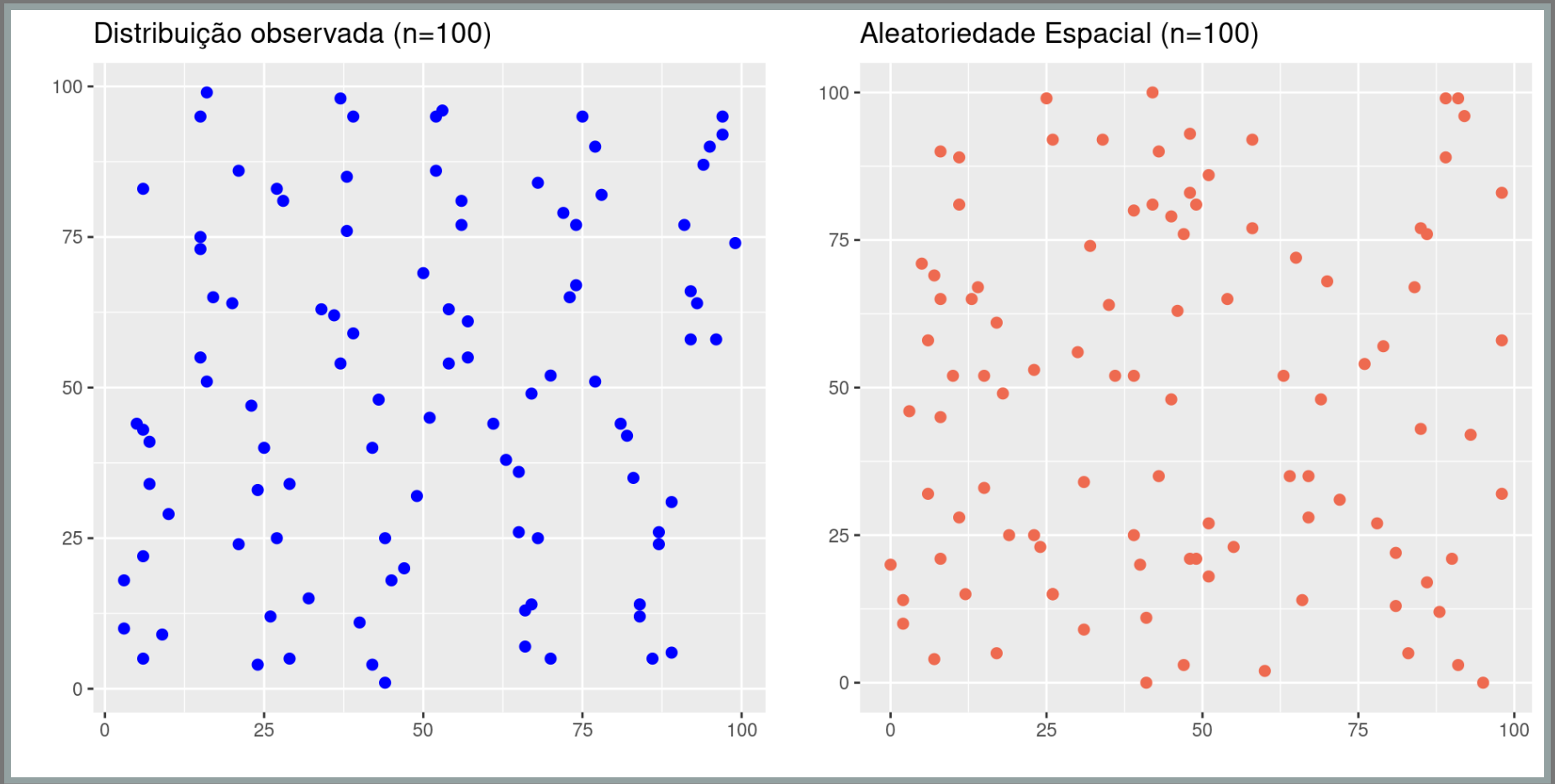
```
      95       96       97       98
2.236068 4.000000 2.236068 8.544004 2.82
```

```
mean(nnd)
```

```
[1] 4.414375
```

# Definir cenário nulo

- completa aleatoriedade espacial



# Simular o cenário nulo

```
rx <- round(runif(100,0,100),0)
ry <- round(runif(100,0,100),0)
rxy <- data.frame(rx, ry)
```

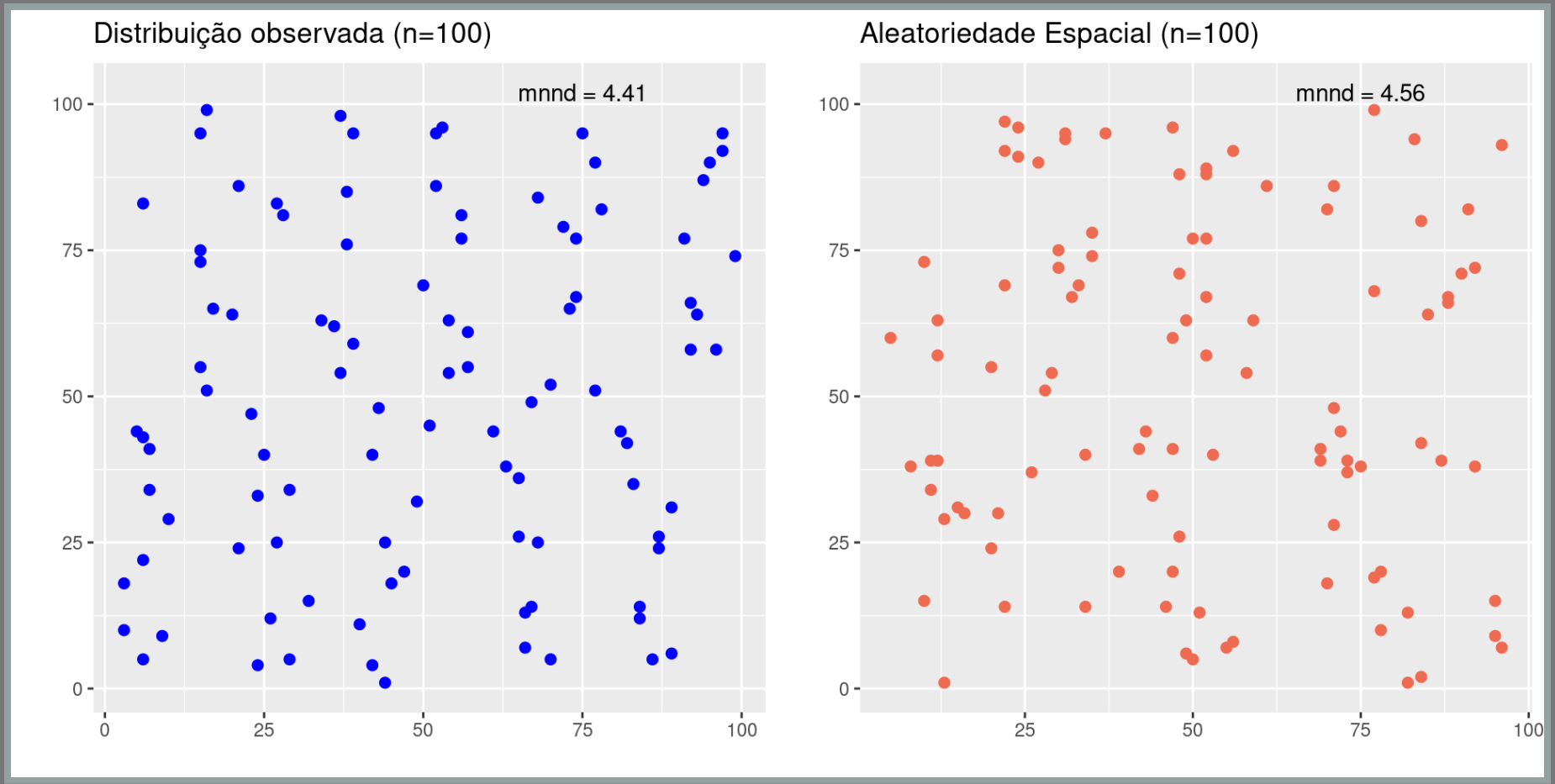
# Calcular a *E*/ no nulo

```
rdistmat <- as.matrix(dist(rxy, diag = F
diag(rdistmat) <- NA
rnnd <- apply(rdistmat, 1, min, na.rm=TR
mean(rnnd)
```

```
[1] 4.556871
```

# Calcular a *EI* no nulo

- completa aleatoriedade espacial



# Distribuição da *EI* no nulo

- definir o número de simulações
- criar o objeto de resultado das simulações

```
nsim = 1000
cnulo = rep(NA, nsim)
cnulo[1] <- mean(nnd)
cnulo[1:5]
```

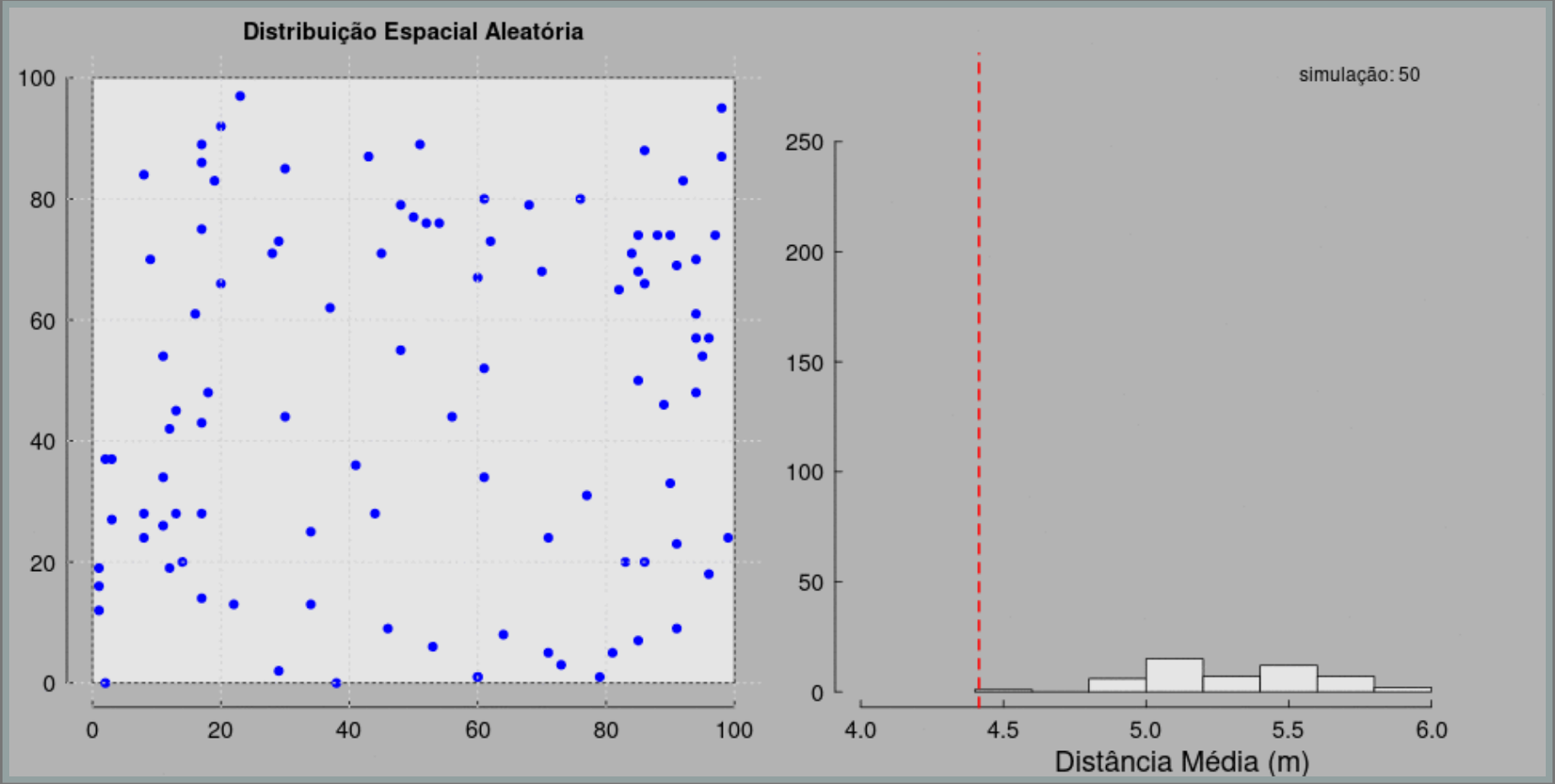
```
[1] 4.414375       NA       NA       NA
```

# Distribuição da *EI* no nulo

- criar o ciclo
- armazenar o resultado na posição

```
for(i in 2:nsim)
{
rx <- round(runif(100,0,100),0)
ry <- round(runif(100,0,100),0)
rxy <- data.frame(rx, ry)
rdistmat <- as.matrix(dist(rxy, diag = F
diag(rdistmat) <- NA
rnnd <- apply(rdistmat, 1, min, na.rm=TR
cnulo[i] <- mean(rnnd)
}
```

# Simulação


Distribuição Espacial Aleatória
100
80
60
40
20
0
0
20
40
60
80
100
simulação: 50
250
200
150
100
50
0
4.0
4.5
5.0
5.5
6.0
Distância Média (m)

# Calcular o *p-valor*

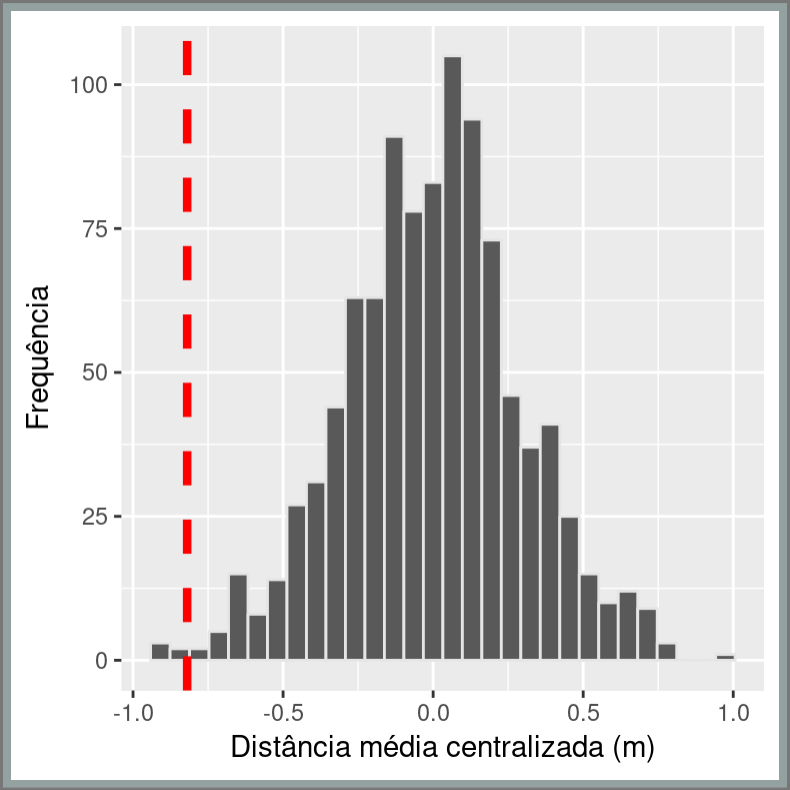
```
cnulo[1:10]
```

```
 [1] 4.414375 5.038223 5.132826 5.619759
 [8] 4.780985 5.035018 4.804148
```

```
ccnulo <- cnulo - mean(cnulo)
```

# Calcular o *p-valor*

```
ccnulo <- cnulo - mean(cnulo)
hist(ccnulo)
abline(v= ccnulo[1], lty=2, col="red")
```

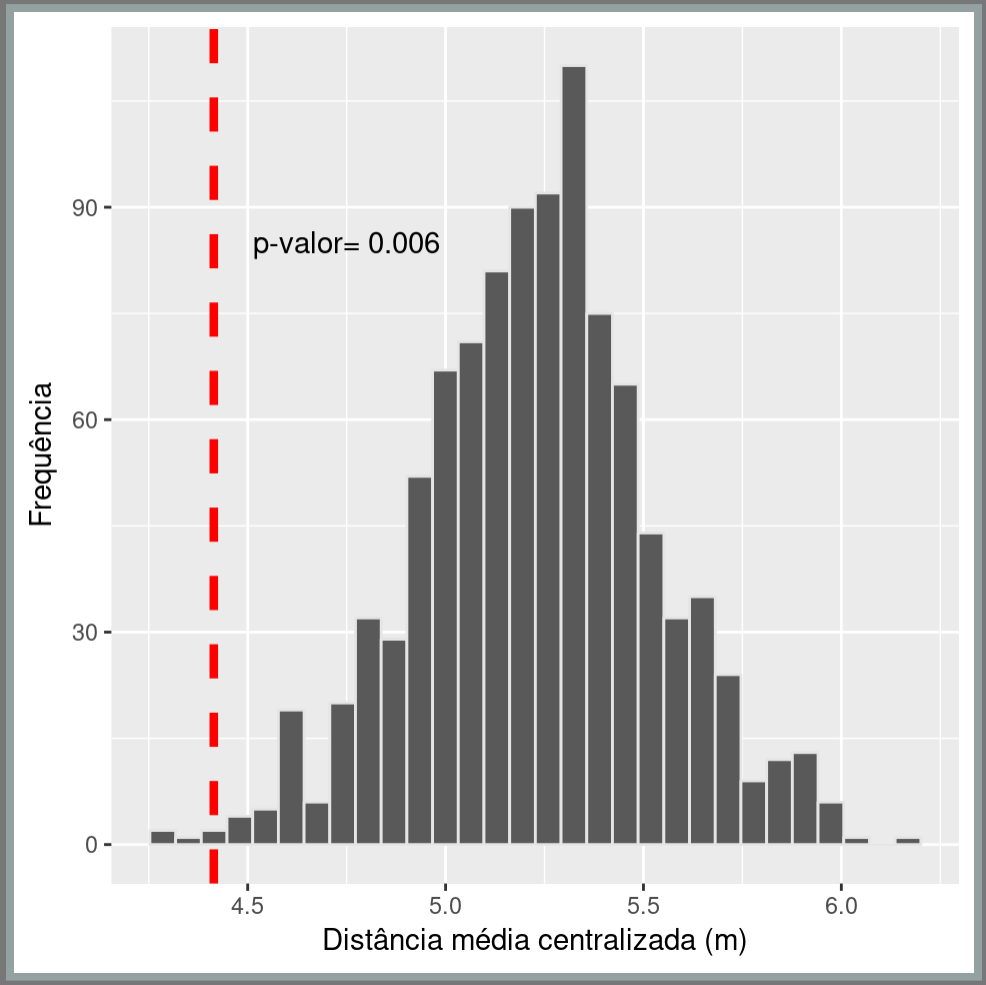


# Calcular o *p-valor*

```
(nmaior <- sum(abs(ccnulo) >= abs( ccnul
[1] 6
(pval <- nmaior/length(ccnulo))
[1] 0.006
```

# Resultado:

- pontos mais próximos do que o esperado pelo cenário de:
  - completa aleatoriedade espacial



# Completa Aleatoriedade Espacial

# ANOVA por reamostragem

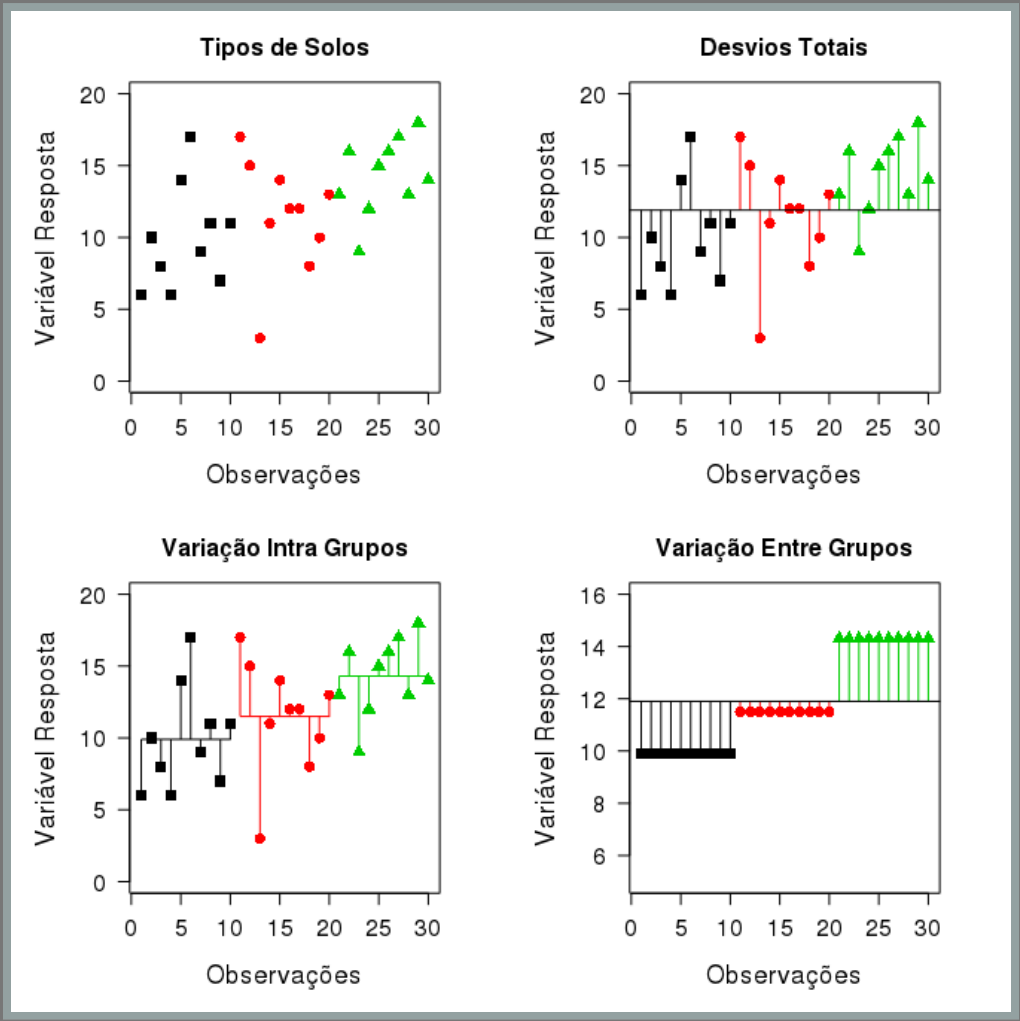
# Categórica com 3 níveis

```
are=c(6,10,8,6,14,17,9,11,7,11)
arg=c(17,15,3,11,14,12,12,8,10,13)
hum=c(13,16,9,12,15,16,17,13,18,14)
crop <- data.frame(solo = rep(c("are", "
head(crop)
```

```
  solo colhe
1  are     6
2  are    10
3  are     8
4  are     6
5  are    14
6  are    17
```

# Anova: partição da variação

$$F = \frac{\sigma^2_{entre}}{\sigma^2_{intra}}$$



# Estatística de interesse

Médias dos solos

```
msolo <- tapply(crop$colhe, crop$solo, m
```

Média Geral

```
mcolhe <- mean(crop$colhe)
```

# Estatística de interesse

## Soma das diferenças

```
msolo - mcolhe
```

```
 are  arg  hum
-2.0 -0.4  2.4
```

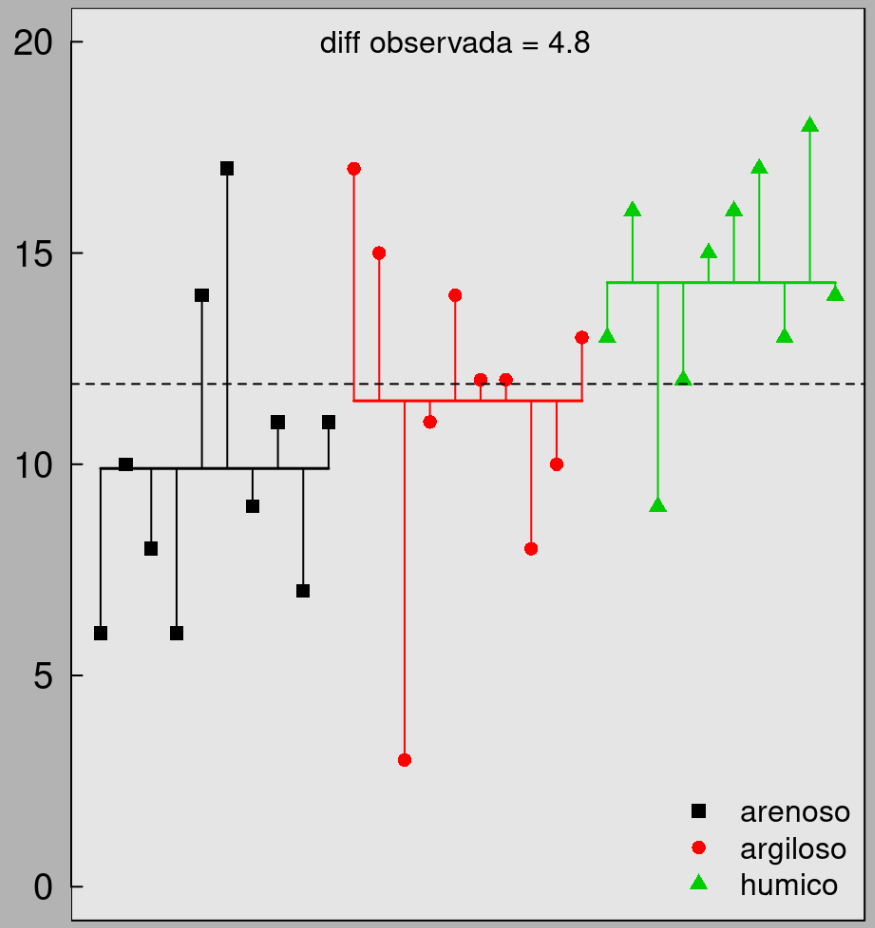
```
sum(msolo - mcolhe)
```

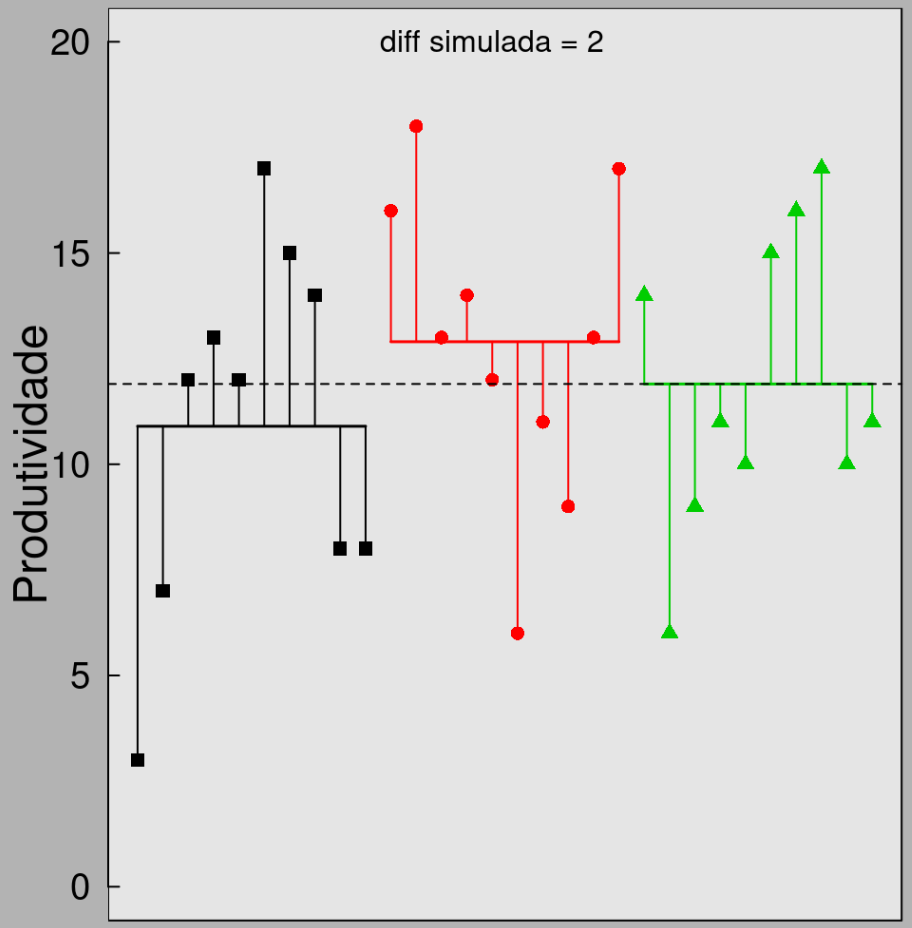
```
[1] 0
```

```
(difobs <- sum(abs(msolo-mcolhe)))
```

```
[1] 4.8
```

# Cenário Nulo



# Distribuição da *EI* no nulo

- definir o número de simulações
- criar o objeto de resultado das simulações

```
nsim = 1000
difnulo = rep(NA, nsim)
difnulo[1] <- difobs
difnulo[1:5]
```

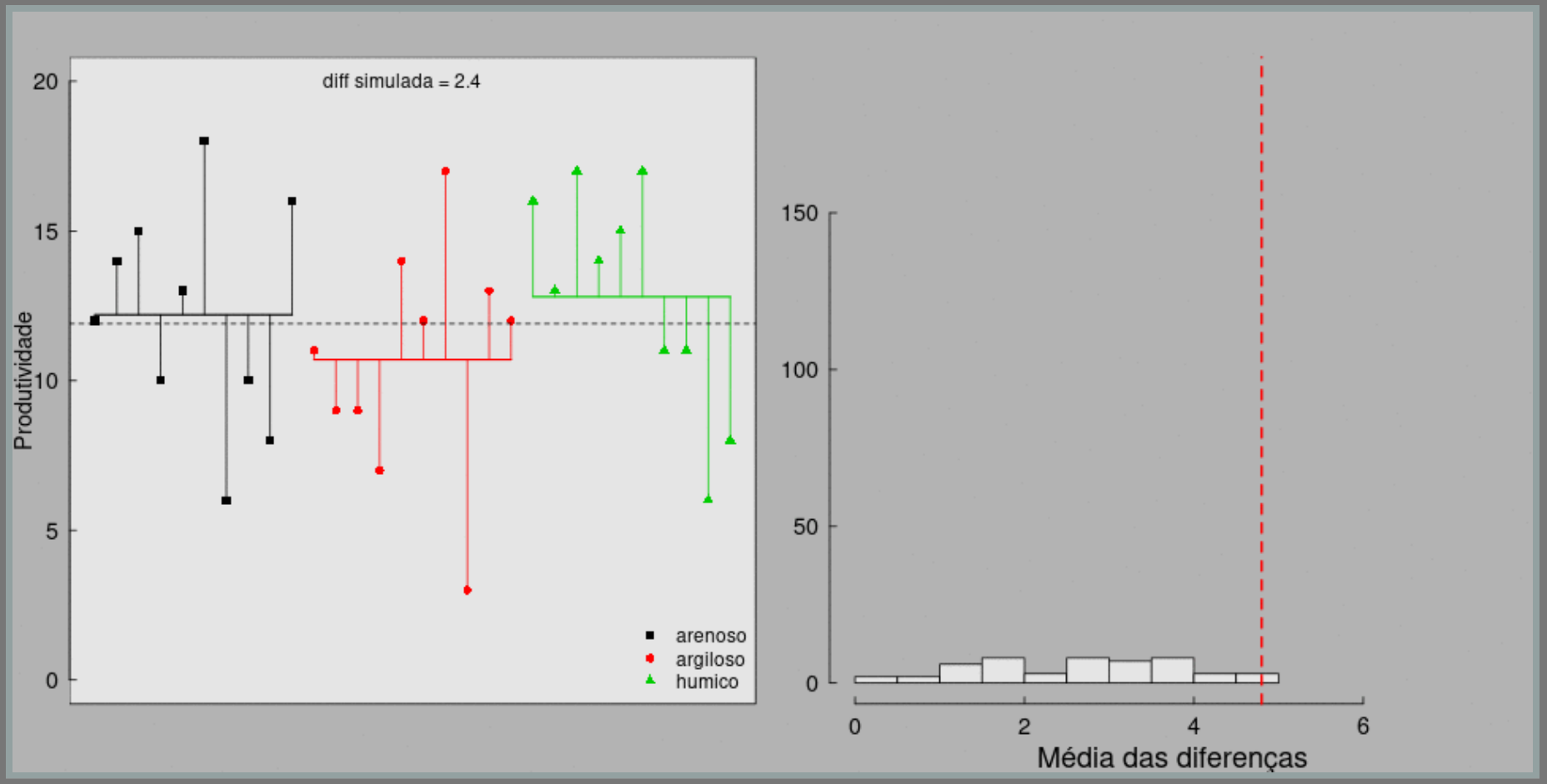
```
[1] 4.8  NA  NA  NA  NA
```

# Distribuição da *EI* no nulo

- criar o ciclo
- armazenar o resultado na posição

```
for(i in 2:nsim)
{
scolhe <- sample(crop$colhe)
smsolo <- tapply(scolhe, crop$solo, mean
difnulo[i] <- sum(abs(smsolo - mcolhe))
}
```

# Simulação


diff simulada = 2.4
Produtividade
20
15
10
5
0
arenoso
argiloso
humico
150
100
50
0
0
2
4
6
Média das diferenças

# Calcular o *p-valor*

```
difnulo[1:10]
```

```
 [1] 4.8 4.2 2.0 3.0 2.2 2.2 1.0 5.2 1.0
```

```
sum(difnulo >= difnulo[1])
```

```
[1] 28
```

```
sum(difnulo >= difnulo[1])/length(difnul
```

```
[1] 0.028
```

# Anova FIM

# REGRESSÃO por reamostragem

# Davis (1990). Appetite (15)13-21

```
library(car)
data(Davis)
kable(summary(Davis))
```

| sex | weight | height | repwt | repht |
|---|---|---|---|---|
| F:112 | Min. : 39.0 | Min. : 57.0 | Min. : 41.00 | Min. :148.0 |
| M: 88 | 1st Qu.: 55.0 | 1st Qu.:164.0 | 1st Qu.: 55.00 | 1st Qu.:160.5 |
| NA | Median : 63.0 | Median :169.5 | Median : 63.00 | Median :168.0 |
| NA | Mean : 65.8 | Mean :170.0 | Mean : 65.62 | Mean :168.5 |
| NA | 3rd Qu.: 74.0 | 3rd Qu.:177.2 | 3rd Qu.: 73.50 | 3rd Qu.:175.0 |
| NA | Max. :166.0 | Max. :197.0 | Max. :124.00 | Max. :200.0 |
| NA | NA | NA | NA's :17 | NA's :17 |

# Davis (1990). Appetite (15)13-21

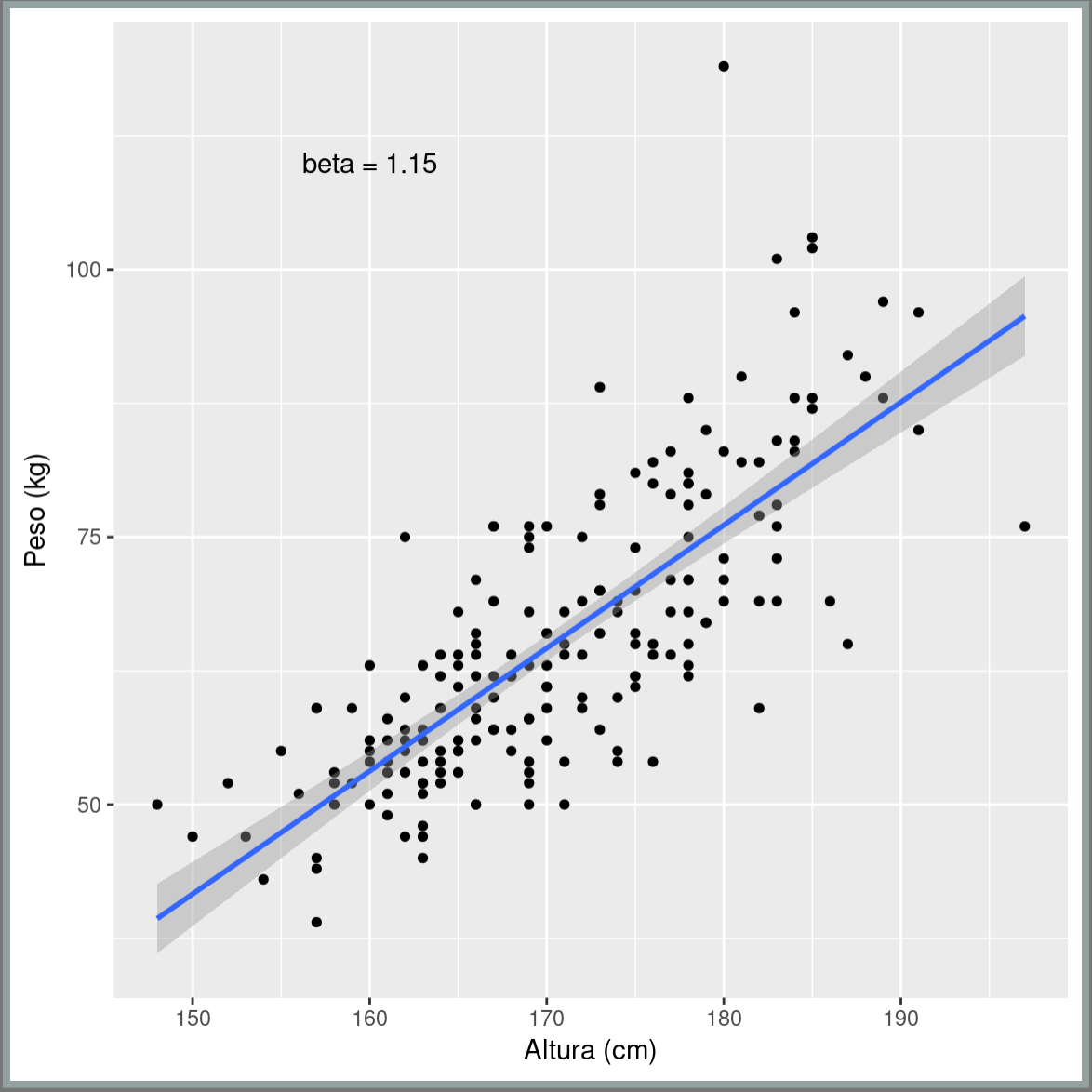
```
Davis[Davis$weight >150,]
```

```
   sex weight height repwt repht
12   F    166     57    56   163
```

```
Davis <- Davis[-12, 1:3]
str(Davis)
```

```
'data.frame':   199 obs. of  3 variables
 $ sex   : Factor w/ 2 levels "F","M": 2
 $ weight: int  77 58 53 68 59 76 76 69
 $ height: int  182 161 161 177 157 170
```

# REGRESSÃO: gráfico



# REGRESSÃO: estatística de interesse

- inclinação da reta ($\beta$)

$$\hat{\beta} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

# REGRESSÃO: cenário nulo

```
Davis$height[1:5]
```

```
[1] 182 161 161 177 157
```

```
sample(Davis$height)[1:5]
```

```
[1] 178 173 174 168 154
```

```
Davis$simh <-  sample(Davis$height)
kable(head(Davis))
```

| sex | weight | height | simh |
|---|---|---|---|
| M | 77 | 182 | 158 |
| F | 58 | 161 | 158 |
| F | 53 | 161 | 165 |
| M | 68 | 177 | 170 |
| F | 59 | 157 | 166 |
| M | 76 | 179 | 178 |

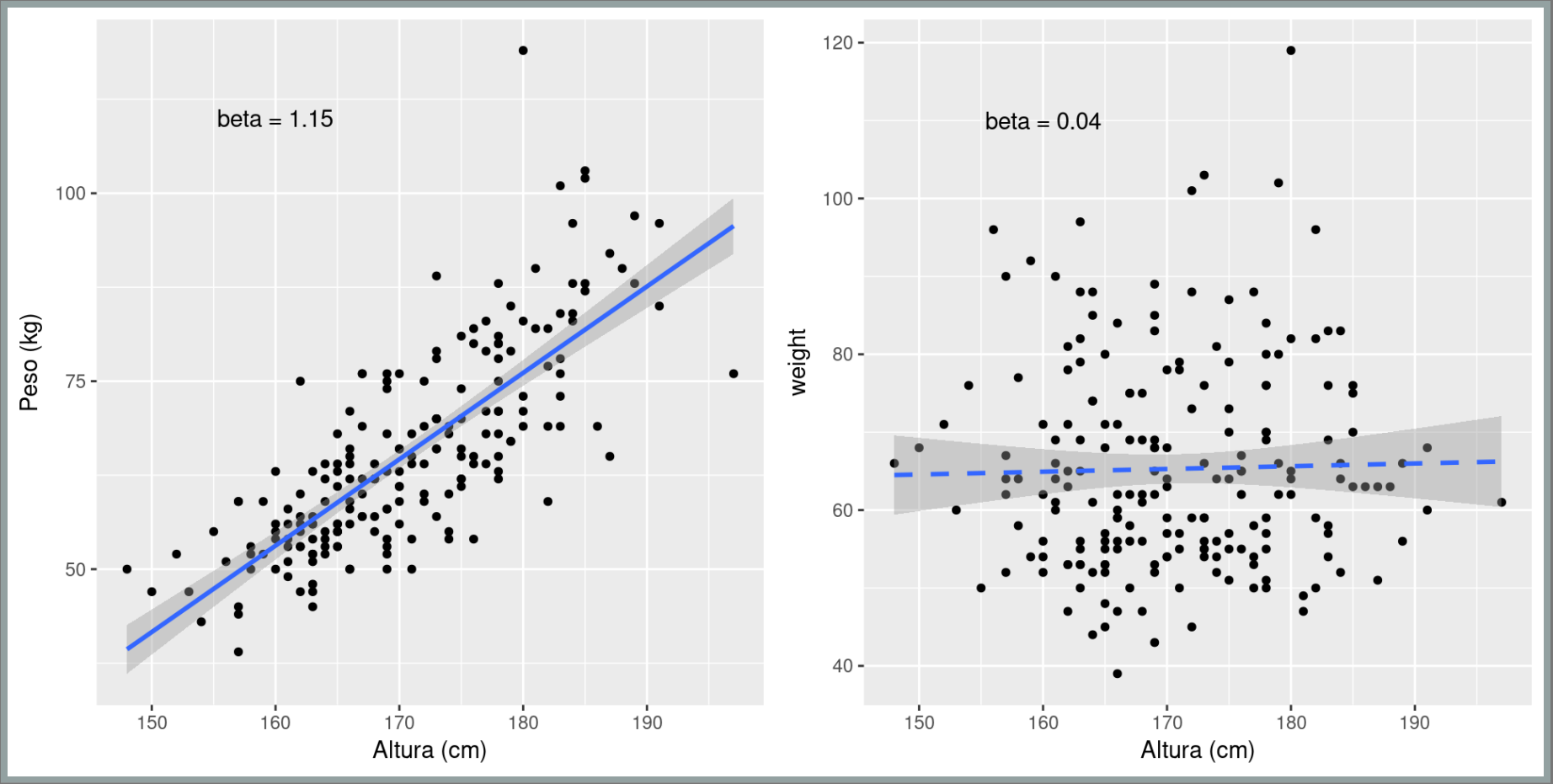
| | | | |
|---|---|---|---|
| M | | 78 | 176 | 178 |

| sex | weight | height | simh |
|---|---|---|---|
| ## Inc | linação: | coef() | |

```
coef(lm(weight ~ height, data =Davis))

(Intercept)      height
-130.746984    1.149222

coef(lm(weight ~ height, data =Davis))[2

  height
1.149222

(bobs <- round(coef(lm(weight ~ height,

height
  1.15

(bsim <- round(coef(lm(weight ~ simh, da
```

simh
0.04

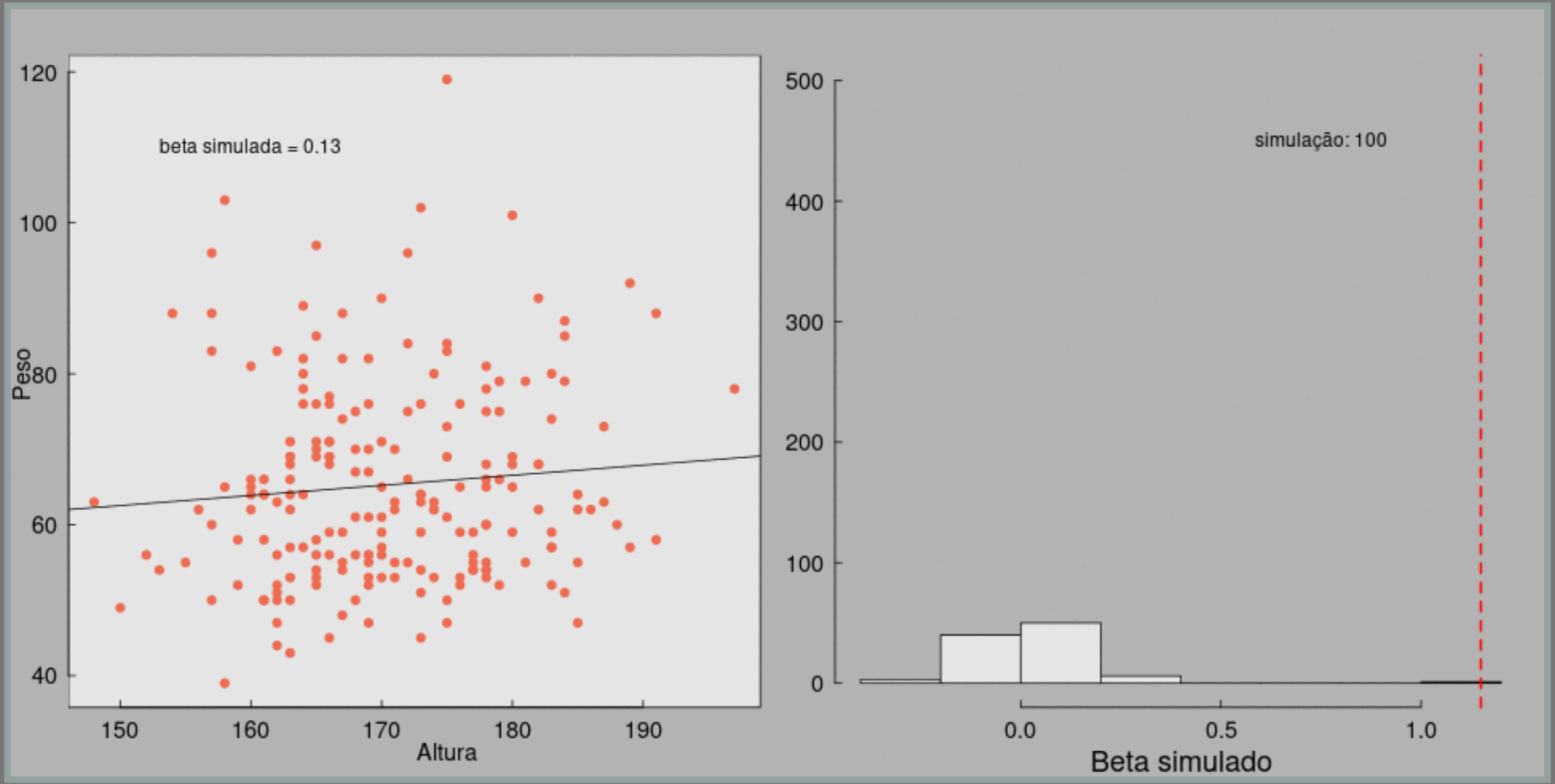
# Cenário Nulo



# Distribuição de pseudovalores

```
nSim = 1000
psv = rep(NA, nSim )
psv[1] <- coef(lm(weight ~ height, data=
for(i in 2:nSim)
{
psv[i] <- coef(lm(weight ~ sample(height
}
```

# Simulação


beta simulada = 0.13
Peso
Altura
120
100
80
60
40
150
160
170
180
190
simulação: 100
Beta simulado
500
400
300
200
100
0
0.0
0.5
1.0

# REGRESSÃO: p-valor

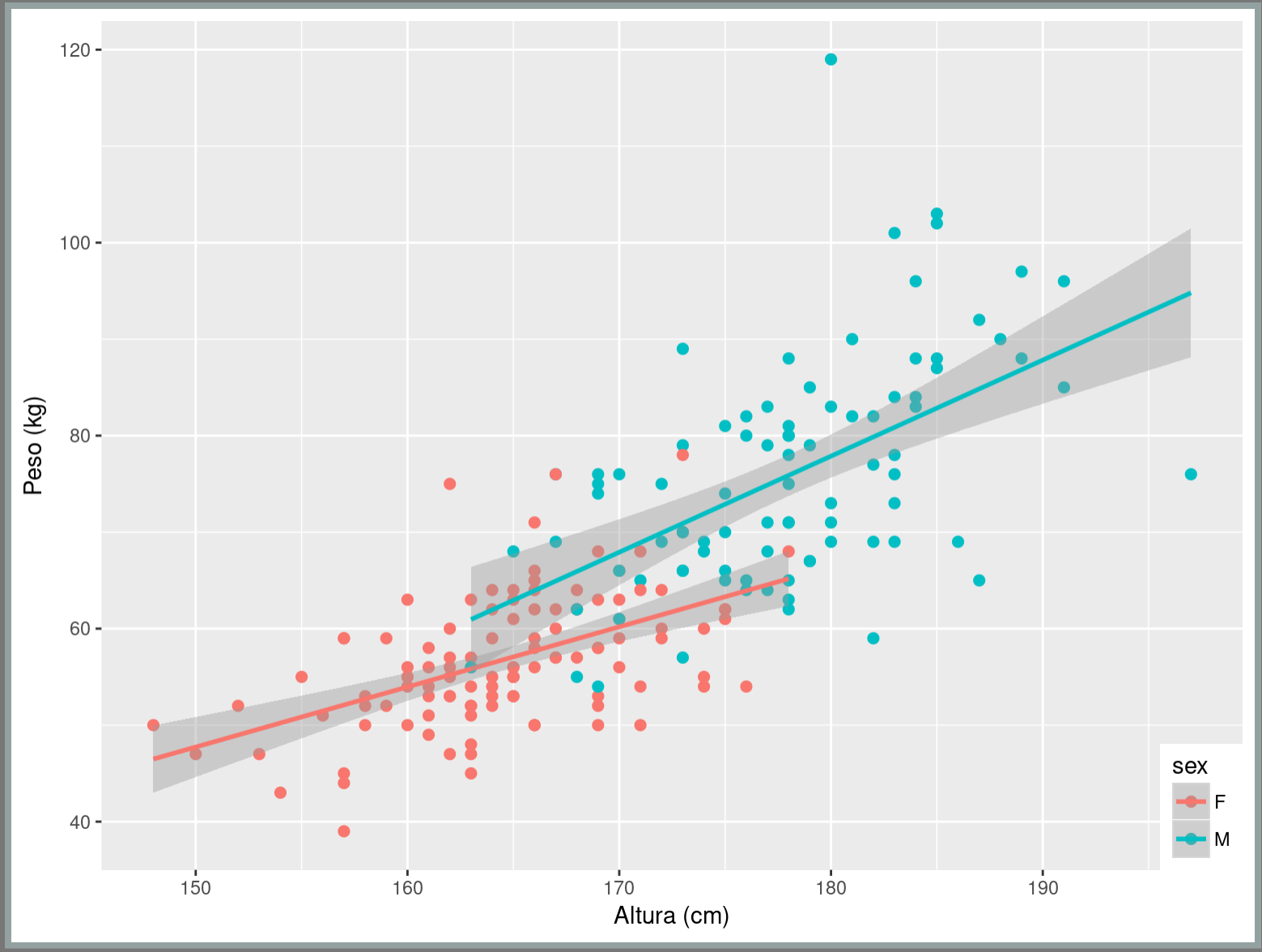
```
pvalor = sum(abs(psv) >= abs(psv[1]))/le
pvalor
```

```
[1] 0.001
```

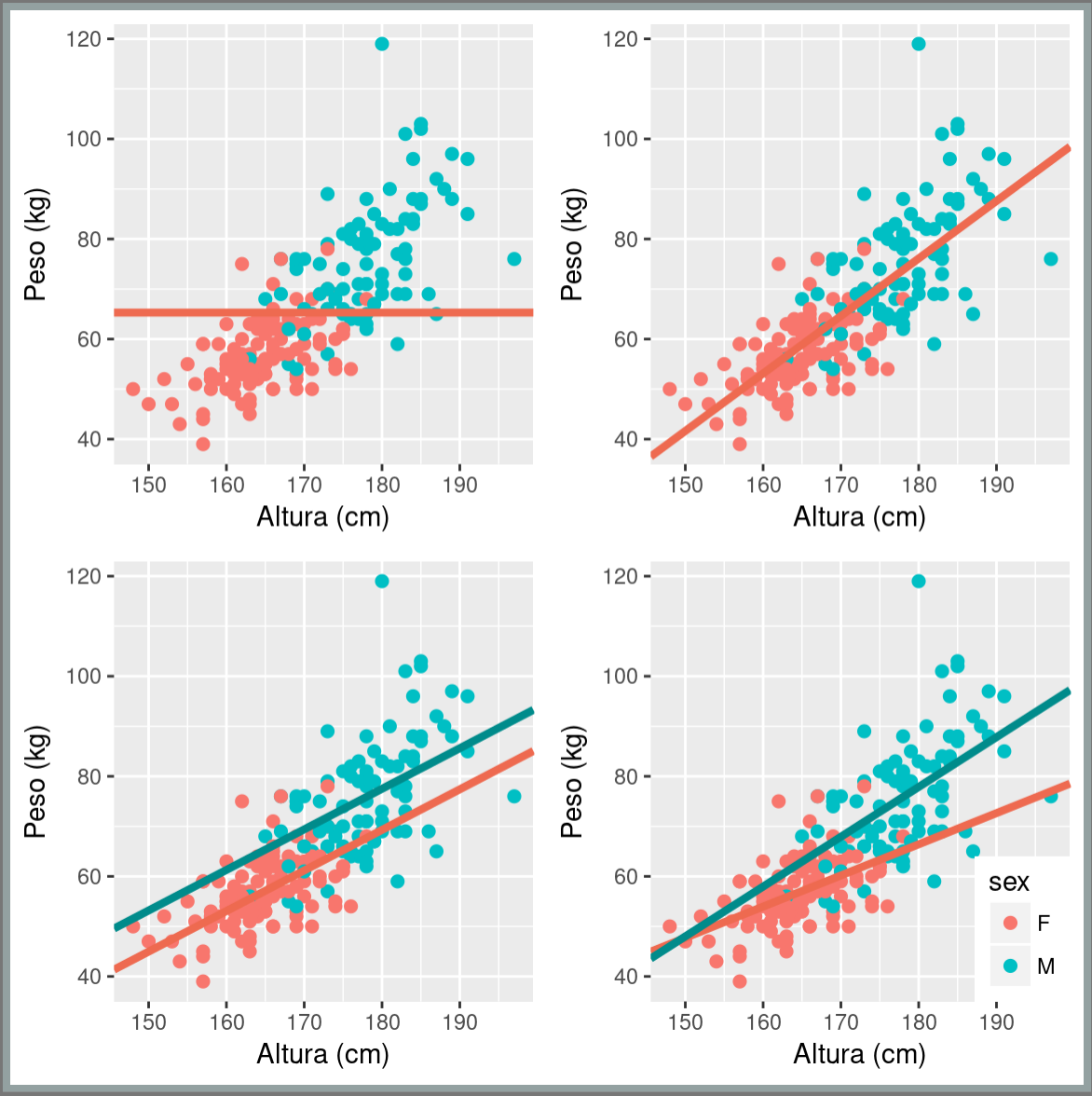
# Regressão: fim

# ANCOVA por reamostragem

# ANCOVA por reamostragem



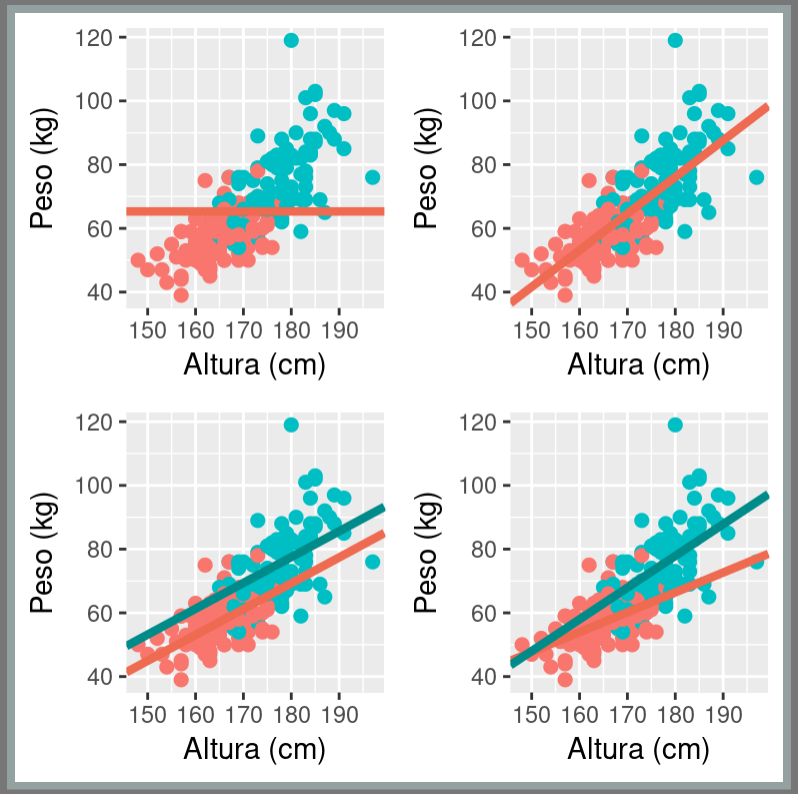
# Que perguntas podemos fazer?



# ANCOVA por reamostragem

## Que perguntas podemos fazer?

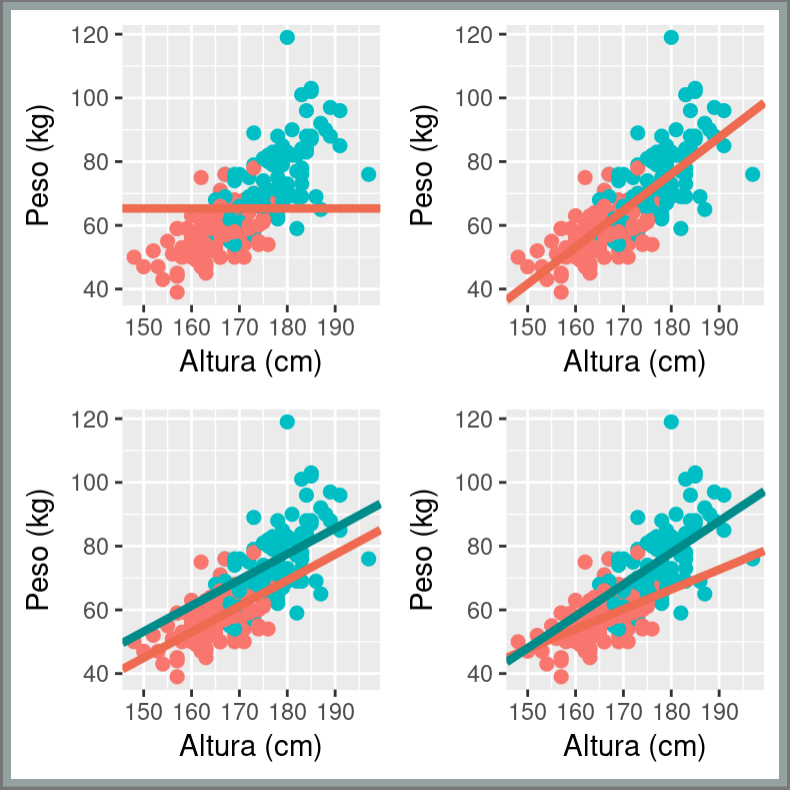
1. há relação entre peso e altura
2. a relação entre os sexos é a mesma, mas há um efeito de ser macho?
3. os sexos apresentam relações diferentes?


Peso (kg)
120
100
80
60
40
150 160 170 180 190
Altura (cm)
Peso (kg)
120
100
80
60
40
150 160 170 180 190
Altura (cm)
Peso (kg)
120
100
80
60
40
150 160 170 180 190
Altura (cm)
Peso (kg)
120
100
80
60
40
150 160 170 180 190
Altura (cm)

# ANCOVA por reamostragem

a relação é a mesma, mas há um efeito de ser macho:

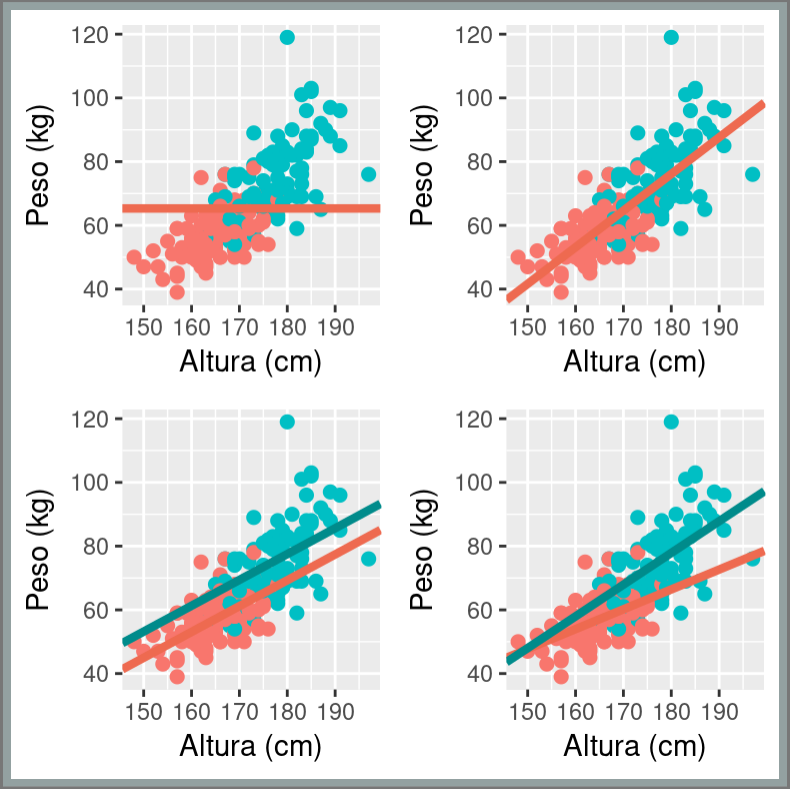
$$\hat{\alpha}_m \neq \hat{\alpha}_f$$

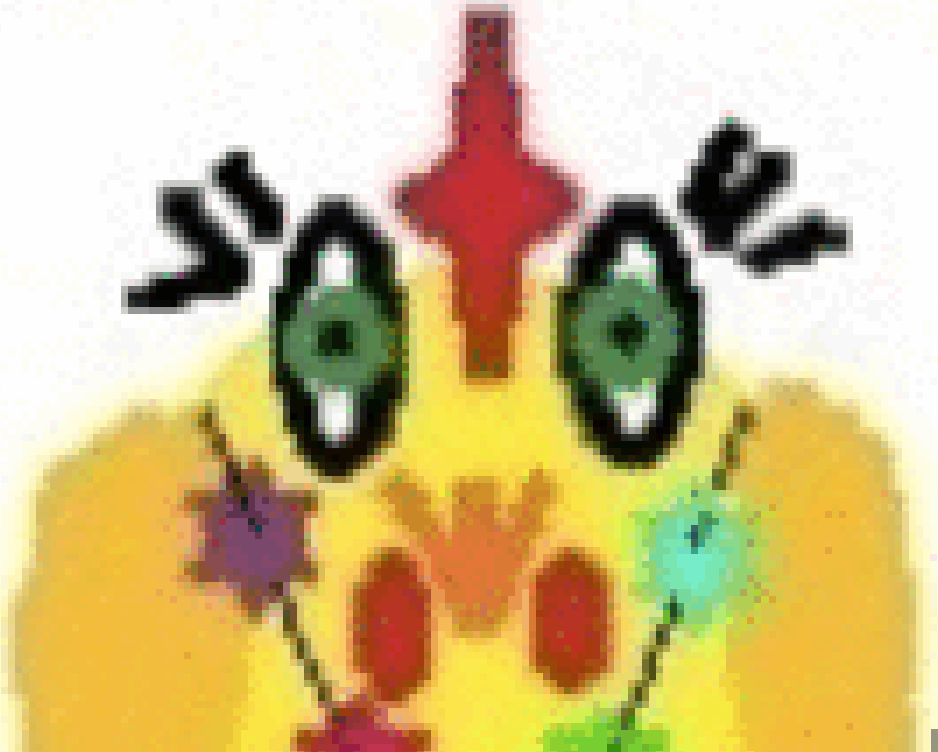


# ANCOVA por reamostragem

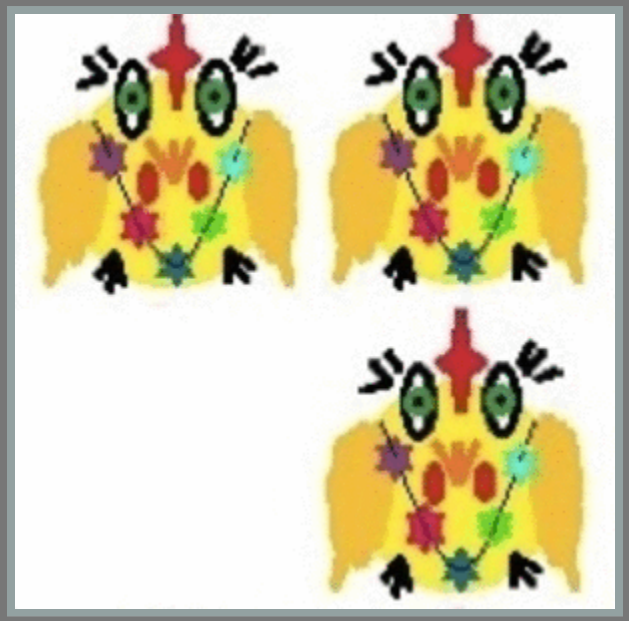
Os sexos apresentam relações diferentes:

$$\hat{\beta}_m \neq \hat{\beta}_f$$





# Rsampling



```
install.packages(c("Rsampling", "shiny",
library("shiny")
runApp("/home/aao/Ale2016/AleCursos/Plar
```

# Reamostragem com reposição

# sample(..., replace = TRUE)

```
ae <- letters[1:5]
sample(ae)
```

```
[1] "e" "d" "a" "b" "c"
```

```
sample(ae, replace=TRUE)
```

```
[1] "c" "d" "a" "d" "d"
```

```
sample(ae, size=10, replace=TRUE)
```

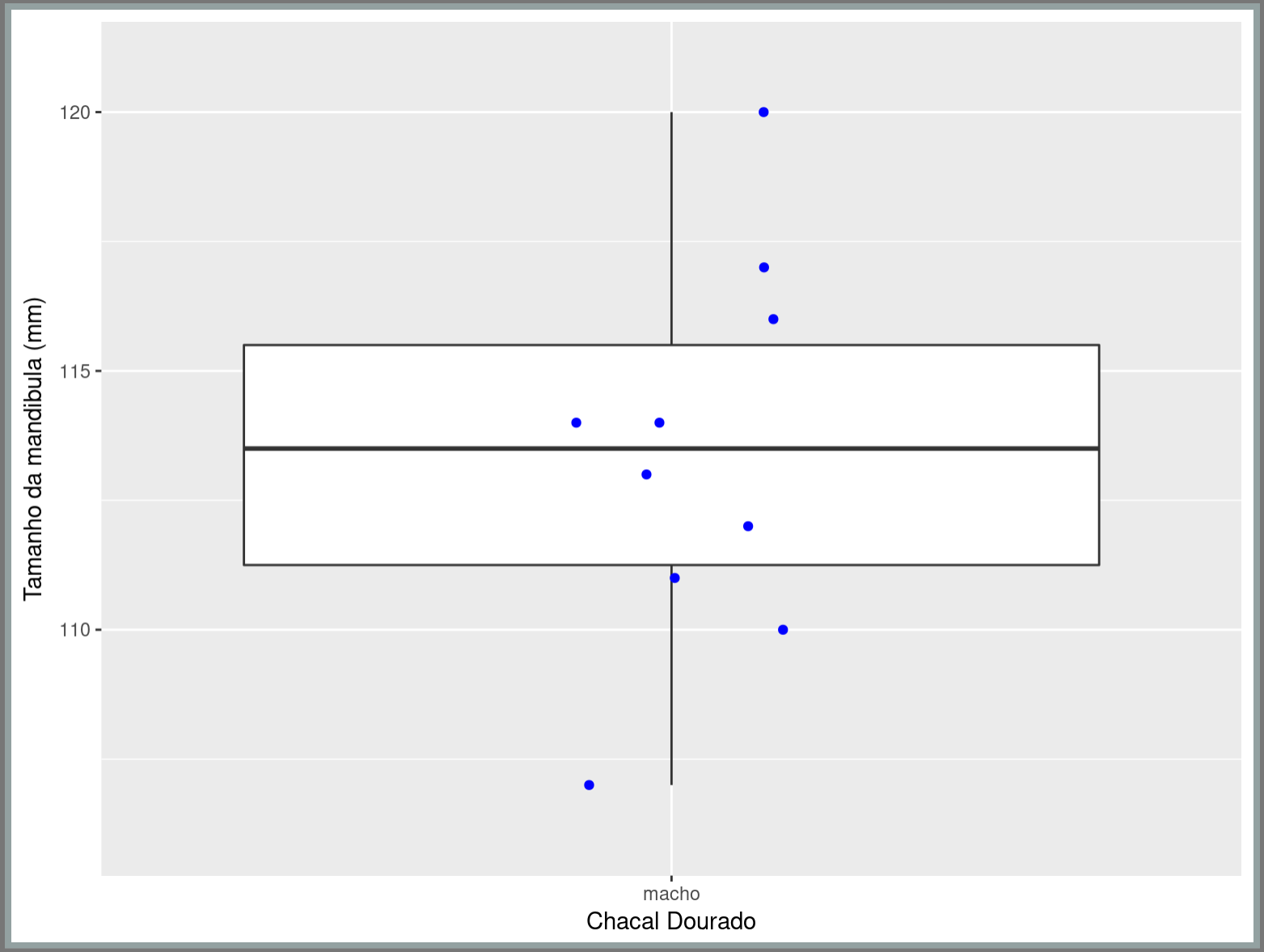
```
 [1] "c" "d" "e" "b" "c" "e" "e" "a" "c"
```

# Bootstrap

## Intervalo de confiança por percentil

```
macho=c(120,107,110,116, 114, 111, 113,1
(media.m=mean(macho))

[1] 113.4
```

# Chacal macho: resultados



# Qual a minha confiança sobre uma estimativa?

intervalo de confiança da média

```
macho
```

```
 [1] 120 107 110 116 114 111 113 117 114
```

```
sample(macho, replace = TRUE)
```

```
 [1] 117 114 116 113 120 114 120 110 112
```

```
sample(macho, replace = TRUE)
```

```
 [1] 116 114 120 112 114 112 114 113 112
```

# Estimativa bootstrap

```
mean(macho)
```

```
[1] 113.4
```

```
mean(sample(macho, replace = TRUE))
```

```
[1] 114.3
```

```
mean(sample(macho, replace = TRUE))
```

```
[1] 112.1
```

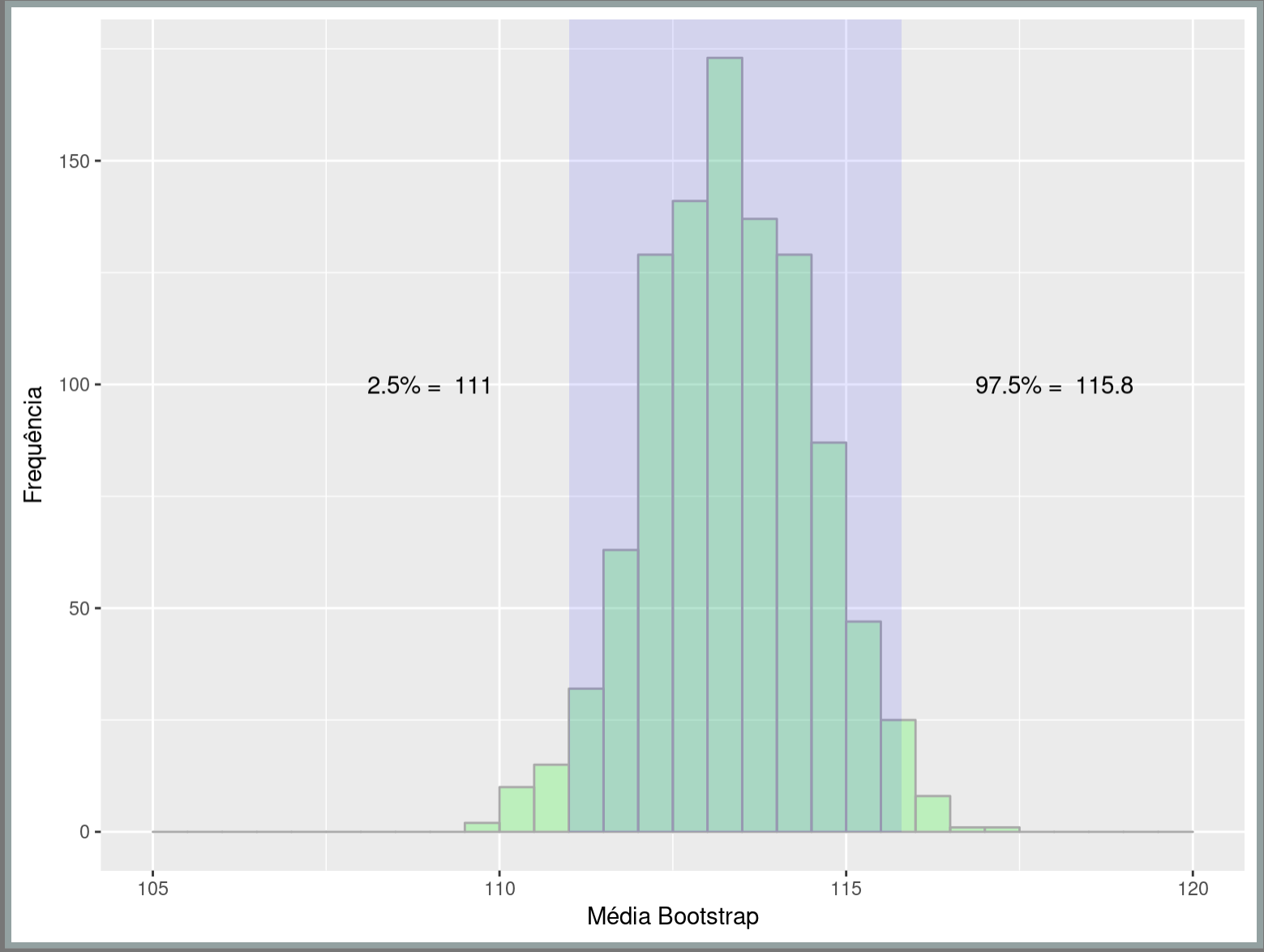
```
mean(sample(macho, replace = TRUE))
```

```
[1] 114.2
```

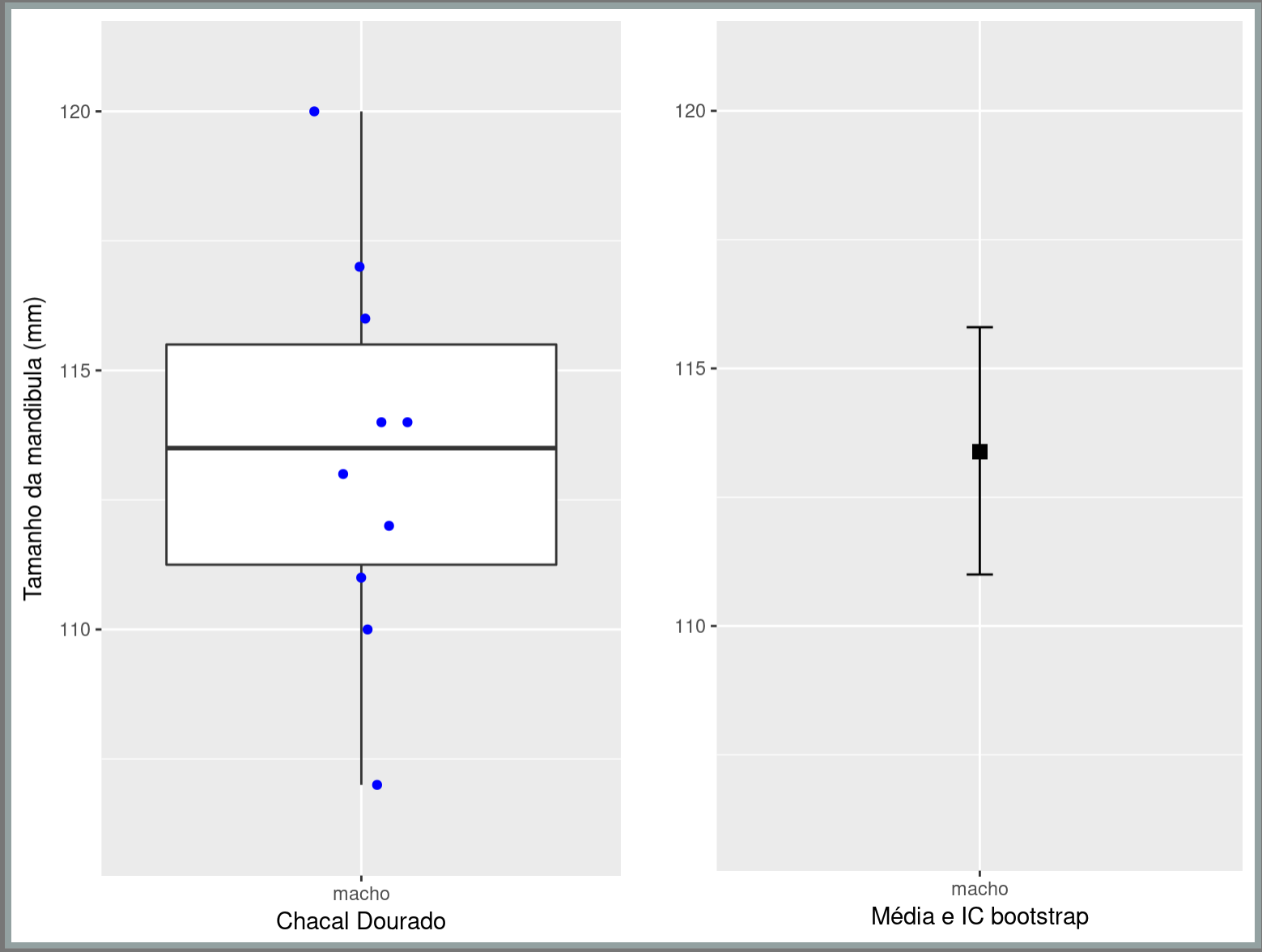
# Bootstrap

```
nBoot = 1000
bvalue <- rep(NA, nBoot )
bvalue[1] <- mean(macho)
  for(i in 2:nBoot)
  {
    bvalue[i] <- mean(sample(macho, repl
  }
```

# Bootstrap

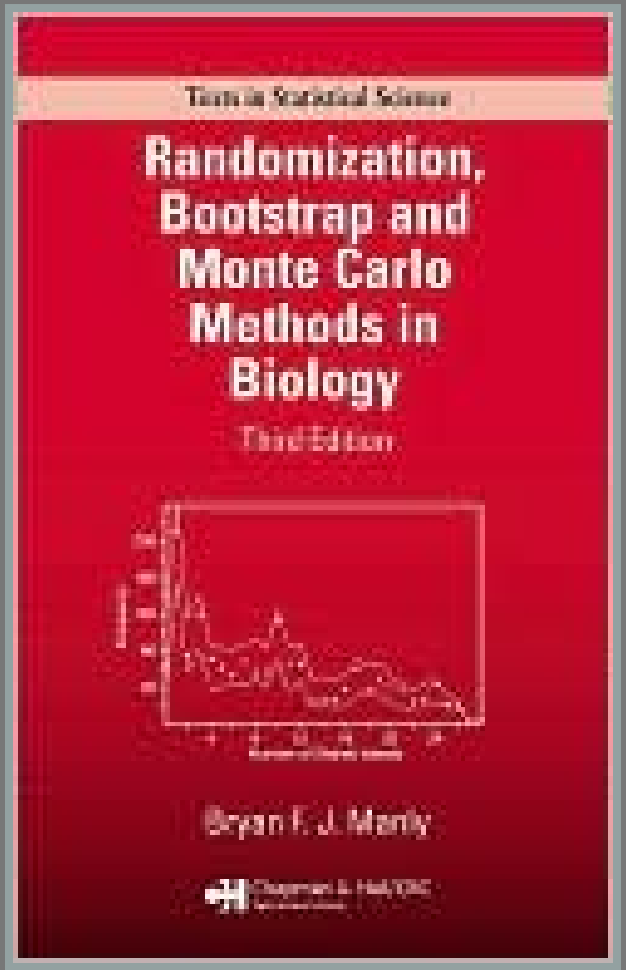

150
100
50
0
Frequência
2.5% = 111
97.5% = 115.8
105
110
115
120
Média Bootstrap

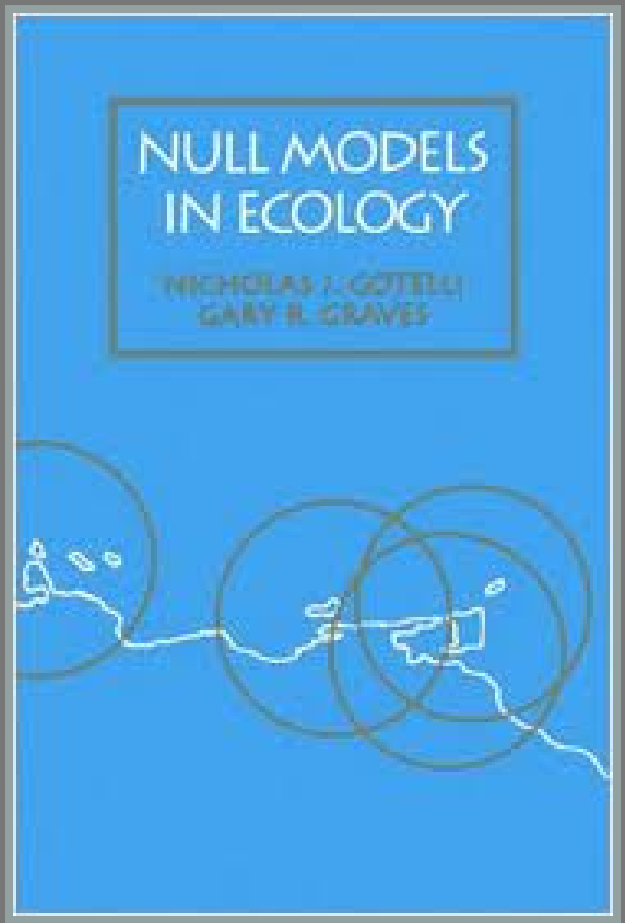
# Chacal macho: Resultado

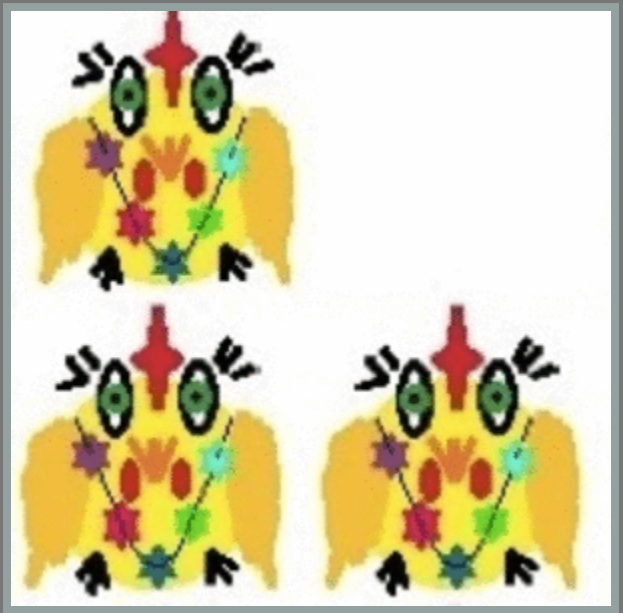


# Fim da Aula

# Bibliografia



# Atividades da Tarde