Curso R

Modelos Lineares

Alexandre Adalardo de Oliveira

Ecologia- IBUSP maio 2017

Use R: Modelos Lineares

Conceitos

UNIFICAÇÂO METODOLÓGICA

- análises frequentistas
- regressão
- regressão x ANOVA
- variavel dummy
- matriz do modelo
- diagnóstico do modelo

Testes Clássicos

Tipo de Variável		Estatística Clássica	
Resposta	Preditora	Teste	Hipótese
Categórica	Categórica	Qui-quadrado	independência
Contínua	Categórica (2 niveis)	Teste t	$\mu_1 = \mu_2$
Contínua	Categórica	Anova	$\mu_1 = \mu_2 = \mu_n$
Contínua	1 Contínua	Regressão	$\beta_1 = 0$
Contínua	>1 Contínua	Reg. múltipla	$\beta_1 = 0 ; \beta_n = 0$
Contínua	Cont + Categ	Ancova	$\beta_1 = \beta_2 ; \alpha_1 = \alpha 2$
Proporção	Contínua	Reg. Logística	$logit(\beta_1) = 1$

Ferramental Analítico

Regressão logística

Correlação

Test T

Qui-quadrado

ANCOVA

ANOVA

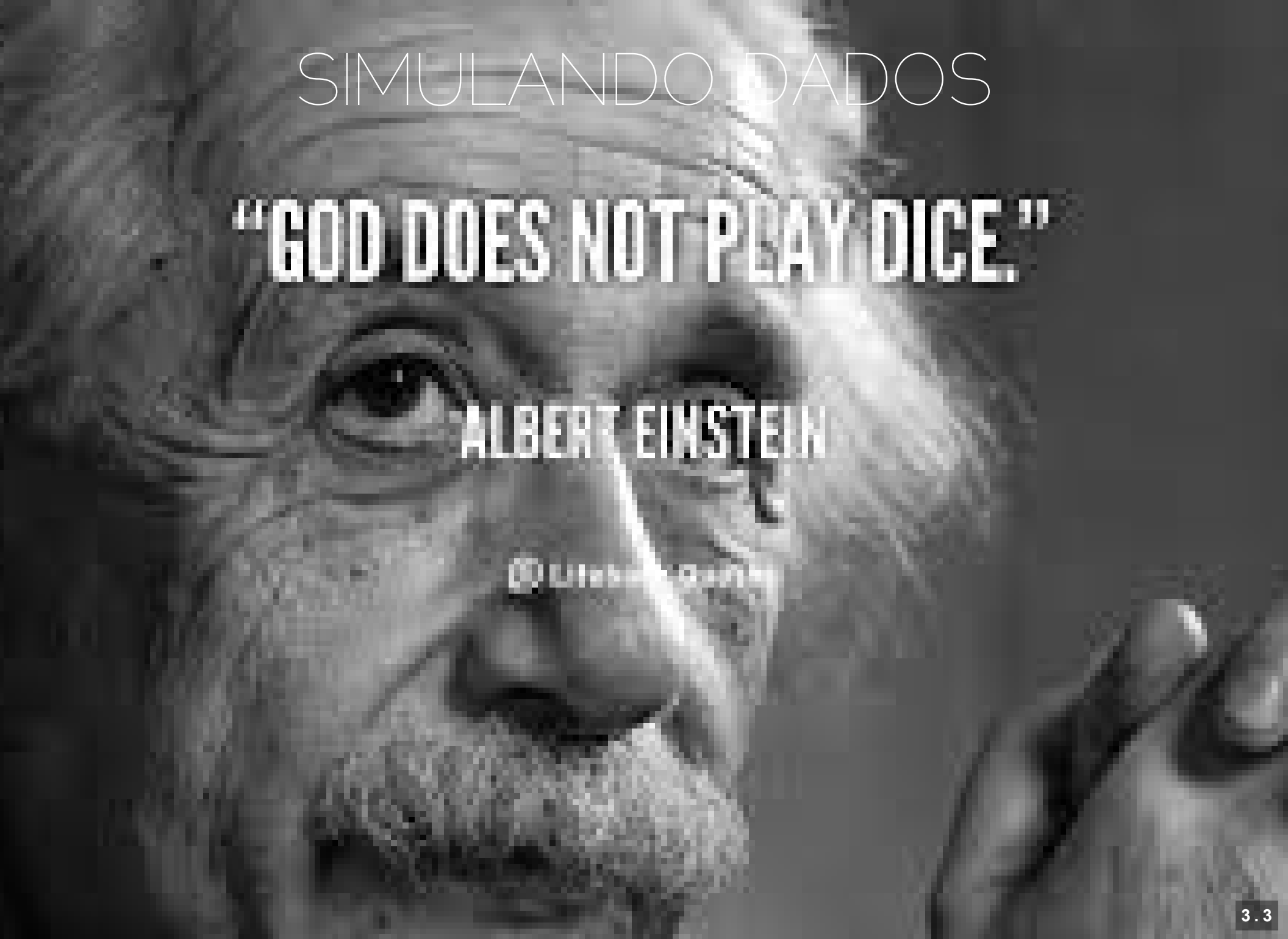
Regressão Linear

Regressão Linear

O modelo de regressão

$$y = \hat{\alpha} + \hat{\beta}x + \epsilon$$
$$\epsilon = N(0, \sigma)$$

SIMULANDO DADOS



Simulando dados

$$y = \hat{\alpha} + \hat{\beta}x + \epsilon$$
$$\epsilon = N(0, \sigma)$$

Simulando dados

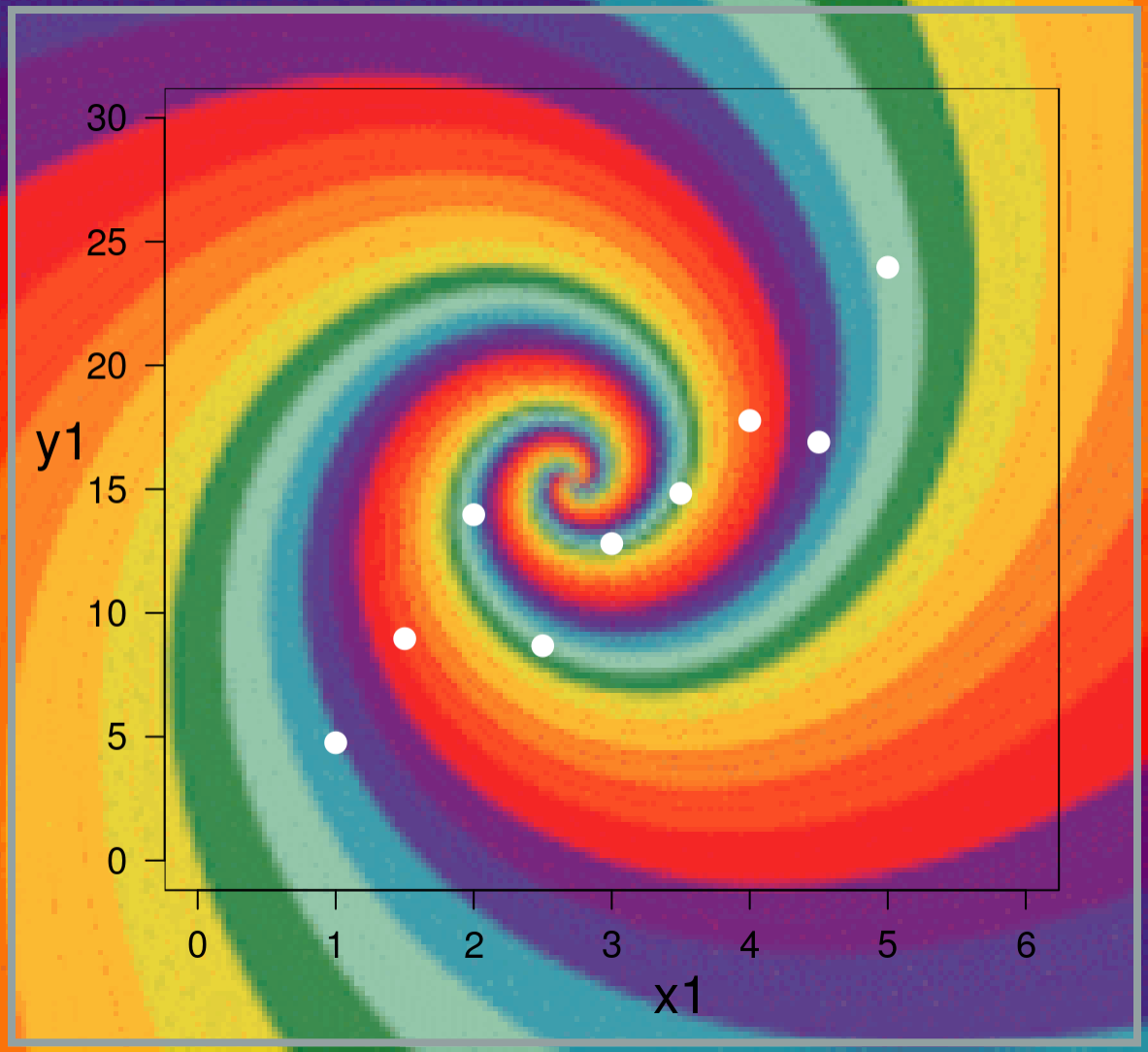
```
set.seed(2)
(x1 = seq(1,5, by=0.5))
```

```
## [1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5
```

```
y1 = 4 + 3 * x1 + rnorm(n= 9, mean= 0, s
y1
```

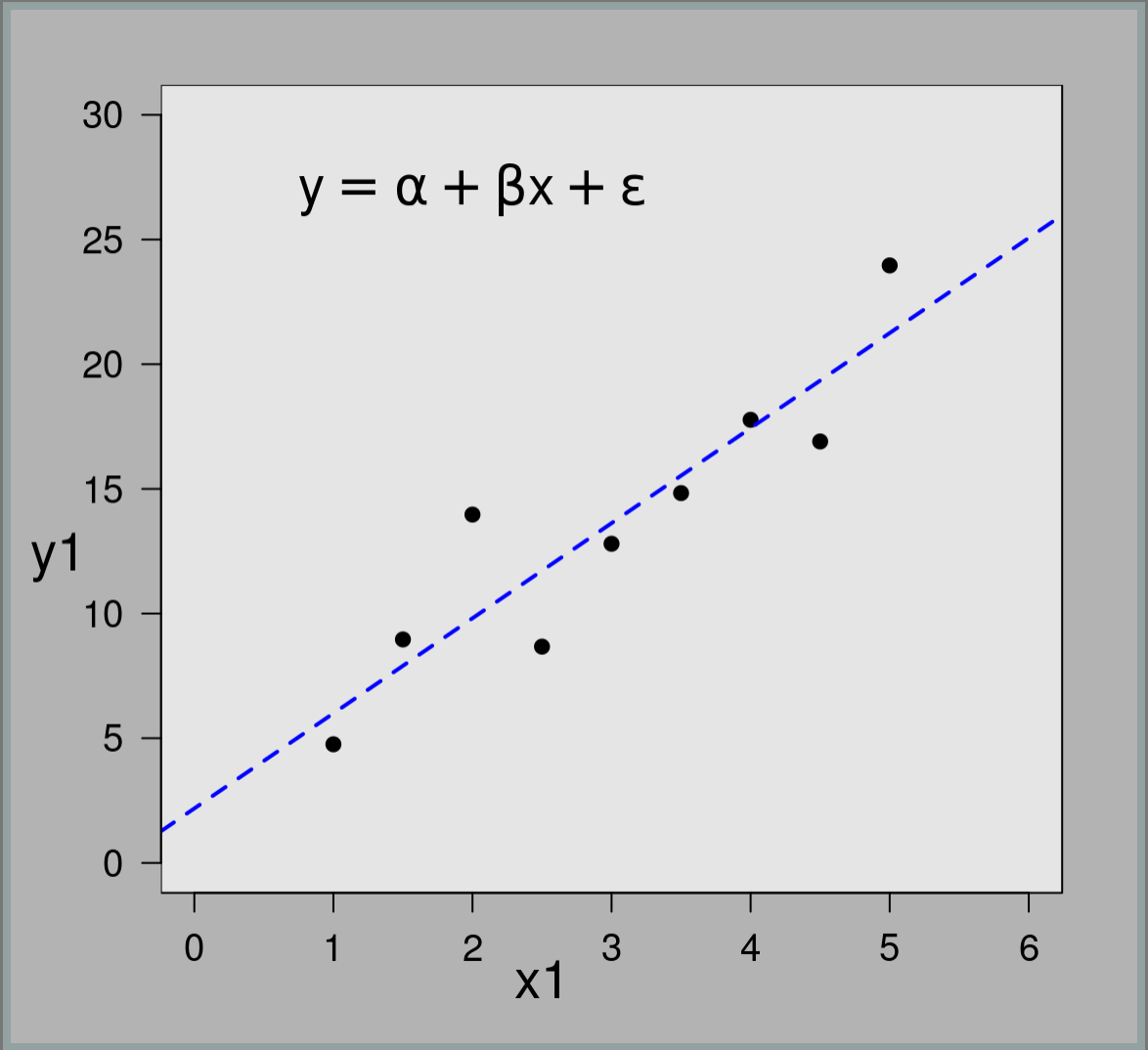
```
## [1]  4.757714  8.962123 13.969613  8.
## [8] 16.900755 23.961185
```

Dados REAIS



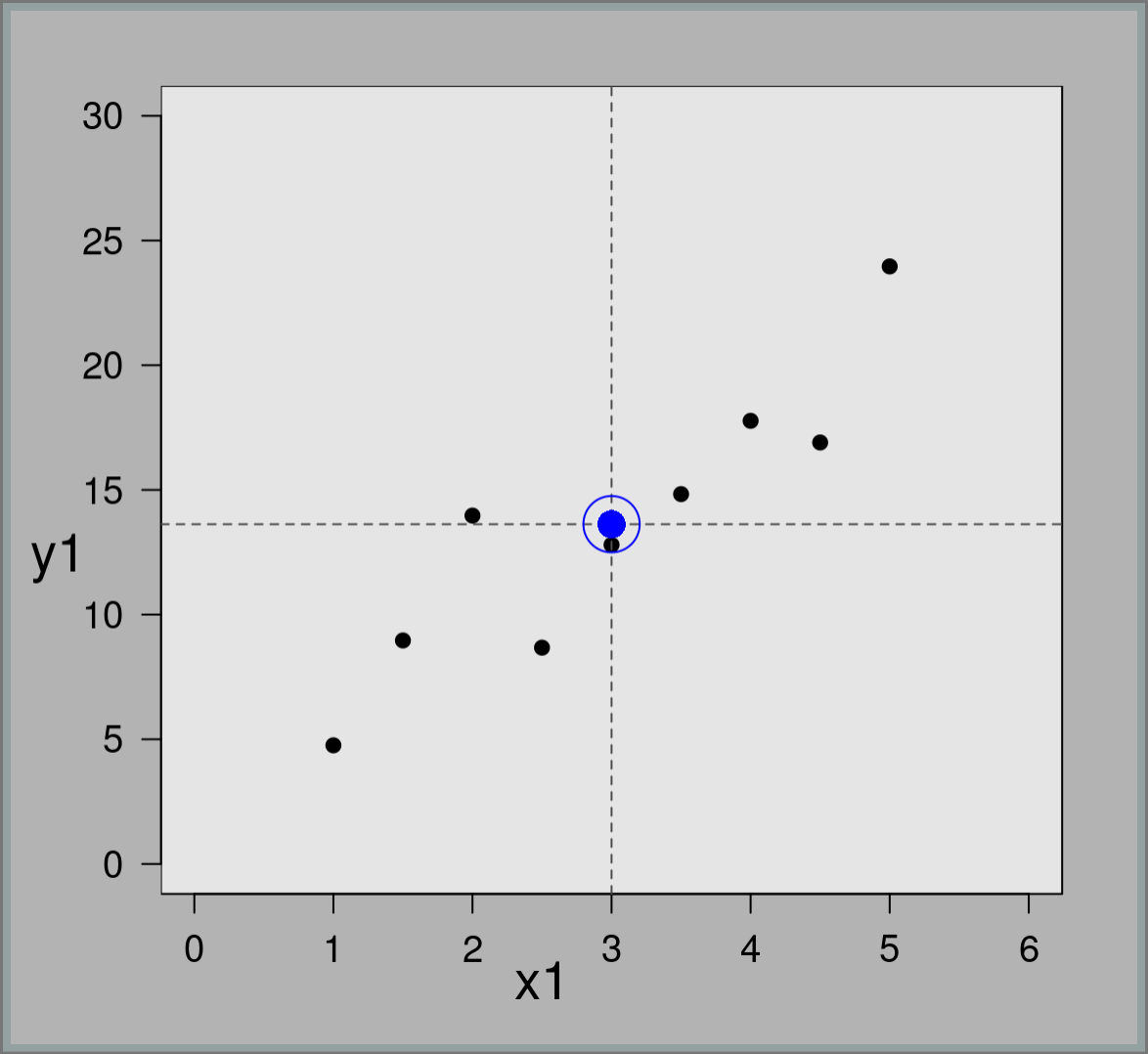
Modelo de Regressão

Estimar os parâmetros:



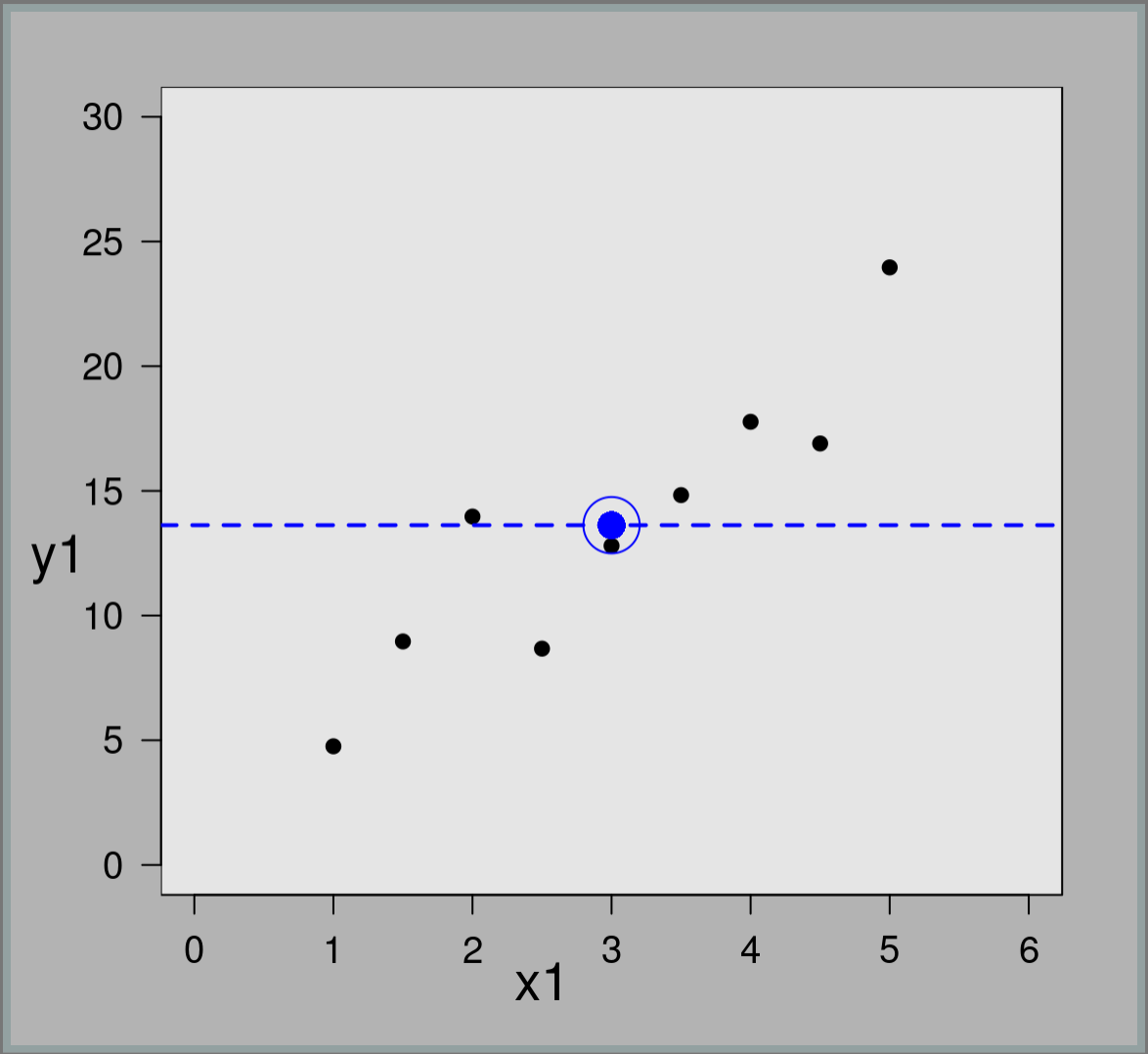
Regressão Linear

$$y = \hat{\alpha} + \hat{\beta}x + \epsilon$$



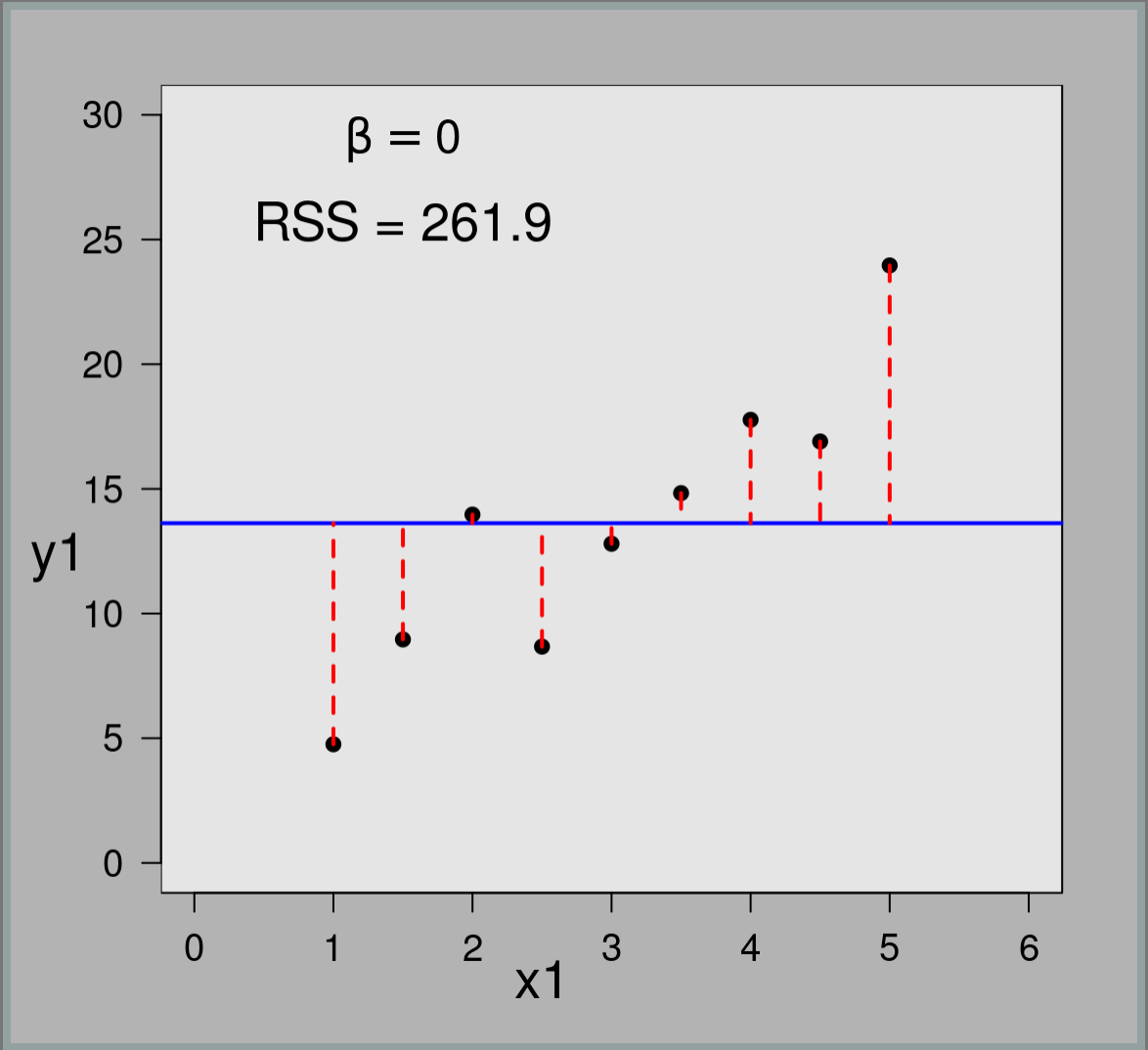
Modelo simples: nulo

$$y = \bar{y}; \beta = 0$$



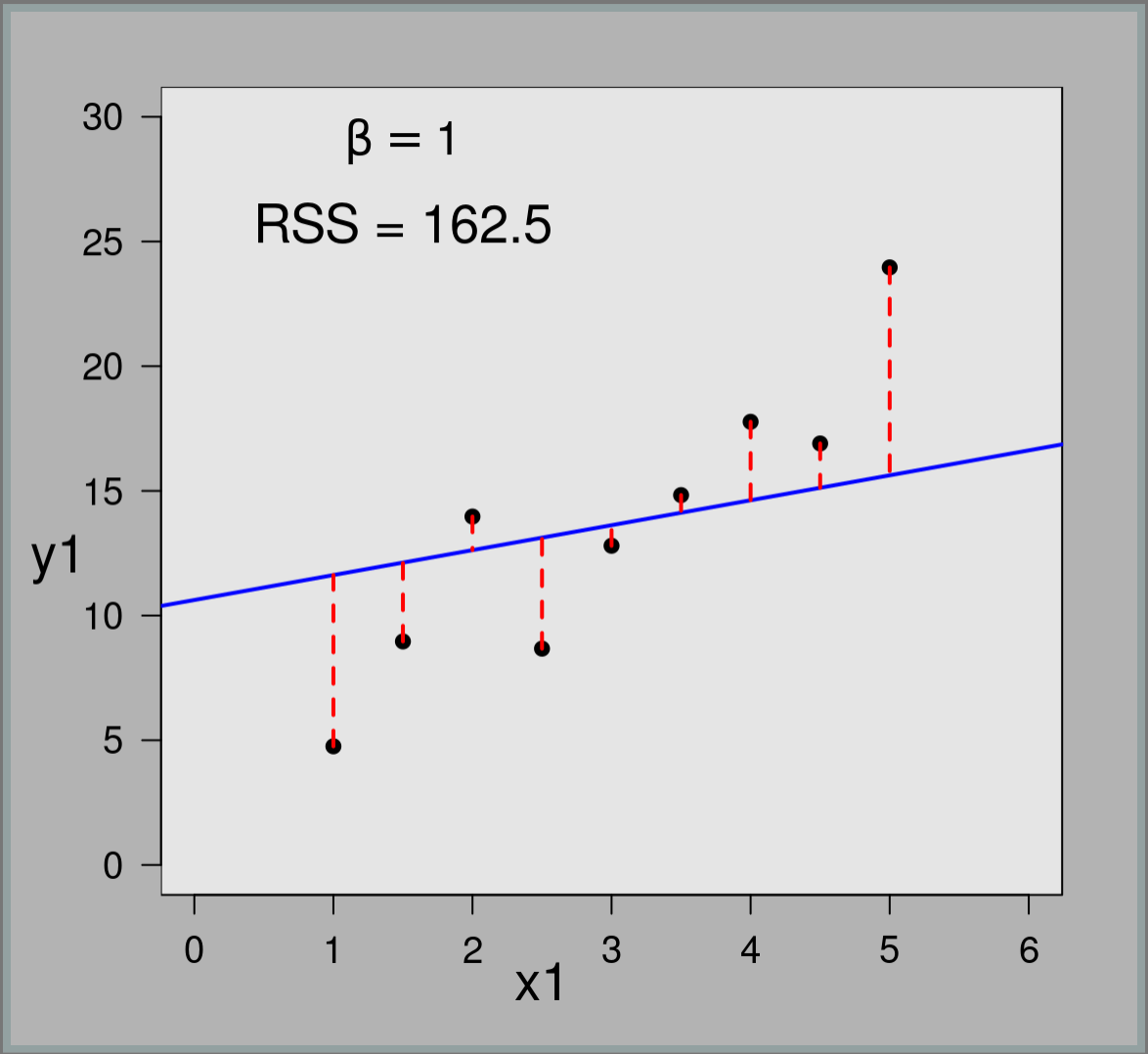
Resíduos e RSS

$$d = y_i - \hat{y}_i$$



Mínimo RSS

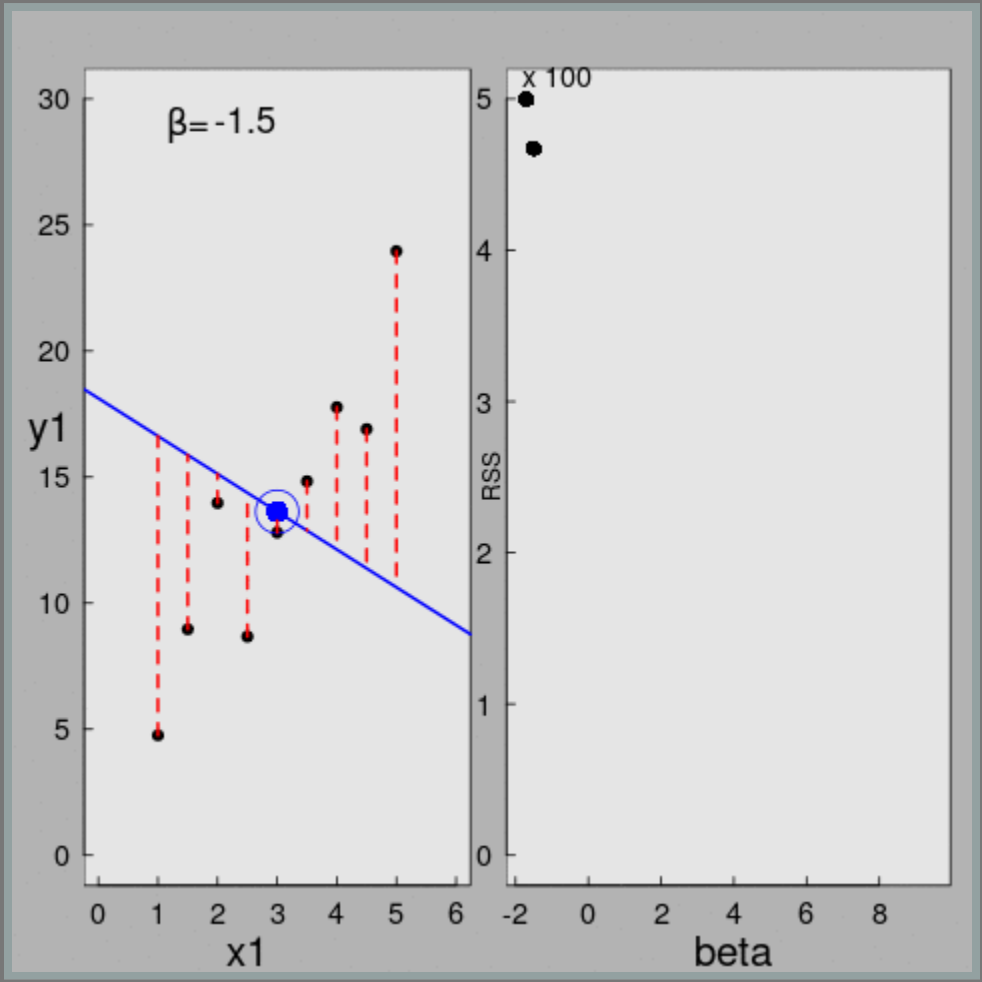
$$RSS = \sum (y_i - \hat{y}_i)^2$$



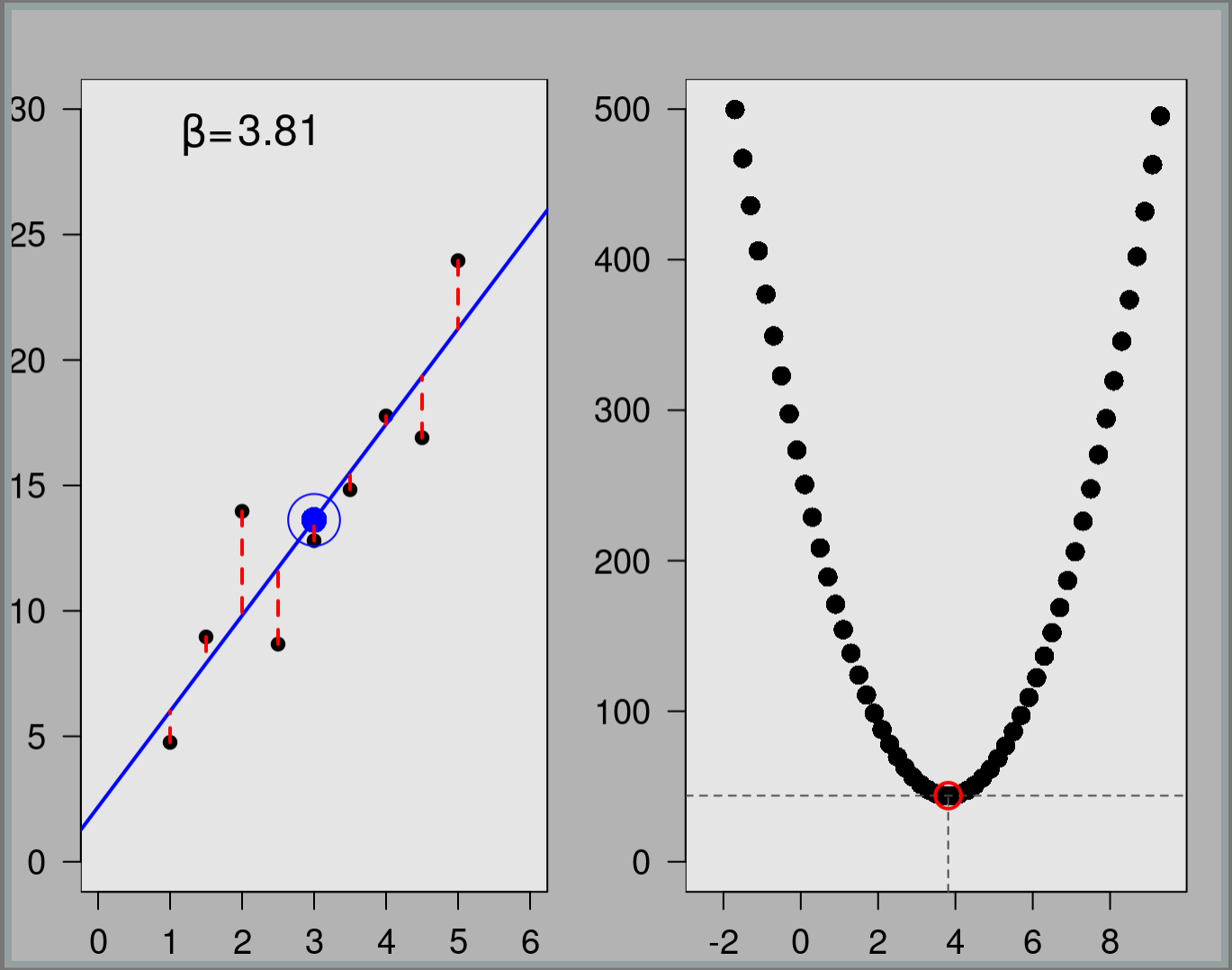
```
## Executing:
## convert -loop 0 -delay 100 Rplot1.png
##     Rplot4.png Rplot5.png Rplot6.png
##     Rplot9.png Rplot10.png Rplot11.pn
##     Rplot14.png Rplot15.png Rplot16.p
##     Rplot19.png Rplot20.png Rplot21.p
##     Rplot24.png Rplot25.png Rplot26.p
##     Rplot29.png Rplot30.png Rplot31.p
##     Rplot34.png Rplot35.png Rplot36.p
##     Rplot39.png Rplot40.png Rplot41.p
## Output at: msr.gif

## [1] TRUE
```

MMQ animado

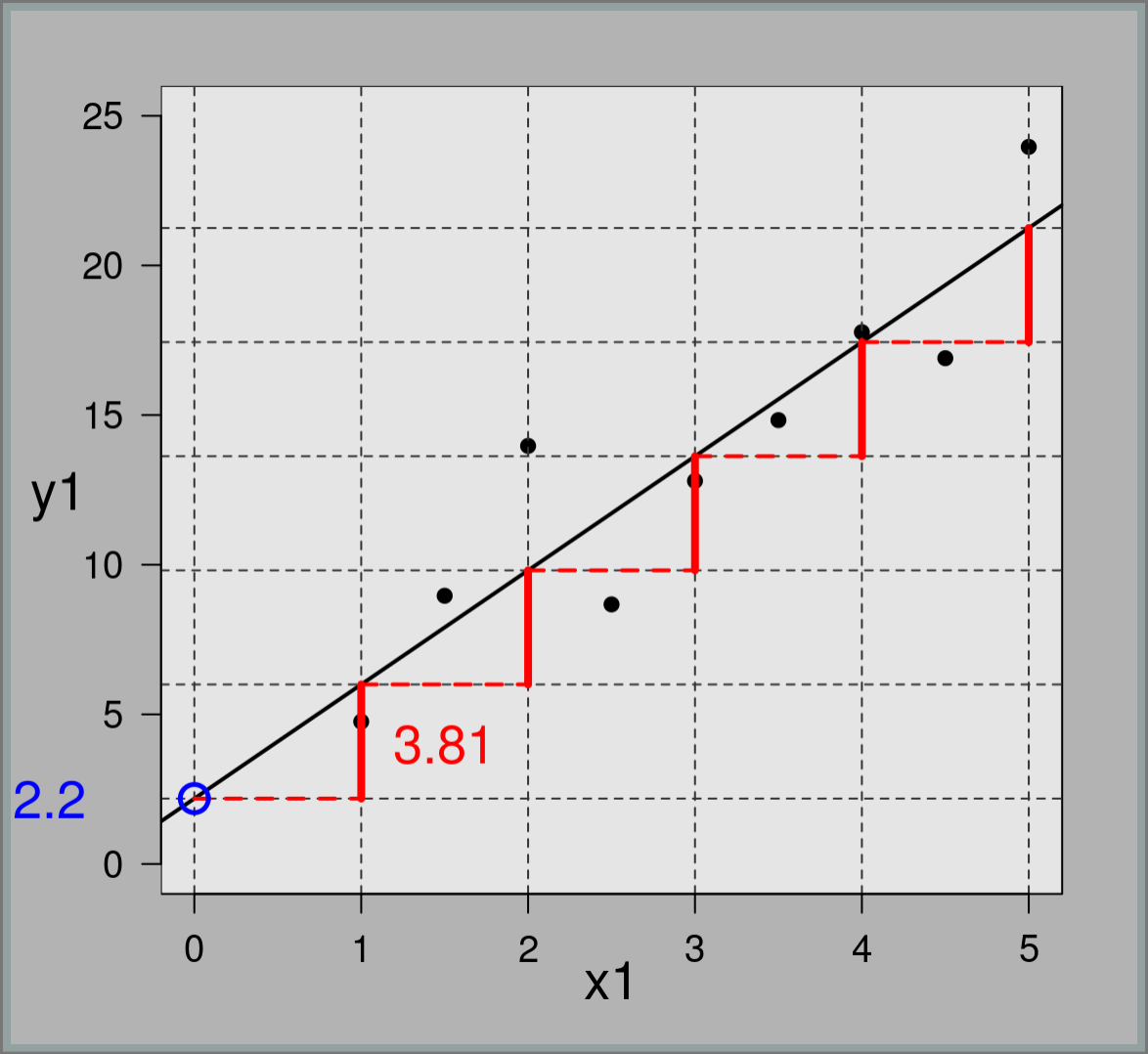


Método dos Mínimos Quadrados

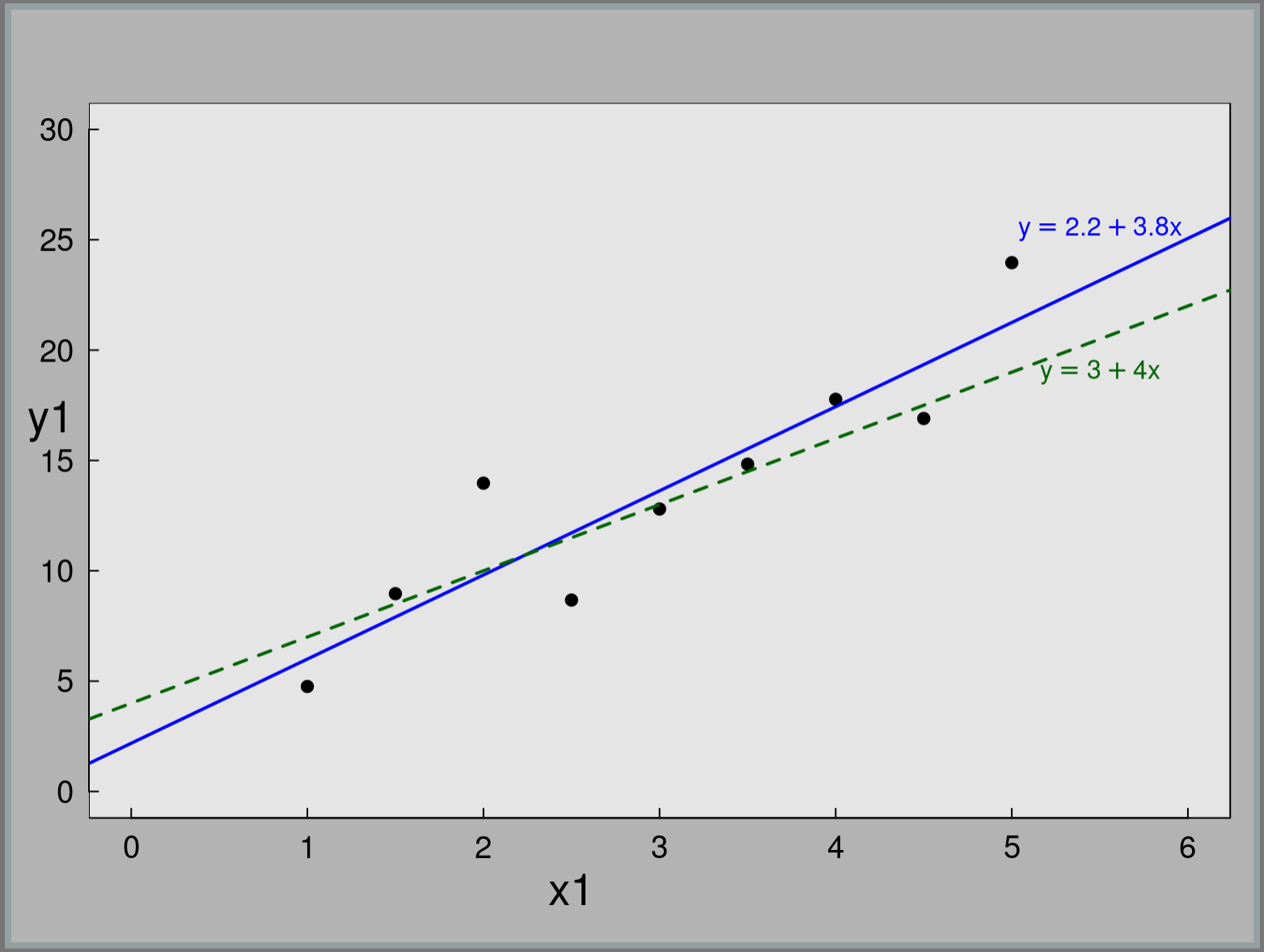


Regressão: dados simulados

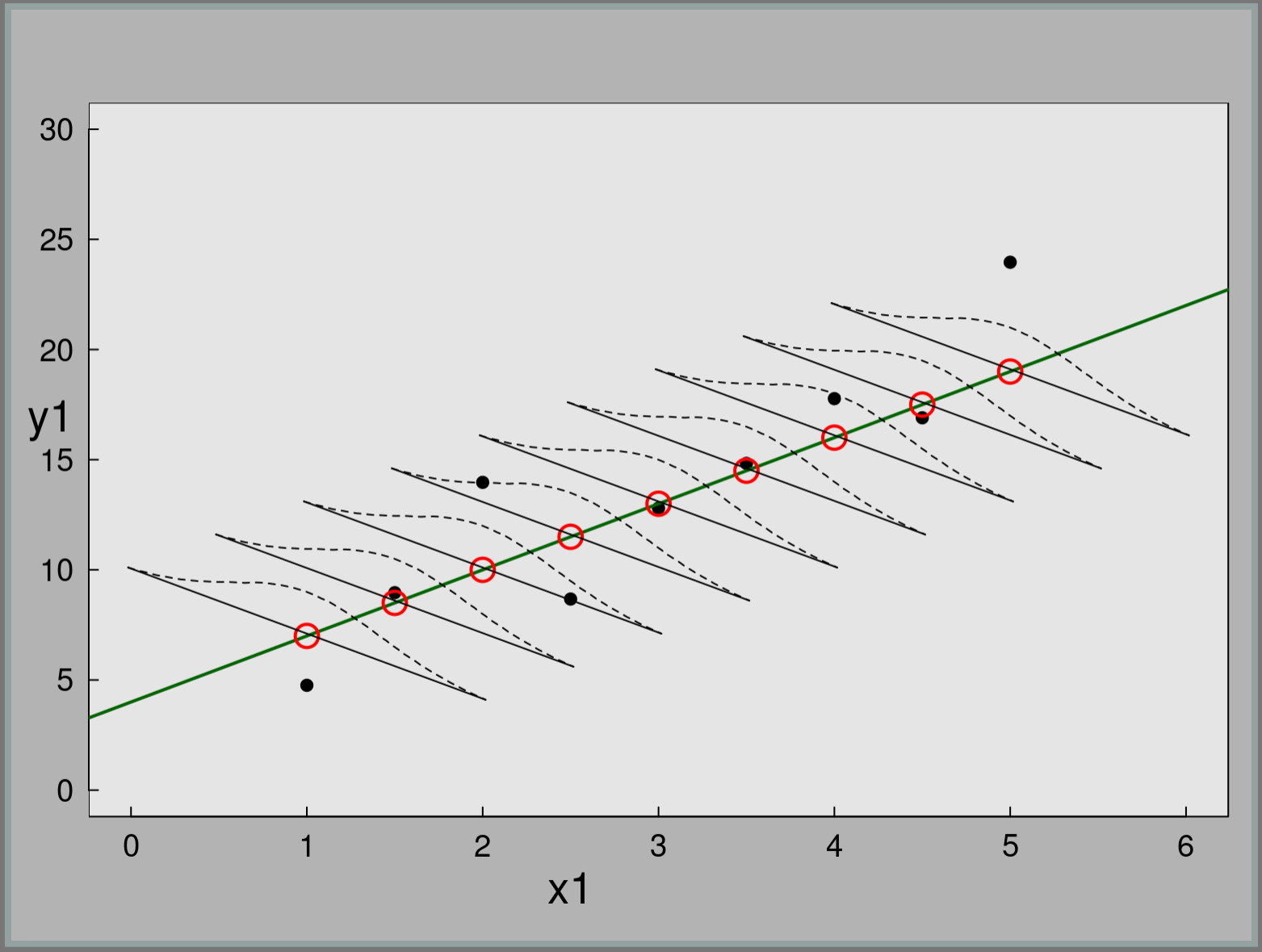
$$y = \hat{\alpha} + \hat{\beta}x + \epsilon$$



Predição x Parâmetros



Resíduos Gaussianos



Exemplo: dieta de lagarta

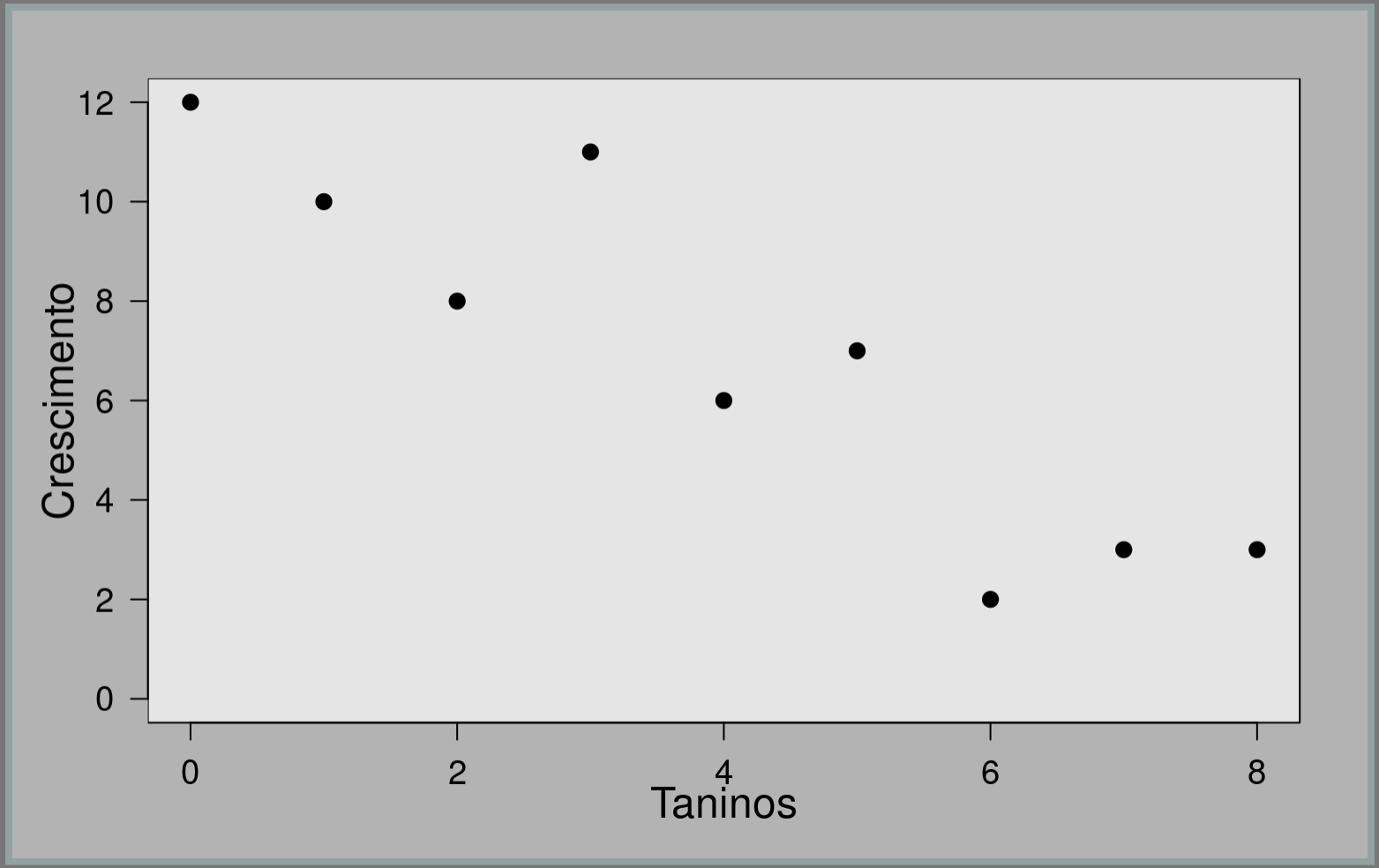
```
lag <- read.table("data/regression.txt",
str(lag)

## 'data.frame':    9 obs. of  2 variabl
##  $ growth: int  12 10 8 11 6 7 2 3 3
##  $ tannin: int  0 1 2 3 4 5 6 7 8

##   growth tannin   residuos  preditos
## 1     12      0  0.2444444 11.755556
## 2     10      1 -0.5388889 10.538889
## 3      8      2 -1.3222222  9.322222
## 4     11      3  2.8944444  8.105556
## 5      6      4 -0.8888889  6.888889
## 6      7      5  1.3277778  5.672222
```

Exemplo: dieta de lagarta

```
plot(growth ~ tannin, data = lag)
```



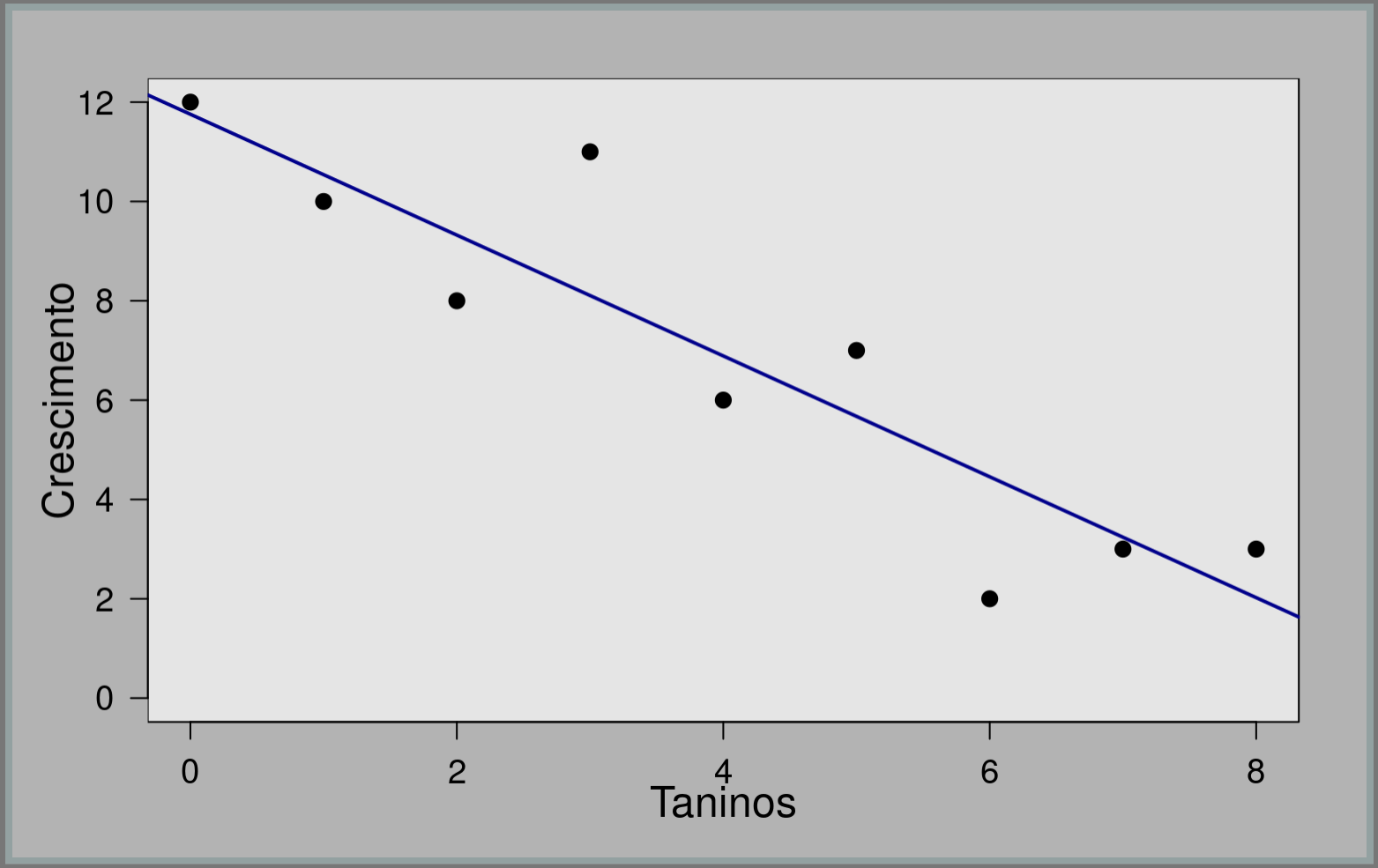
Modelo Linear: lagartos

```
lmlag <- lm(growth ~ tannin, data = lag)
summary(lmlag)
```

```
##
## Call:
## lm(formula = growth ~ tannin, data =
##
## Residuals:
##     Min      1Q  Median      3Q     M
## -2.4556 -0.8889 -0.2389  0.9778  2.89
##
## Coefficients:
##             Estimate Std. Error t val
```

Exemplo: dieta de lagarta

```
plot(growth ~ tannin, data = lag)
abline(lmlag)
```

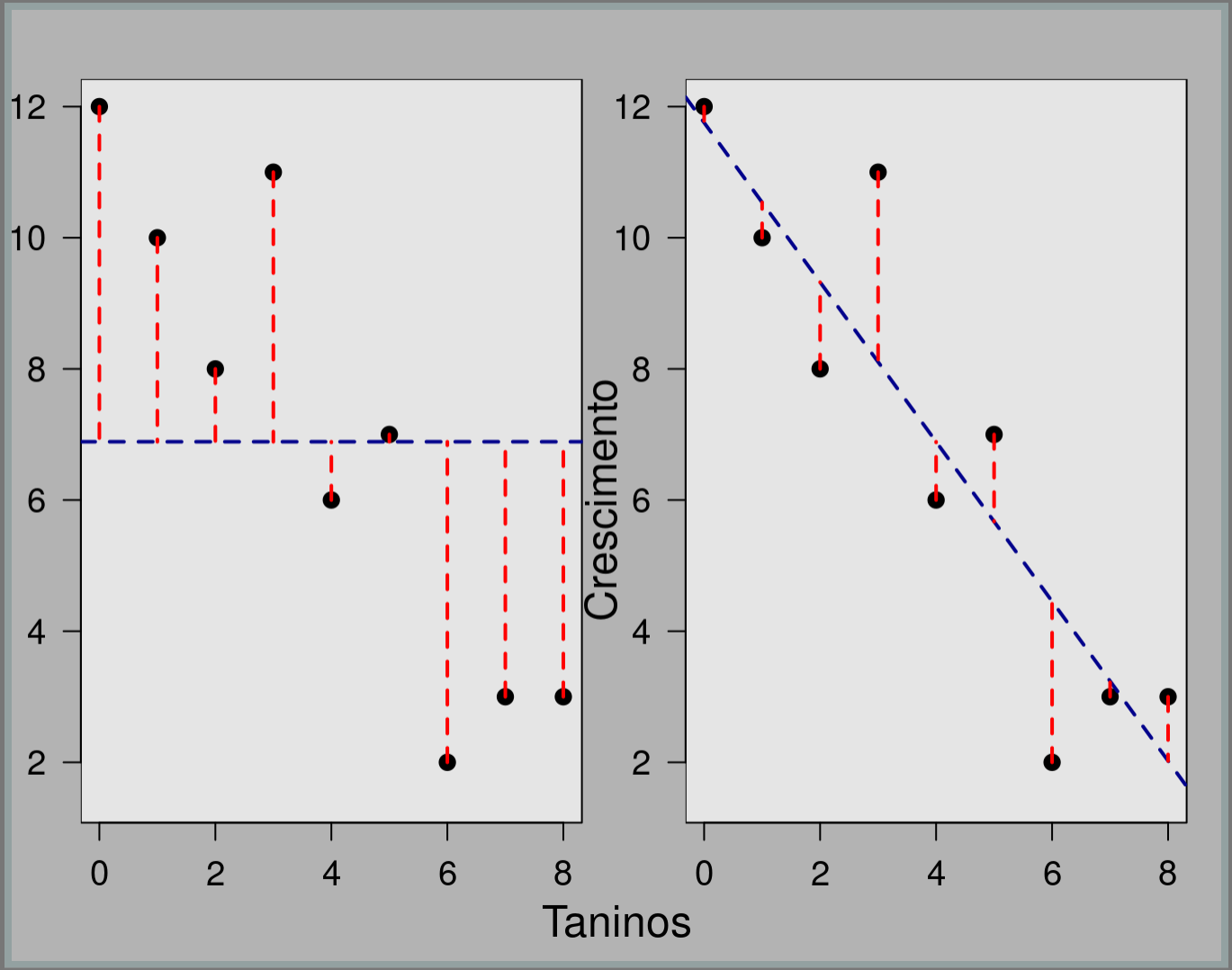


Anova do Modelo

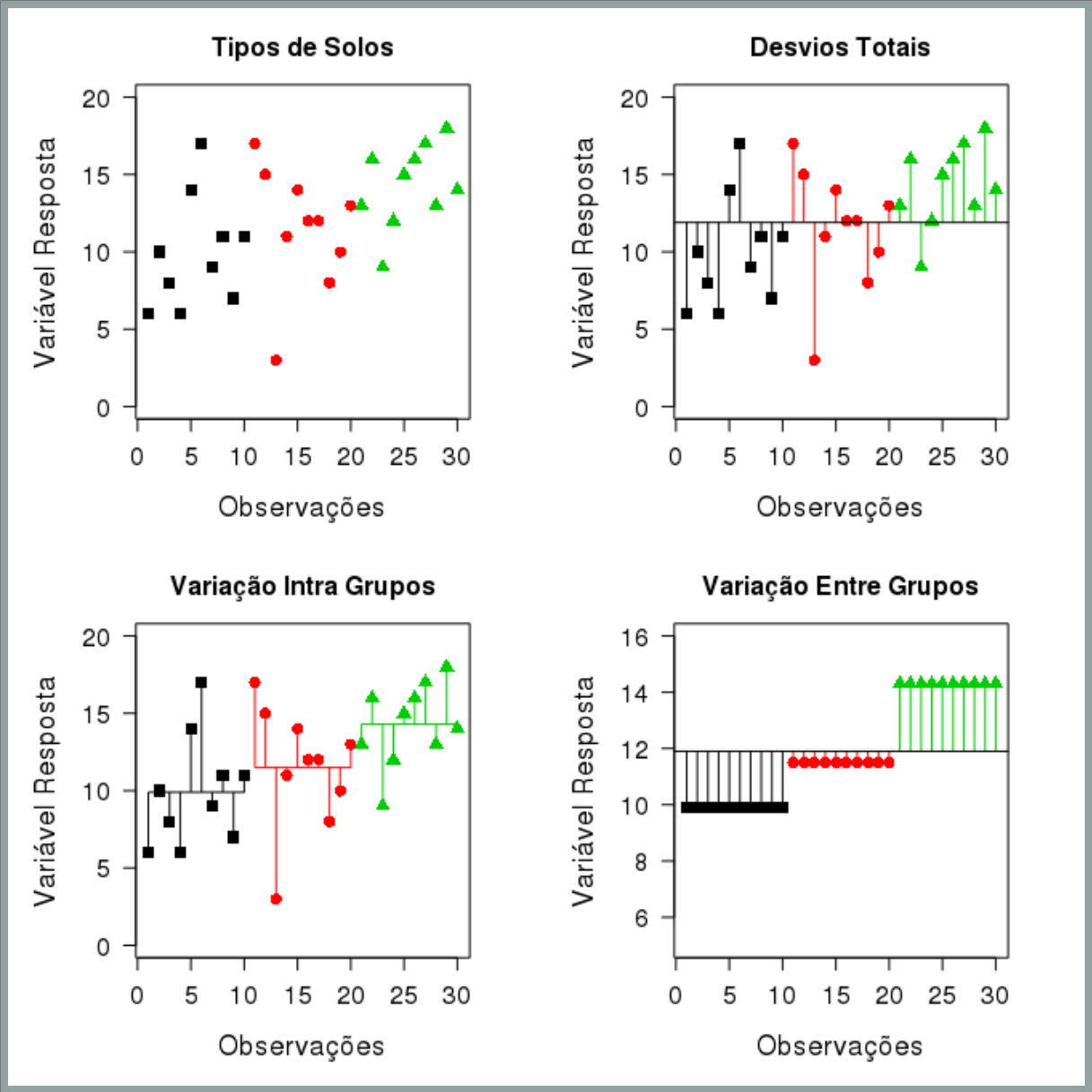
```
anova(lmlag)
```

```
## Analysis of Variance Table
##
## Response: growth
##           Df Sum Sq Mean Sq F value
## tannin     1 88.817  88.817  30.974 0
## Residuals  7 20.072   2.867
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.
```

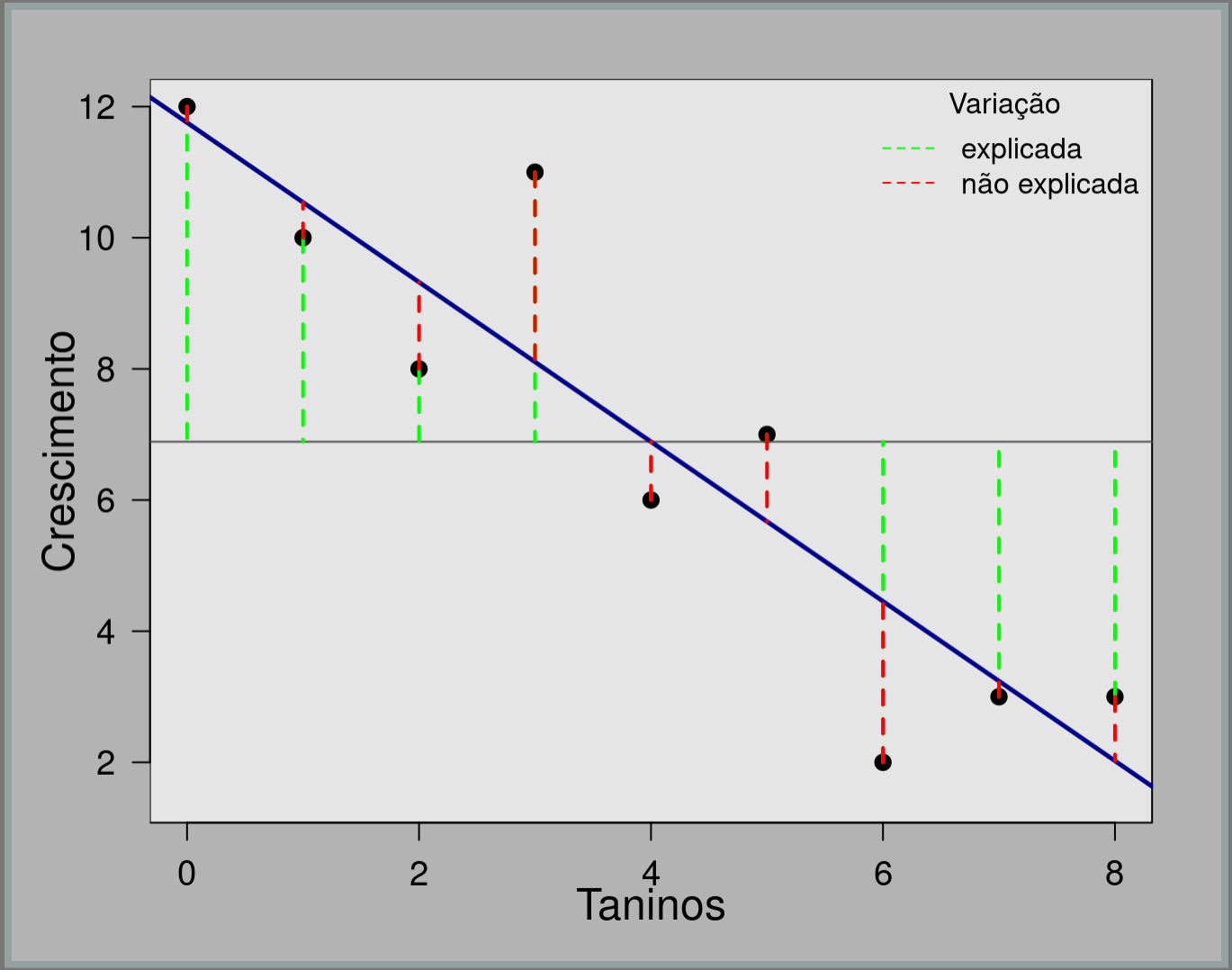
Lógica da ANOVA



Anova: partição da variação



Regressão: ANOVA



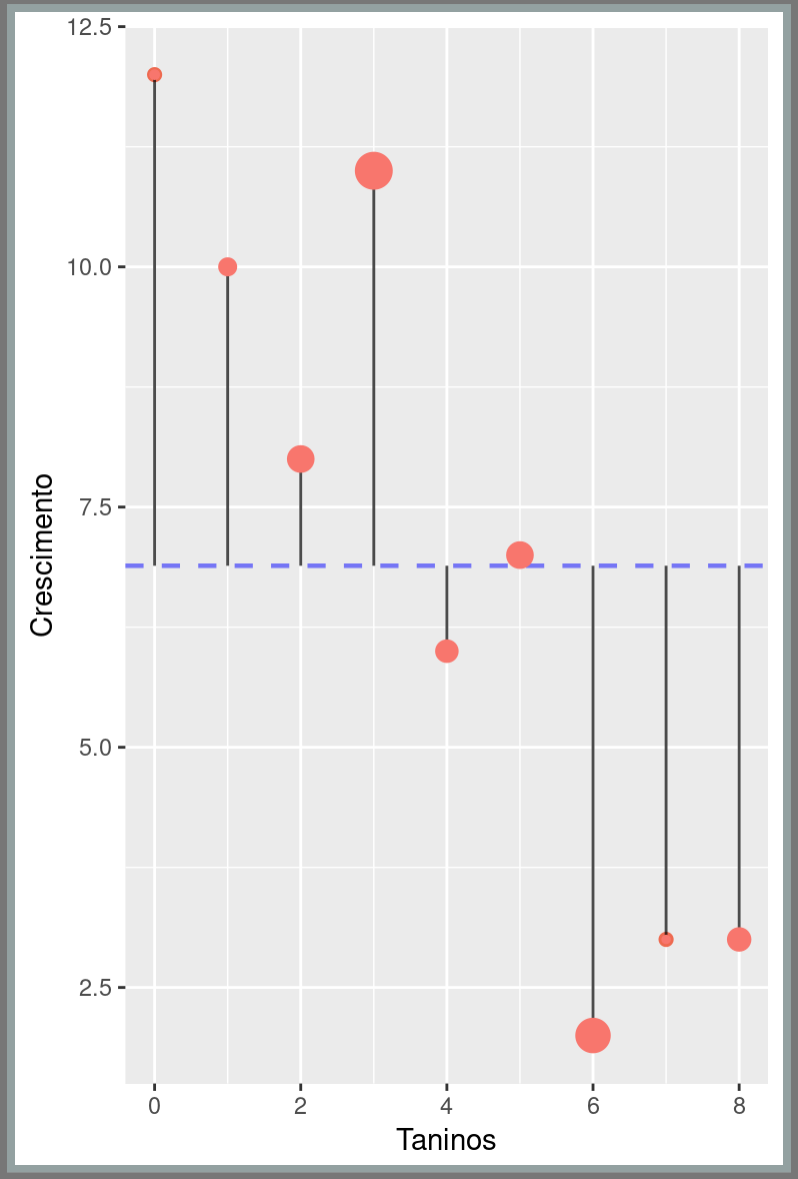
Lógica da Anova

$$SS_{total} = SS_{entre} + SS_{intra}$$

Lógica da Regressão

$$SS_{total} = SS_{regr} + SS_{erro}$$

Modelo mínimo



$$SS_{total} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

Desvios quadráticos total

$$SS_{total} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

```
(dt <- lag$growth - mean(lag$growth))
```

```
## [1]  5.1111111  3.1111111  1.1111111
## [7] -4.8888889 -3.8888889 -3.8888889
```

```
dt^2
```

```
## [1] 26.12345679  9.67901235  1.234567
## [7] 23.90123457 15.12345679 15.123456
```

```
(ss_total <- sum(dt^2))
```

```
## [1] 108.8889
```

Desvios quadráticos do ERRO

$$SS_{error} = \sum_{i=1}^{n} (y_i - \hat{y})^2$$

Desvios quadráticos do ERRO

$$SS_{error} = \sum_{i=1}^{n} (y_i - \hat{y})^2$$

```
(coeflag <- coef(lmlag))
```

```
## (Intercept)      tannin
##   11.755556   -1.216667
```

```
(predlag <- coeflag[1] + coeflag[2] * la
```

```
## [1] 11.755556 10.538889  9.322222  8.
## [8]  3.238889  2.022222
```

```
lag$growth
```

```
## [1] 12 10  8 11  6  7  2  3  3
```

Desvios quadráticos do ERRO

$$SS_{error} = \sum_{i=1}^{n} (y_i - \hat{y})^2$$

```
(ss_erro <- sum((lag$growth - predlag)^2
```

```
## [1] 20.07222
```

Lógica da Regressão

$$SS_{total} = SS_{regr} + SS_{erro}$$

```
(ss_reg <- ss_total - ss_erro)
```

```
## [1] 88.81667
```

Tabela de Anova

Fonte	SumSquare	GL	MeanSquare
Regressão	88.82	1	88.82
Erro	20.07	7	2.87
Total	108.89	8	NA

Teste de hipótese: F e r^2

```
(r2 <- ss_reg/ss_total)
```

```
## [1] 0.8156633
```

```
(flag <- ss_reg/(ss_erro/7))
```

```
## [1] 30.97398
```

```
1- pf(flag, 1, 7)
```

```
## [1] 0.0008460738
```

Regressão no R: lagarta

```
laglm <- lm(growth ~ tannin, data=lag)
anova(laglm)

## Analysis of Variance Table
##
## Response: growth
##           Df Sum Sq Mean Sq F value
## tannin     1 88.817  88.817  30.974 0
## Residuals  7 20.072   2.867
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.
```

Comparando Modelos no R: lagarta

```
nullag <- lm(growth ~ 1, data = lag)
anova(nullag, laglm)
```

```
## Analysis of Variance Table
##
## Model 1: growth ~ 1
## Model 2: growth ~ tannin
##   Res.Df     RSS Df Sum of Sq      F
## 1      8 108.889
## 2      7  20.072  1    88.817 30.974
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.
```

Comparando Modelos no R: lagarta

Anova do modelo: anova(laglm)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
tannin	1	88.81667	88.81667	30.97398	0.0008461
Residuals	7	20.07222	2.86746	NA	NA

Anova da comparação de modelos: anova(nullag, laglm)

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
8	108.88889	NA	NA	NA	NA
7	20.07222	1	88.81667	30.97398	0.0008461

NÃO DESESPERE, ESPERE!

KEEP CALM!!


R

Variável categórica

Regressão de Variável Categórica

```
crop <- read.table("/home/aao/Ale2016/Al
str(crop)

## 'data.frame':    30 obs. of  2 variab
##  $ solo : chr  "are" "are" "are" "are
##  $ colhe: int  6 10 8 6 14 17 9 11 7
```

Variáveis Dummy ou Indicadoras

```
croplin <- crop[,c( "colhe", "solo")]
croplin$solo
```

```
##  [1] are are are are are are are are
## [18] arg arg arg hum hum hum hum hum
## Levels: are arg hum
```

```
croplin$arg <-0
croplin$arg[crop$solo=="arg"] <- 1
croplin$hum <- 0
croplin$hum[crop$solo=="hum"] <- 1
```

Variável Dummy ou Indicadora

```
croplin[c(1,2,3,11,12,13,21,22,23),]
```

```
##     colhe solo arg hum
## 1       6  are   0   0
## 2      10  are   0   0
## 3       8  are   0   0
## 11     17  arg   1   0
## 12     15  arg   1   0
## 13      3  arg   1   0
## 21     13  hum   0   1
## 22     16  hum   0   1
## 23      9  hum   0   1
```

Número de níveis do fator menos 1 (intercepto)

Modelo linear: dummy

Modelo

$$y = \alpha_{d_1} + \beta_2 x_{d_2} + \beta_3 x_{d_3}$$

Intercepto:

$$\alpha_{d_1} = \bar{x}_1$$

Coeficientes:

$$\beta_2 = \bar{x}_2 - \bar{x}_1$$

$$\beta_3 = \bar{x}_3 - \bar{x}_1$$

Regressão dummy

```
lmdum <- lm(colhe ~ arg + hum, croplin)
summary(lmdum)

##
## Call:
## lm(formula = colhe ~ arg + hum, data
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##    -8.5    -1.8     0.3     1.7     7.1
##
## Coefficients:
##             Estimate Std. Error t val
##   are   arg   hum
##   9.9 11.5 14.3
```

Modelo Linear Normal

```
lmCrop <- lm(colhe~solo, data = crop)
summary(lmCrop)

##
## Call:
## lm(formula = colhe ~ solo, data = cro
##
## Residuals:
##    Min     1Q Median     3Q    Max
##   -8.5   -1.8    0.3    1.7    7.1
##
## Coefficients:
##             Estimate Std. Error t val
```

Coeficientes do modelo

```
coef(lmdum)
```

```
## (Intercept)         arg         hum
##         9.9         1.6         4.4
```

```
tapply(crop$colhe, crop$solo, mean)
```

```
##  are  arg  hum
##  9.9 11.5 14.3
```

$$y = \hat{\alpha}_{d_1} + \hat{\beta}_2 x_{d_2} + \hat{\beta}_3 x_{d_3}$$

Regressão de Fator

Modelo

$$y = \alpha_{d_1} + \beta_2 x_{d_2} + \beta_3 x_{d_3}$$

Intercepto:

$$\alpha_{d_1} = \bar{x}_1$$

Coeficientes:

$$\beta_2 = \bar{x}_2 - \bar{x}_1$$

$$\beta_3 = \bar{x}_3 - \bar{x}_1$$

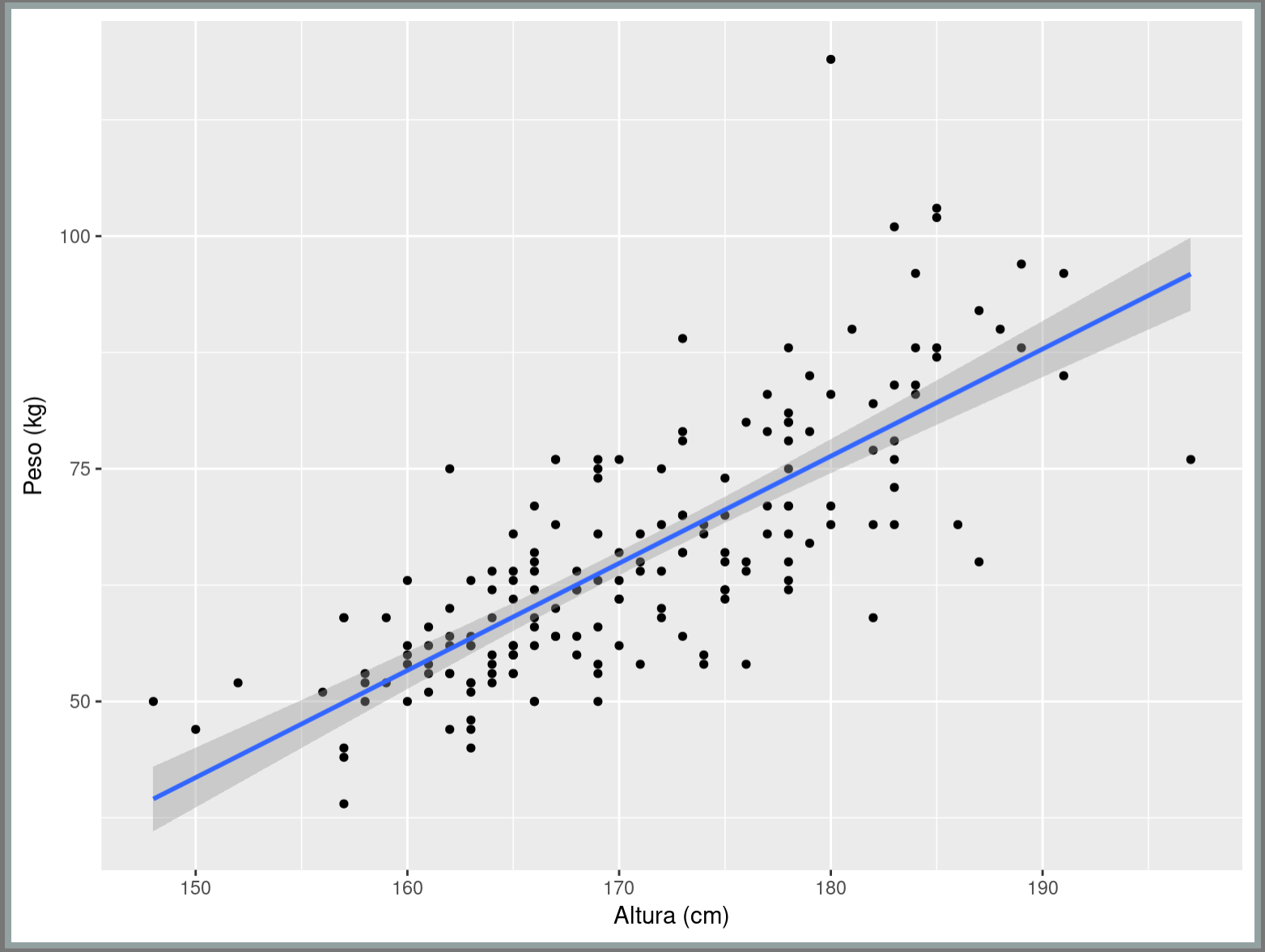
Retomando a regressão

Peso ~ altura

```
library(car)
data(Davis)
str(Davis)
```

```
## 'data.frame':    200 obs. of  5 varia
##  $ sex   : Factor w/ 2 levels "F","M"
##  $ weight: int  77 58 53 68 59 76 76
##  $ height: int  182 161 161 177 157 1
##  $ repwt : int  77 51 54 70 59 76 77
##  $ repht : int  180 159 158 175 155 1
```

Gráfico da Regressão:

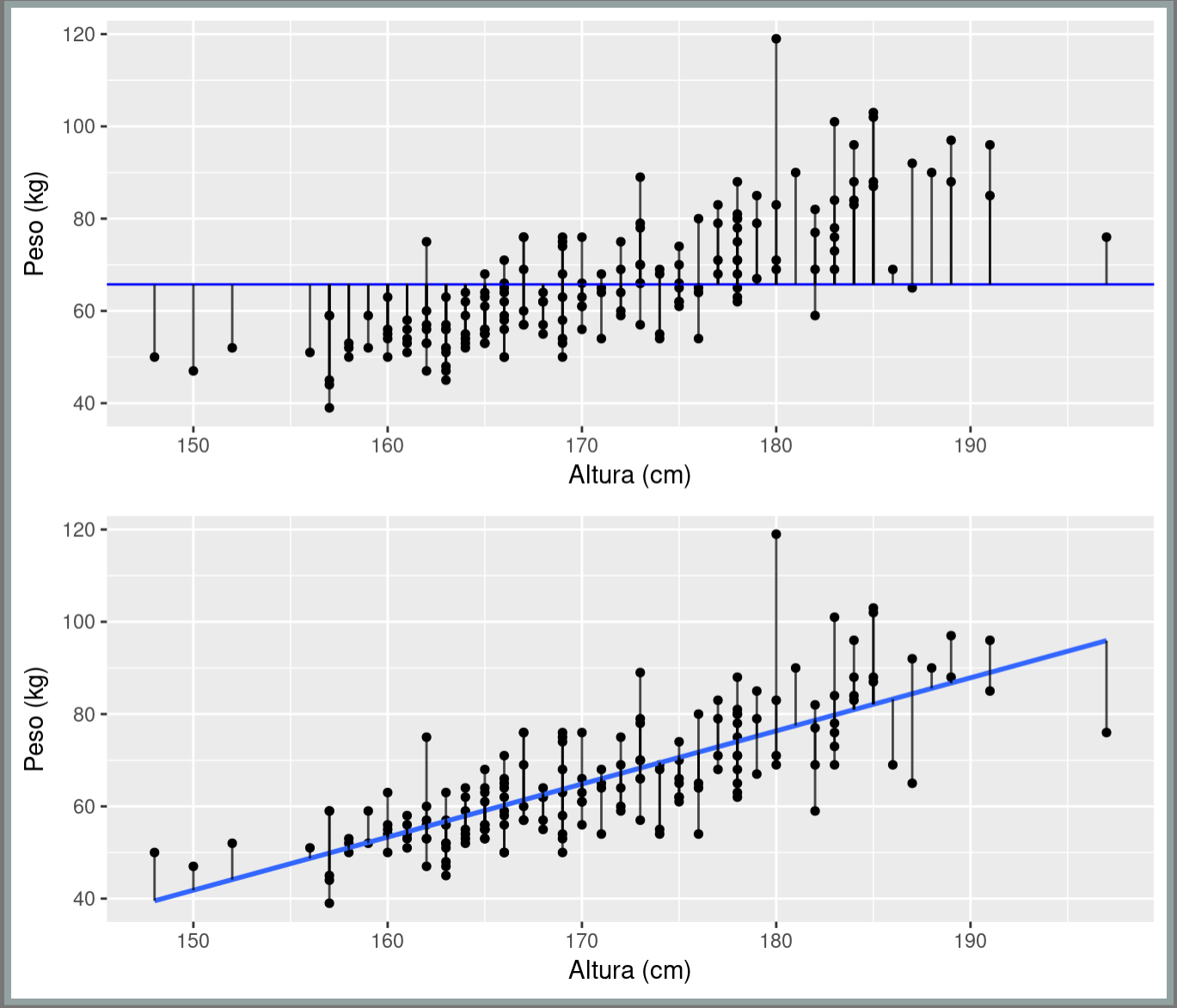


Modelo da Regressão

```
lmdavis <- lm(weight~height, data = Davi
summary(lmdavis)

##
## Call:
## lm(formula = weight ~ height, data =
##
## Residuals:
##     Min      1Q  Median      3Q     M
## -19.928  -5.406  -0.651   4.891  42.6
##
## Coefficients:
##             Estimate Std. Error t v
```

Regressão: peso ~ altura



anova(davisNull,lmdavis)

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
179	32367.75	NA	NA	NA	NA
178	13272.71	1	19095.04	256.0832	0

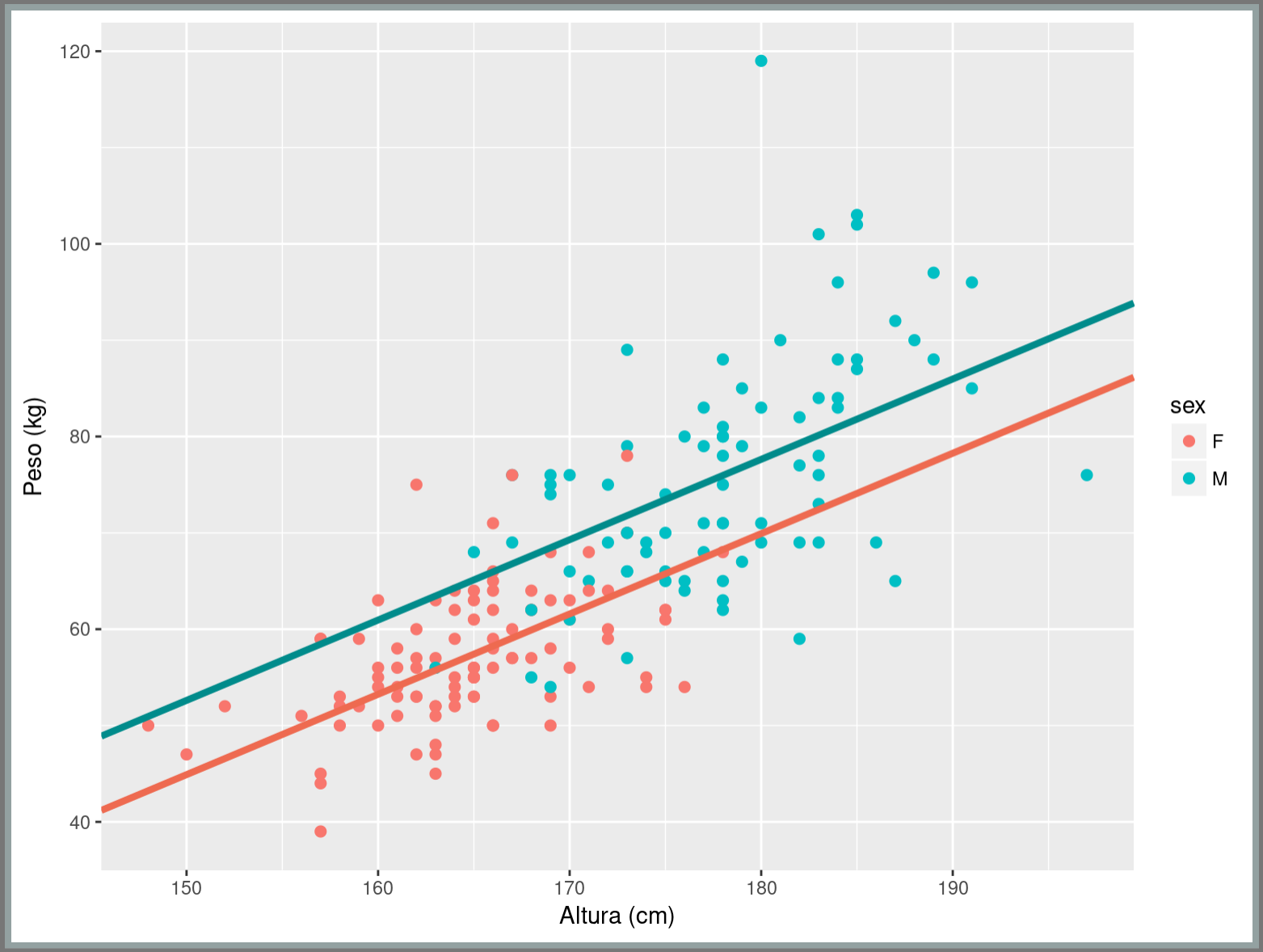
$$p_{valor} = 2.2e-16$$

$$p_{valor} = 2.2 * 10^{-16}$$

$$r^2 = 0.587$$

Modelo de Regressão:

lm(weight ~ height + sex, data = Davis)


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

sexo: variável dummy com dois níveis (mulher = 0, homem = 1)

```
lmdavis01 <- lm(weight~ height + sex, da
summary(lmdavis01)

##
## Call:
## lm(formula = weight ~ height + sex, d
##
## Residuals:
##     Min      1Q  Median      3Q     M
## -20.302  -4.808  -0.335   5.239  41.3
##
## Coefficients:
##             Estimate Std. Error t val
```

lm(weight ~ height + sex, data = Davis)

```
## (Intercept)      height        sexM
## -80.2107328   0.8340964   7.7070166
```

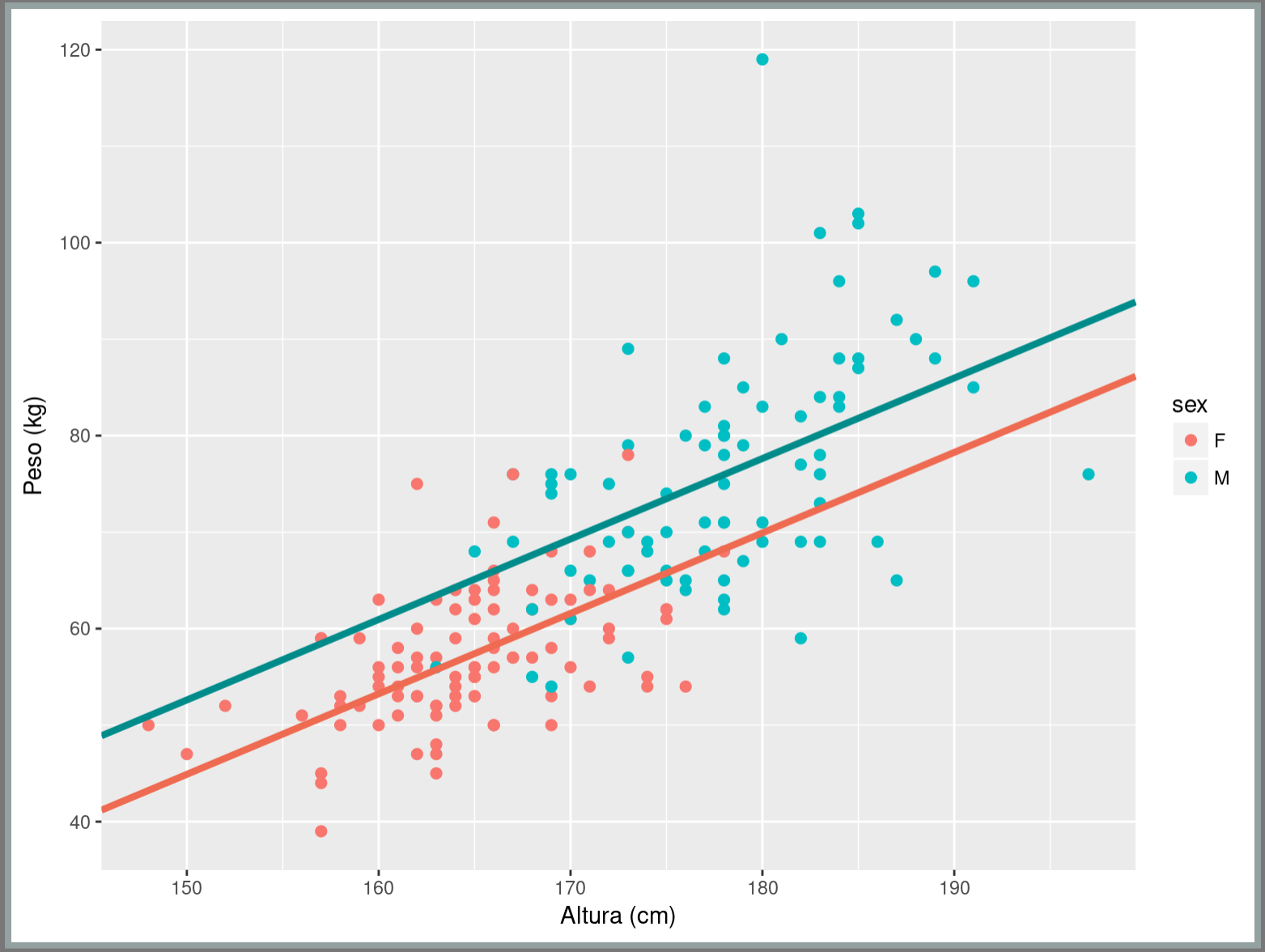
Mulher ($sex = 0$)

$$w_f = \hat{\alpha} + \hat{\beta}_s sex + \hat{\beta}_h * height$$
$$w_f = \hat{\alpha} + \hat{\beta}_h * height$$

Homem ($sex = 1$)

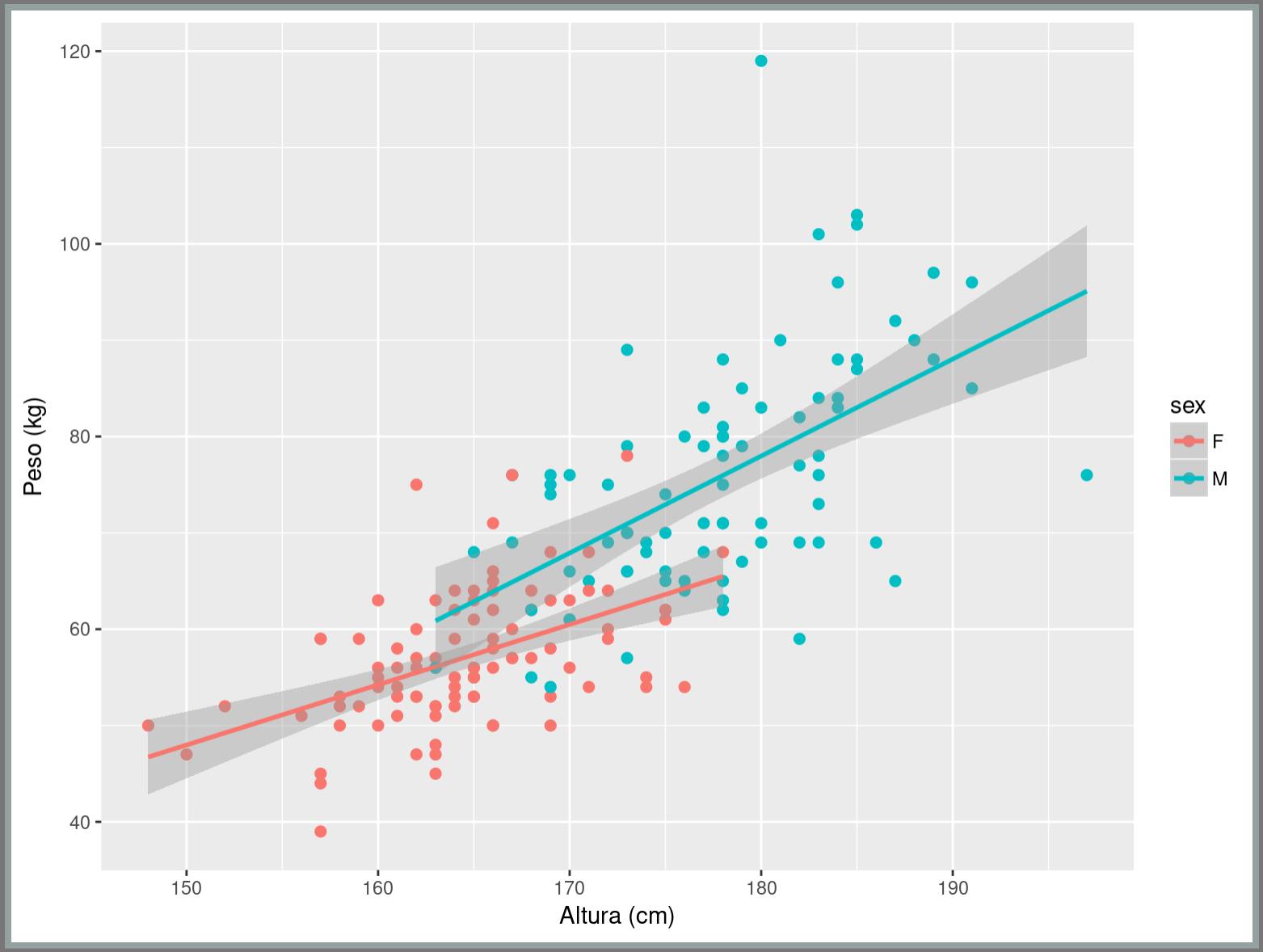
$$w_h = \hat{\alpha} + \hat{\beta}_s * sex + \hat{\beta} * height$$
$$w_h = \hat{\alpha} + \hat{\beta}_s + \hat{\beta}_h * height$$

lm(weight ~ height + sex)



Interação

```
lmdavisfull <- lm(weight ~ height + sex
```


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

```
lmdavisfull <- lm(weight ~ height + sex*
summary(lmdavisfull)

##
## Call:
## lm(formula = weight ~ height + sex *
##
## Residuals:
##     Min      1Q  Median      3Q     M
## -20.990  -4.548  -0.926   4.821  41.0
##
## Coefficients:
##             Estimate Std. Error t val
```

lm(weight ~ height + sex*height, data=Davis)

```
## (Intercept)      height        sexM h
## -45.7988220   0.6252035 -57.4326307
```

Mulher ($sex = 0$)

$$w = \hat{\alpha} + \hat{\beta}_s sex + \hat{\beta}_h height + \hat{\beta}_{s:h} sex * height$$
$$w_m = \hat{\alpha} + \hat{\beta}_h height$$

Homem ($sex = 1$)

$$w = \hat{\alpha} + \hat{\beta}_s sex + \hat{\beta}_h height + \hat{\beta}_{h:s} sex * height$$
$$w_h = \hat{\alpha} + \hat{\beta}_s + (\hat{\beta}_h + \hat{\beta}_{h:s}) * height$$

Predição do modelo

Uma mulher de 161cm de altura

$$w = \hat{\alpha} + \hat{\beta}_s sex + \hat{\beta}_h height + \hat{\beta}_{s:h} sex * height$$
$$sex = 0$$

```
(coefull <- coef(lmdavisfull))
```

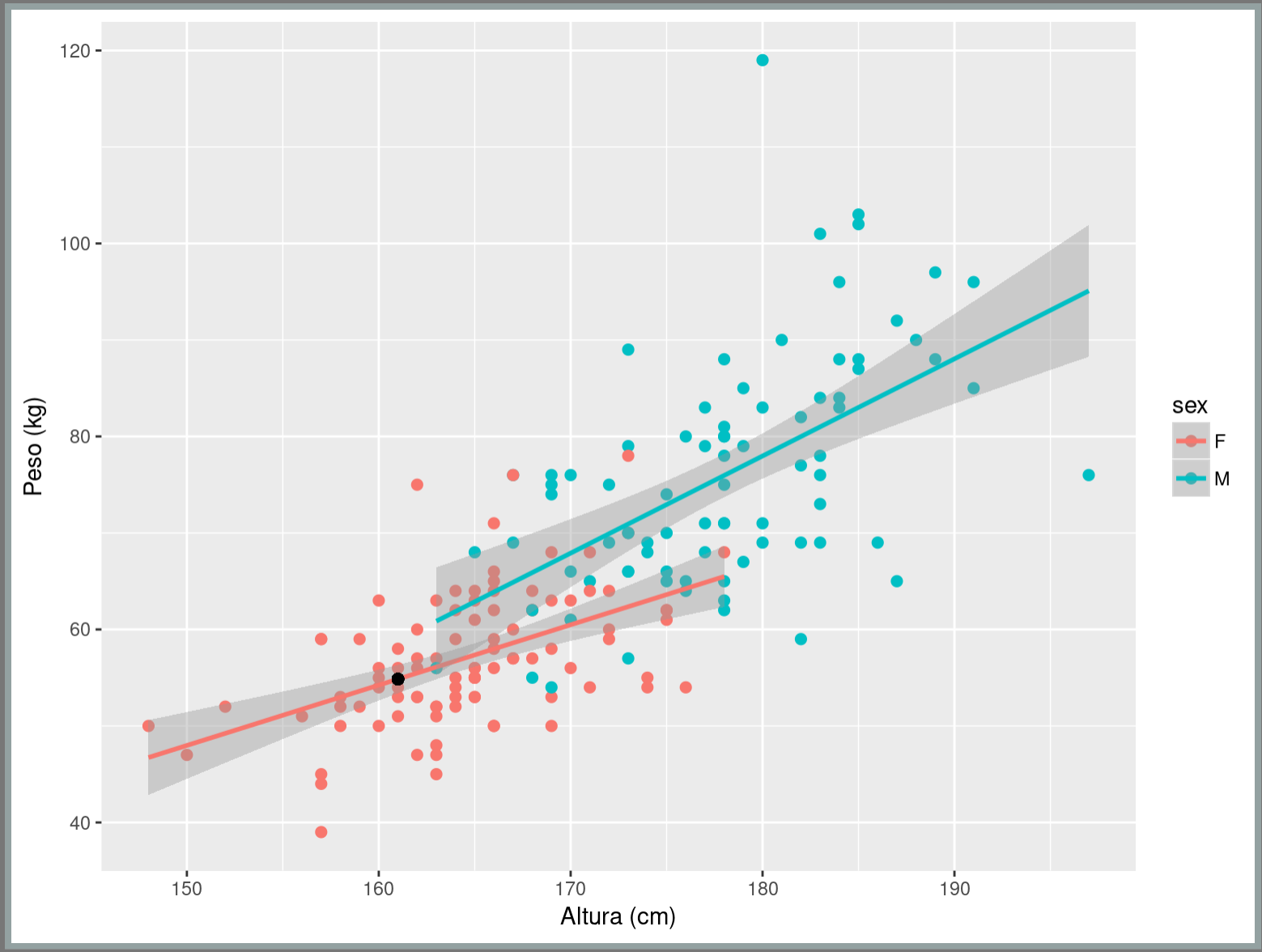
```
##  (Intercept)       height         sexM h
## -45.7988220    0.6252035 -57.4326307
```

```
predMulher <- coefull[1] + coefull[2] *
(predMulher <- as.numeric(predMulher))
```

```
## [1] 54.85893
```

lm(weight ~ height + sex*height, data=Davis)

- Uma mulher com 161cm de altura tem peso 54.86 kg.



Predito do Modelo

Homem com 182cm

$$w = \hat{\alpha} + \hat{\beta}_s sex + \hat{\beta}_h height + \hat{\beta}_{s:h} sex * height$$
$$sex = 1$$

```
coefull
```

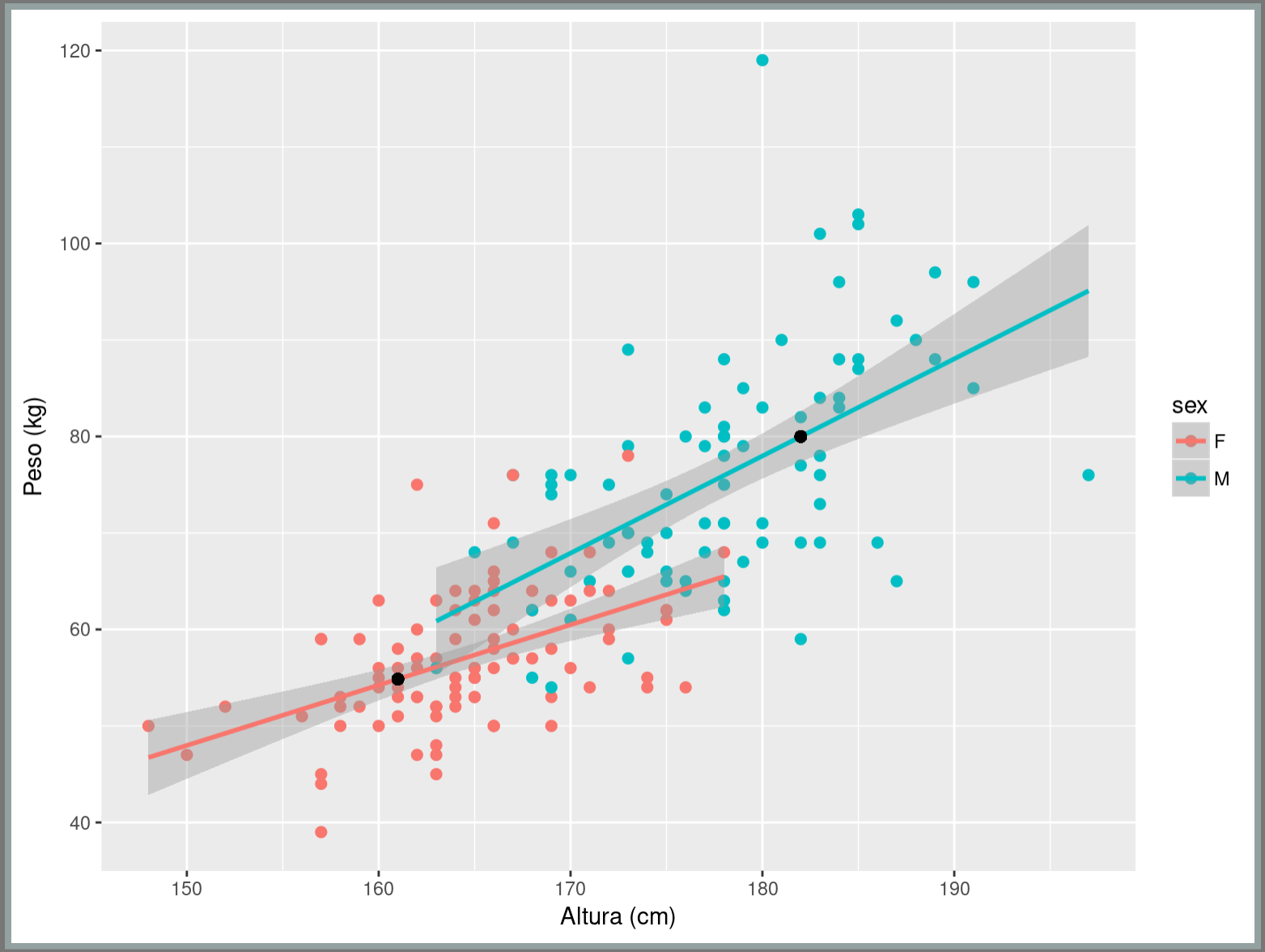
```
## (Intercept)        height          sexM h
## -45.7988220    0.6252035 -57.4326307
```

```
predHomem <- (coefull[1]+ coefull[3]) +
(predHomem <- as.numeric(predHomem))
```

```
## [1] 79.99018
```

lm(weight ~ height + sex*height, data=Davis)

- Um homem com 182cm de altura tem peso 79.99 kg.



Matrix do Modelo

```
Davis[1:2,1:3]
```

```
##    sex weight height
## 1    M     77    182
## 2    F     58    161
```

```
model.matrix(lmdavisfull)[1:2,]
```

```
##   (Intercept) height sexM height:sexM
## 1           1    182    1         182
## 2           1    161    0           0
```

```
coef(lmdavisfull)
```

```
## (Intercept)      height        sexM h
## -45.7988220   0.6252035 -57.4326307
```

Matrix do Modelo

```
model.matrix(lmdavisfull)[1:2,] %*% coef
```

```
##        [,1]
## 1 79.99018
## 2 54.85893
```

```
predict(lmdavisfull)[1:2]
```

```
##        1        2
## 79.99018 54.85893
```

Qual o melhor modelo?

Princípio da parcimônia (Navalha de Occam)

- devem ter menos parâmetros possível
- linear é melhor que não-linear
- reter menos pressupostos
- simplificado ao mínimo adequado
- explicações mais simples são preferíveis

Simplificação do modelo

Método do modelo cheio ao mínimo adequado

1. ajuste o modelo máximo (cheio)
2. simplifique o modelo:
    - inspecione os coeficientes (summary)
    - remova termos não significativos
3. ordem de remoção de termos:
    - interação não significativos (maior ordem)
    - termos quadráticos ou não lineares
    - variáveis explicativas não significativas
    - agrupe níveis de fatores sem diferença
    - ANCOVA: intercepto não significativoa -> 0

Simplificação do modelo: continuação

Compare o modelo anterior com o simplificado

A diferença não é significativa:

*** retenha o modelo mais simples**
*** continue simplificando**

A difereça é significativa

*** retenha o modelo complexo**
*** este é o modelo MINÍMO ADEQUADO**

Simplificando Modelo: exemplo

```
anova(lmdavisfull, lmdavis01)

## Analysis of Variance Table
##
## Model 1: weight ~ height + sex * heig
## Model 2: weight ~ height + sex
##   Res.Df   RSS Df Sum of Sq      F  P
## 1    176 11833
## 2    177 12069 -1   -235.82 3.5075 0.
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.
```

Simplificando Modelo: exemplo

```
anova(lmdavis01, lmdavis)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ height + sex
## Model 2: weight ~ height
##   Res.Df   RSS Df Sum of Sq     F
## 1    177 12069
## 2    178 13273 -1   -1203.5 17.65 4.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.
```

Modelo Mínimo Adequado

```
summary(lmdavis01)

##
## Call:
## lm(formula = weight ~ height + sex, d
##
## Residuals:
##     Min      1Q  Median      3Q     M
## -20.302  -4.808  -0.335   5.239  41.3
##
## Coefficients:
##             Estimate Std. Error t val
```

Modelo Mínimo Adequado

```
coef(lmdavis01)
```

```
## (Intercept)      height        sexM
## -80.2107328   0.8340964   7.7070166
```

```
confint(lmdavis01)
```

```
##                    2.5 %     97.5 %
## (Intercept) -113.44661 -46.974852
## height         0.63259   1.035603
## sexM           4.08671  11.327323
```

Diagnóstico do Modelo: plot(modelo)

```
par(mfrow = c(2,2))
plot((lmdavis01)
```

Diagnóstico: plot(modelo)