Modelo linear

2023

É apresentado um exemplo de ajuste em R com algumas ferramentas de diagnóstico aplicadas a um modelo linear normal.

```
# Separador decimal nos resultados: ","
options(OutDec = ",")
```

O conjunto de dados **iris** é usado. A variável resposta é "Sepal.Length" e as variáveis explicativas são "Sepal.Width" "Petal.Length" e "Petal.Width" referentes à espécie *Iris setosa*.

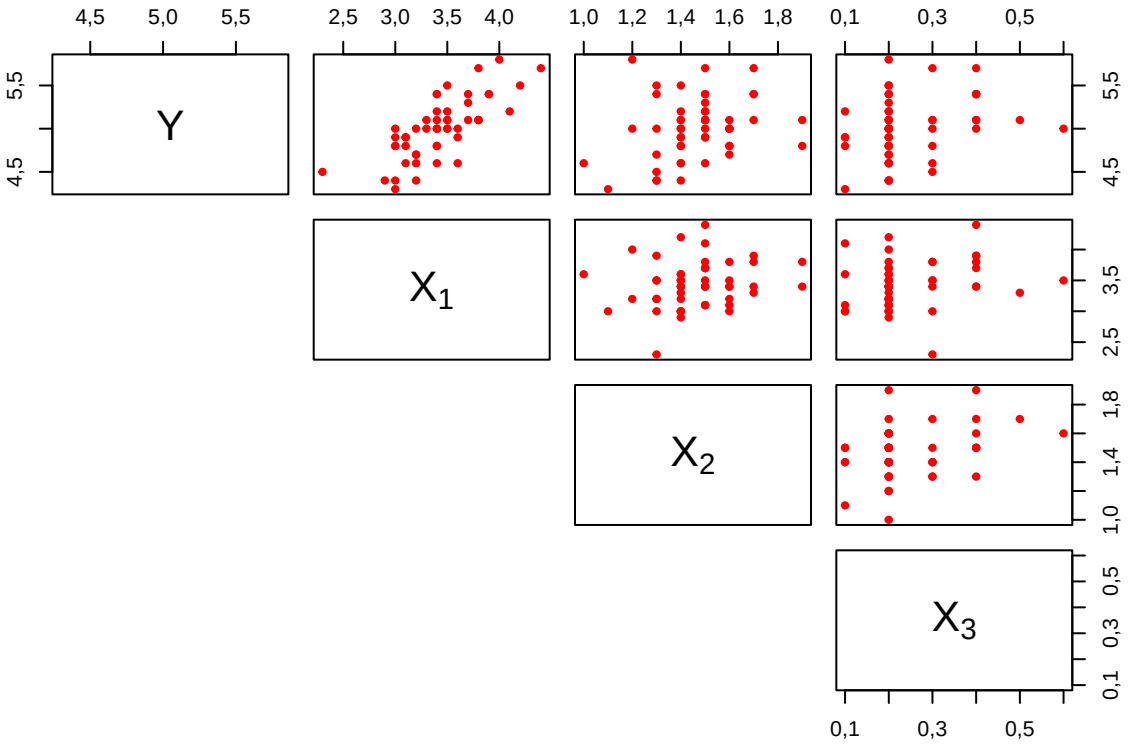
```
dados <- iris[1:50, 1:4]
colnames(dados) <- c("Y", "X1", "X2", "X3")
(n <- nrow(dados))
```

```
## [1] 50
```

```
summary(dados)
```

```
##        Y               X1              X2              X3
##  Min.   :4,300   Min.   :2,300   Min.   :1,000   Min.   :0,100
##  1st Qu.:4,800   1st Qu.:3,200   1st Qu.:1,400   1st Qu.:0,200
##  Median :5,000   Median :3,400   Median :1,500   Median :0,200
##  Mean   :5,006   Mean   :3,428   Mean   :1,462   Mean   :0,246
##  3rd Qu.:5,200   3rd Qu.:3,675   3rd Qu.:1,575   3rd Qu.:0,300
##  Max.   :5,800   Max.   :4,400   Max.   :1,900   Max.   :0,600
```

```
pairs(dados, labels = c("Y", expression(X[1]), expression(X[2]),
   expression(X[3])), pch = 20, lower.panel = NULL, col = "red")
```



Nota 1. Comente os gráficos acima.

Um modelo linear é ajustado com a função `lm`.

```
m1 <- lm(Y ~ X1 + X2 + X3, data = dados)
names(m1)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

Nota 2. Explique cada um dos componentes do objeto `m1` acima.

Nota 3. Refaça o ajuste utilizando a função `glm`.

```
summary(m1)
```

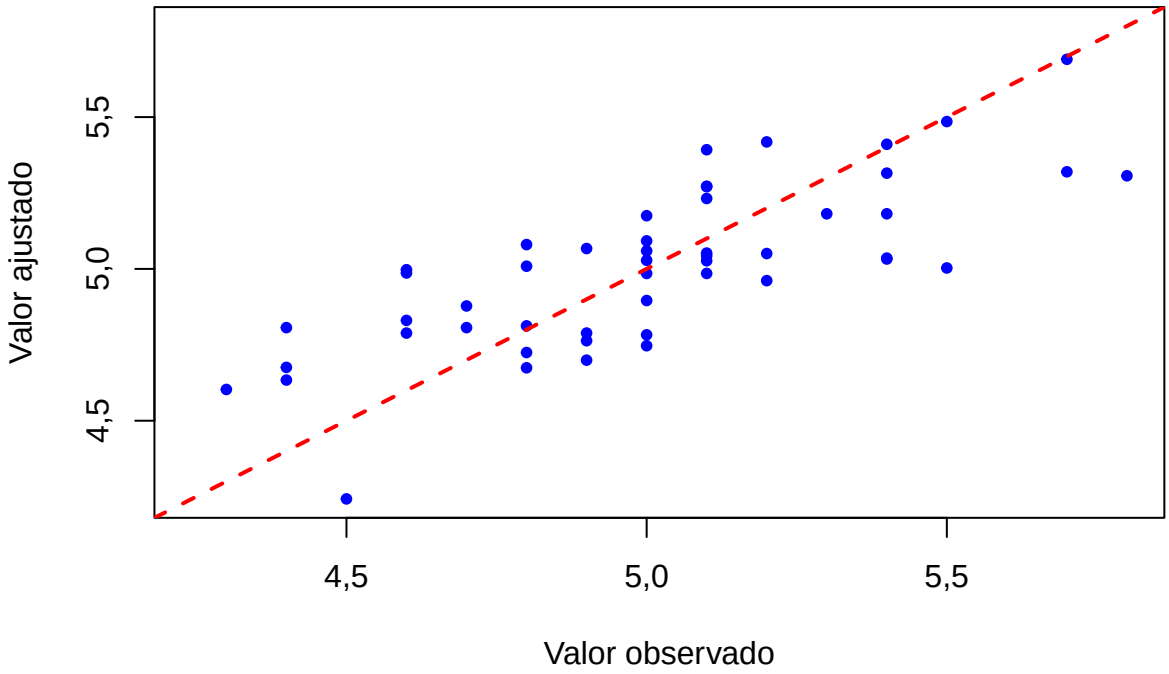
```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3, data = dados)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0,40662 -0,17721  0,01222  0,13388  0,49693
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2,35189    0,39287   5,986 3,03e-07 ***
## X1           0,65483    0,09245   7,083 6,83e-09 ***
## X2           0,23756    0,20802   1,142    0,259
## X3           0,25213    0,34686   0,727    0,471
## ---
## Signif. codes:  0 '***' 0,001 '**' 0,01 '*' 0,05 '.' 0,1 ' ' 1
```

```
##
## Residual standard error: 0,2371 on 46 degrees of freedom
## Multiple R-squared:  0,5751, Adjusted R-squared:  0,5474
## F-statistic: 20,76 on 3 and 46 DF,  p-value: 1,192e-08
```

Nota 4. Comente os resultados acima.

Os valores observados e ajustados pelo modelo são apresentados abaixo.

```
rm1 <- range(dados$Y, m1$fitted.values)
plot(dados$Y, m1$fitted.values, pch = 20, xlab = "Valor observado",
     ylab = "Valor ajustado", xlim = rm1, ylim = rm1, col = "blue")
abline(0, 1, lty = 2, col = "red", lwd = 2)
```

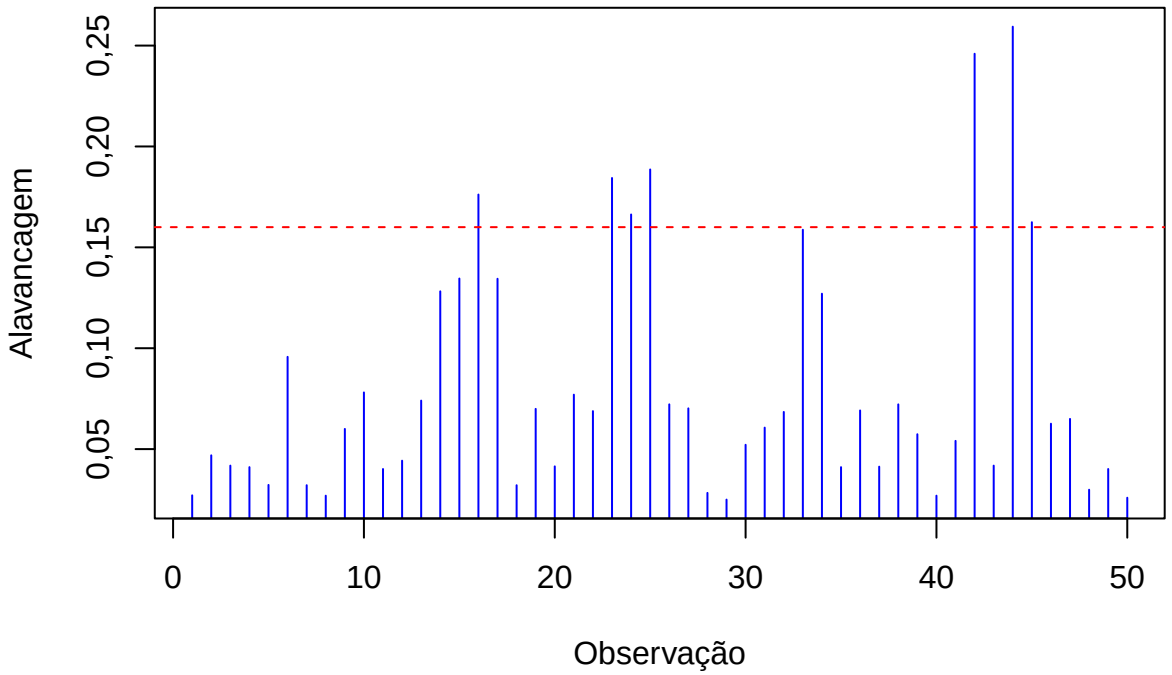


Nota 5. Justifique o uso da função `range` no trecho acima.

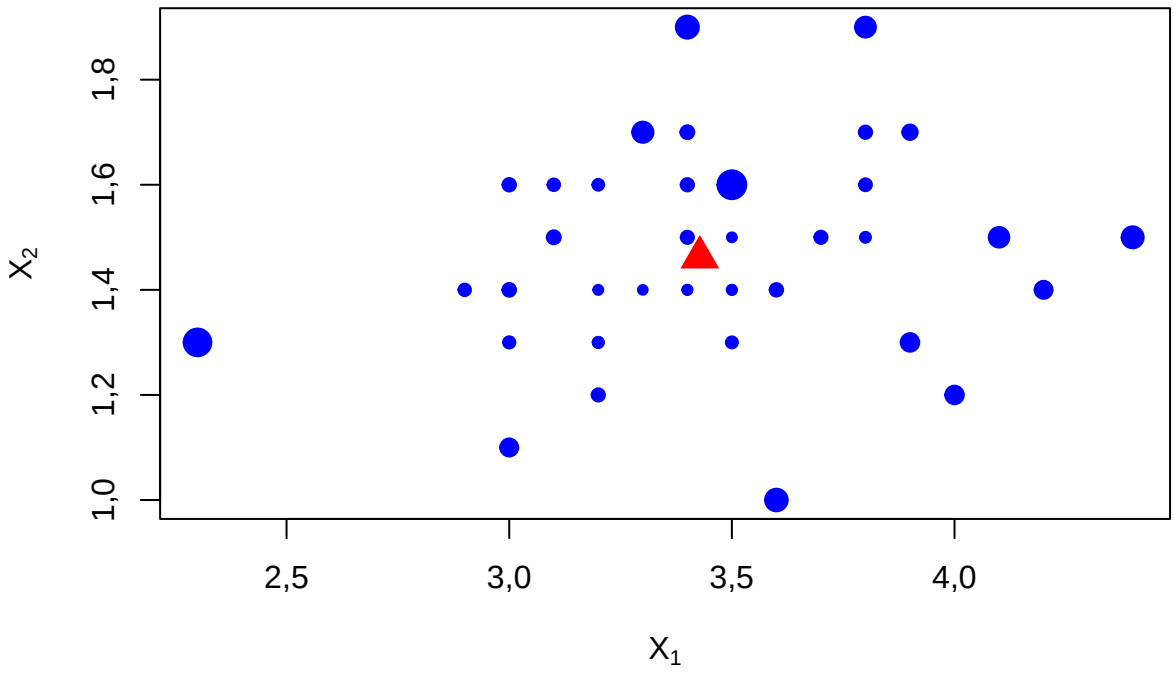
Nota 6. Comente o resultado do ajuste com base no gráfico acima.

Alguns gráficos representando a medida de alavancagem são mostrados abaixo.

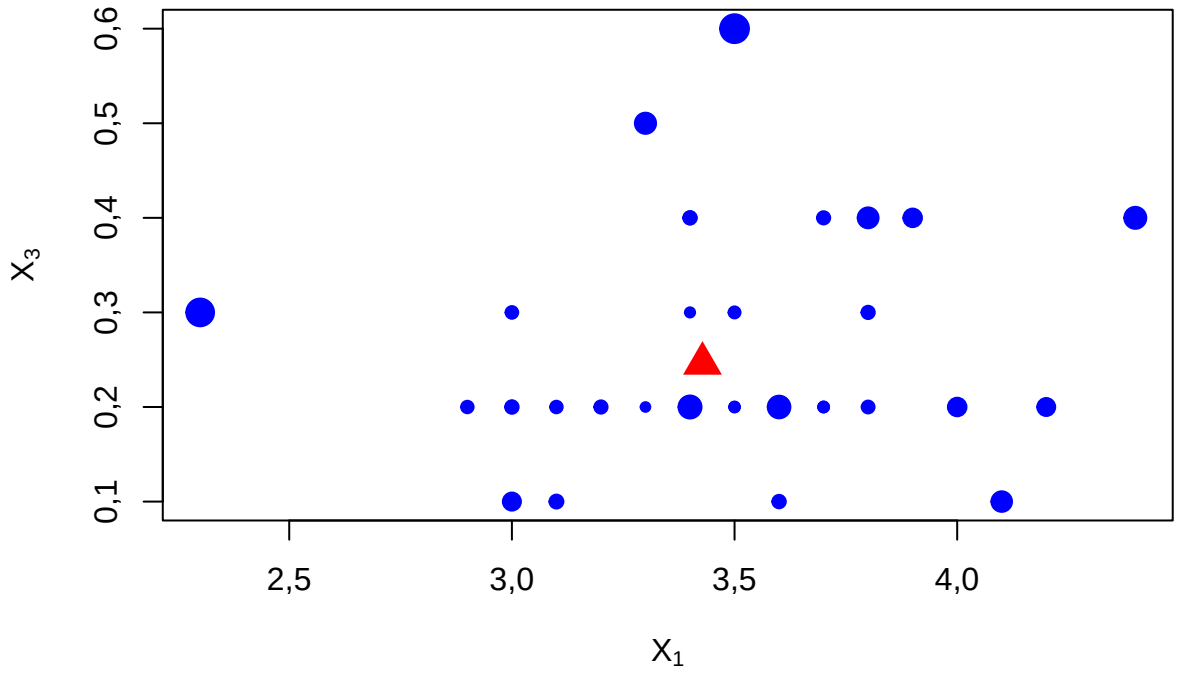
```
p <- length(m1$coefficients)
minh <- min(lm.influence(m1)$h)
cexh <- 2 * (lm.influence(m1)$h - minh) / (max(lm.influence(m1)$h) - minh) + 1
plot(lm.influence(m1)$h, type = "h", xlab = "Observação",
     ylab = "Alavancagem", col = "blue")
abline(h = 2 * p / n, lty = 2, col = "red")
```



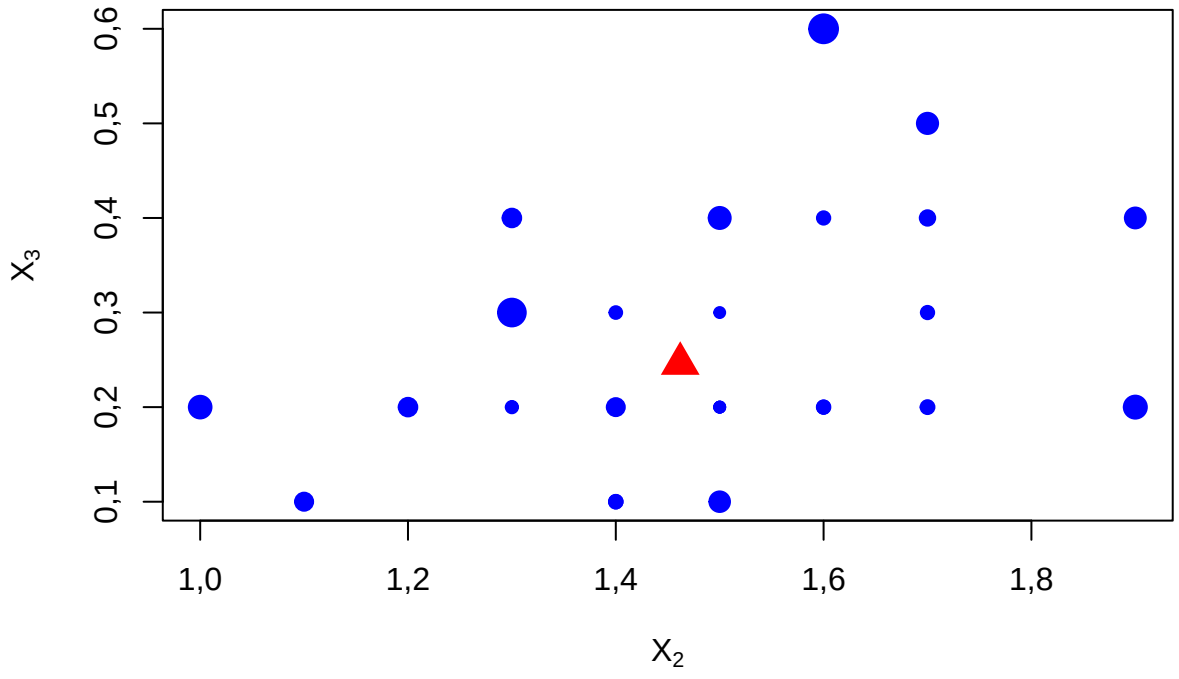

```
plot(dados[, 2], dados[, 3], pch = 20, cex = cexh,
     xlab = expression(X[1]), ylab = expression(X[2]), col = "blue")
points(mean(dados[, 2]), mean(dados[, 3]), pch = 17, col = "red",
       cex = 2)
```




```
plot(dados[, 2], dados[, 4], pch = 20, cex = cexh,
     xlab = expression(X[1]), ylab = expression(X[3]), col = "blue")
points(mean(dados[, 2]), mean(dados[, 4]), pch = 17, col = "red",
       cex = 2)
```




```
plot(dados[, 3], dados[, 4], pch = 20, cex = cexh,
     xlab = expression(X[2]), ylab = expression(X[3]), col = "blue")
points(mean(dados[, 3]), mean(dados[, 4]), pch = 17, col = "red",
       cex = 2)
```



Nota 7. Explique os resultados gerados pela função `lm.influence`.

Nota 8. Nos gráficos acima identifique as observações com alavancagem mais alta.

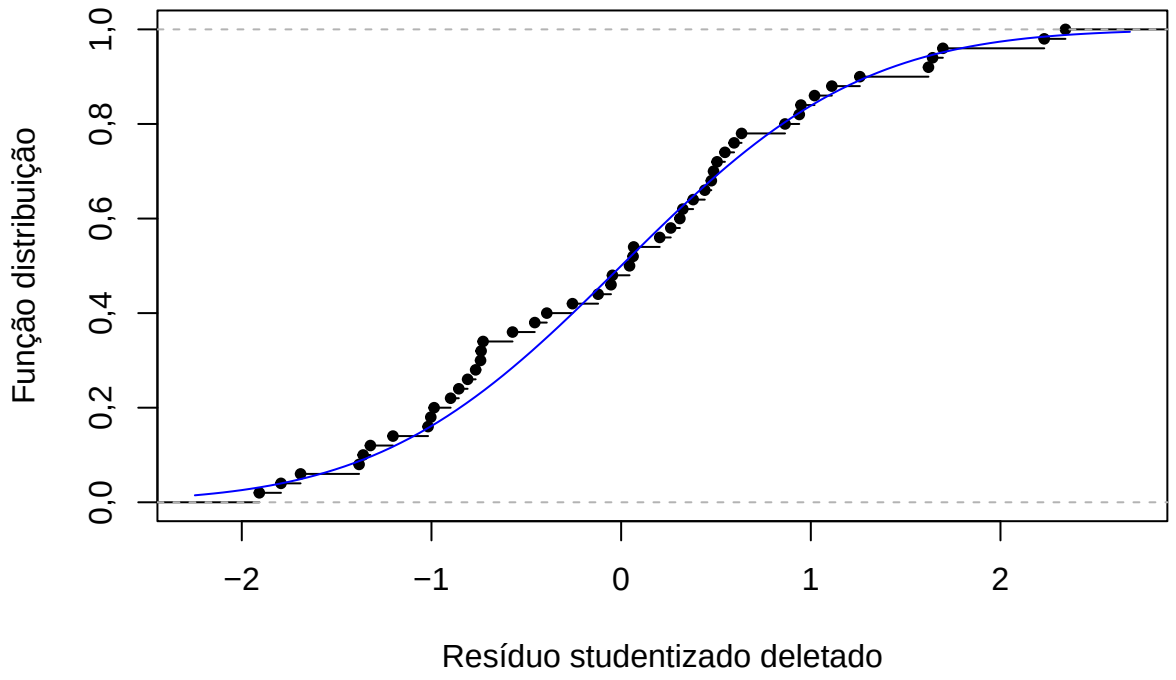
Em seguida apresentamos gráficos com os resíduos studentizados deletados.

```
# Resíduo studentizado deletado
tis <- m1$resid * sqrt((n - p - 1) /
```

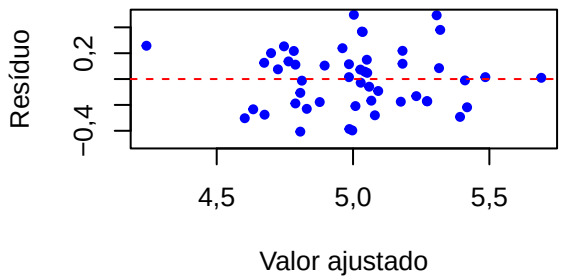
```
   (summary(m1)$s^2 * m1$df.residual * (1 - lm.influence(m1)$h) -
   m1$resid^2))

summary(tis)
```

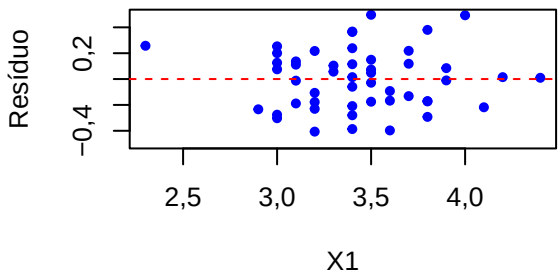
```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -1,907550 -0,798755  0,053678 -0,001755  0,583479  2,342551
```

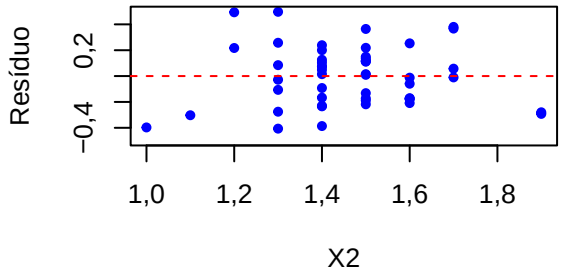
```
plot(ecdf(tis), main = "", xlab = "Resíduo studentizado deletado",
     ylab = "Função distribuição", pch = 20)
curve(pt(x, df = n - p - 1), col = "blue", add = TRUE)
```

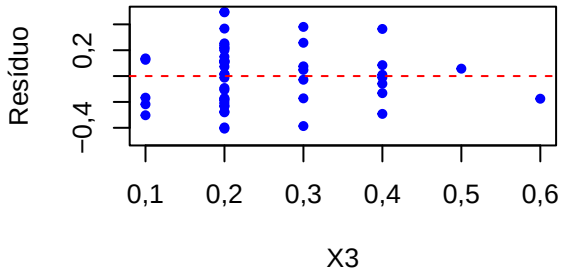



```
# Resíduos versus valores preditos e variáveis explicativas
par(mfrow = c(2, 2))
maxe <- max(abs(m1$resid))
plot(m1$fitted.values, m1$residuals, pch = 20, ylab = "Resíduo",
     xlab = "Valor ajustado", ylim = c(-maxe, maxe), col = "blue")
abline(h = 0, lty = 2, col = "red")
for (j in 2:4) {
   plot(dados[, j], m1$residuals, pch = 20, ylab = "Resíduo",
        xlab = names(dados)[j], ylim = c(-maxe, maxe), col = "blue")
   abline(h = 0, lty = 2, col = "red")
}
```

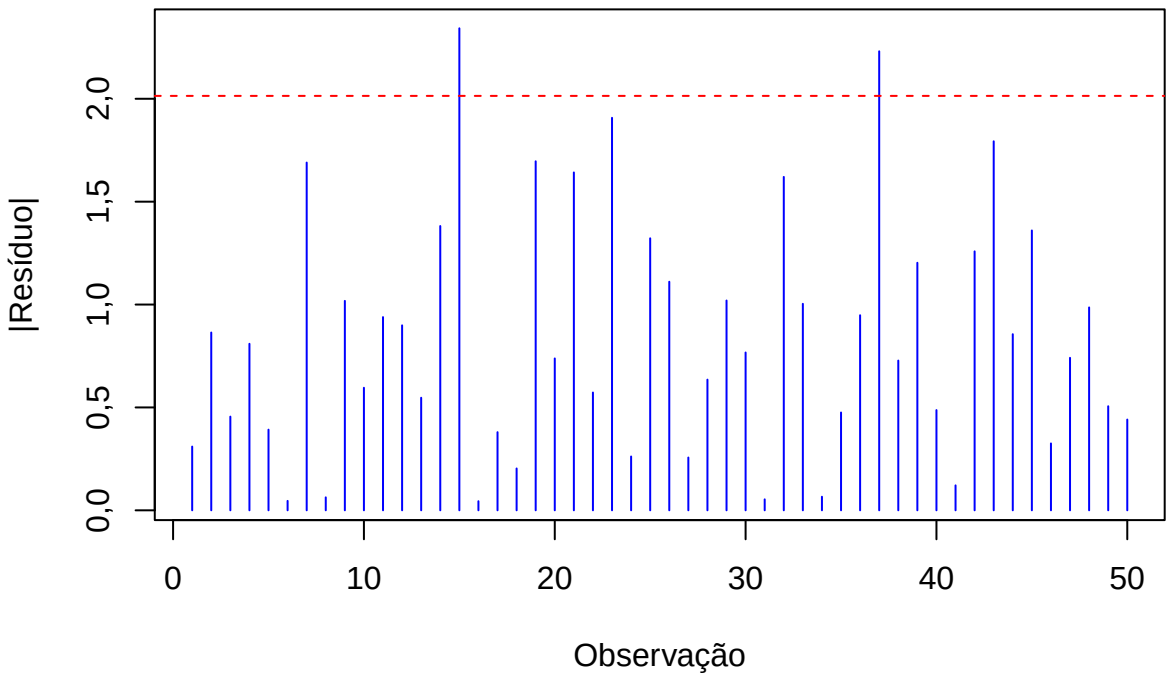




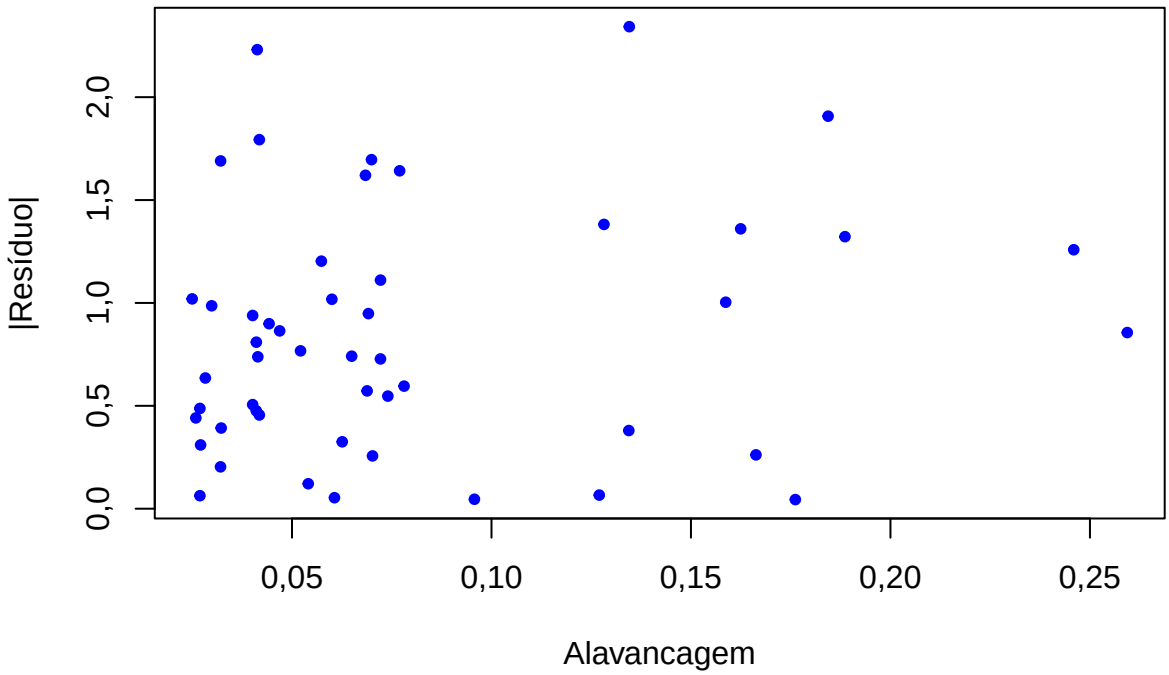





```
plot(abs(tis), type = "h", xlab = "Observação", col = "blue",
     ylab = "|Resíduo|")
abline(h = qt(0.975, df = n - p - 1), lty = 2, col = "red")
```




```
plot(lm.influence(m1)$h, abs(tis), pch = 20, xlab = "Alavancagem",
     ylab = "|Resíduo|", col = "blue")
```



Nota 9. Comente os gráficos acima relacionando-os com suposições do modelo.

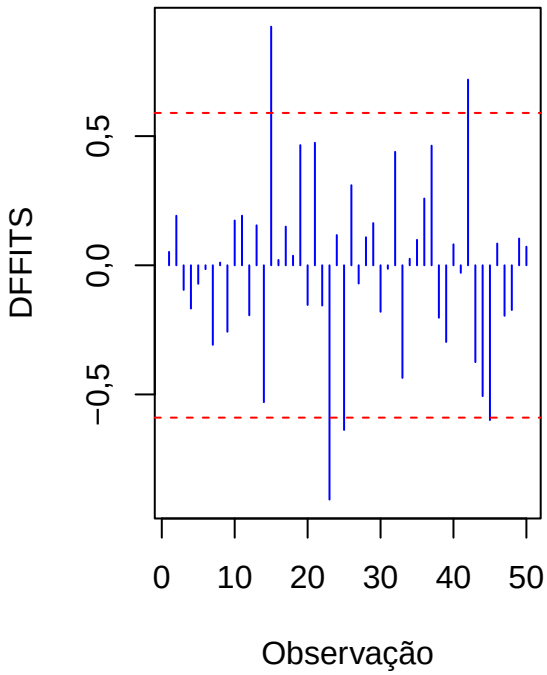
Nota 10. Apresente o gráfico de quantis dos resíduos studentizados deletados com envelope.

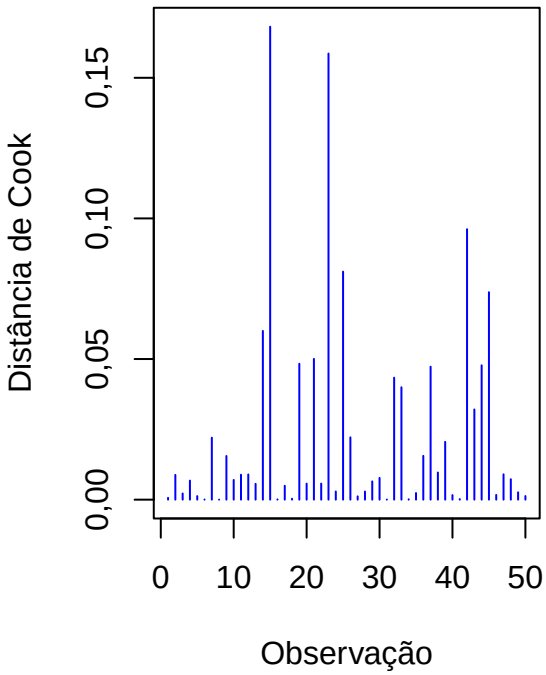
Para ajudar a identificar observações influentes, são apresentados gráficos de índices das medidas DFFITS e distância de Cook. Além destas, para cada coeficiente do modelo, é apresentada a mudança na estimativa decorrente da exclusão de uma observação de cada vez.

```
# DFFITS e distância de Cook
dffits <- tis * sqrt(lm.influence(m1)$h / (1 - lm.influence(m1)$h))
dffitsc <- 2 * sqrt(p / (n - p))
dcook <- m1$resid^2 * lm.influence(m1)$h /
   (p * summary(m1)$s^2 * (1 - lm.influence(m1)$h))
(dcookc <- qf(0.95, df1 = p, df2 = n - p))
```

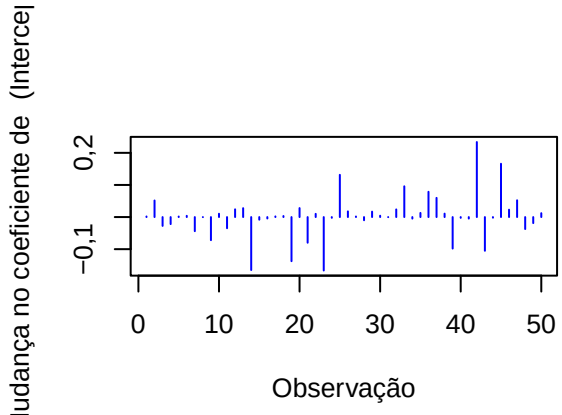
```
## [1] 2,574035
```

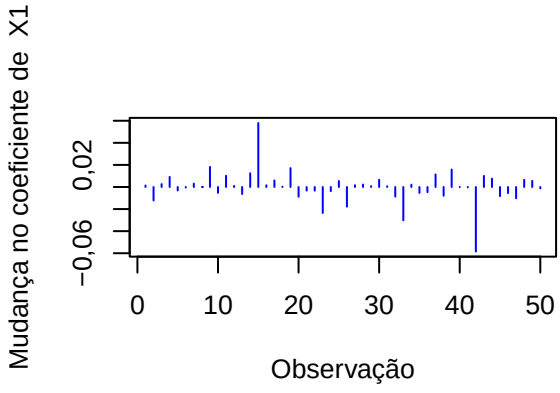
```
par(mfrow = c(1, 2))
plot(dffits, type = "h", xlab = "Observação", ylab = "DFFITS",
     col = "blue")
abline(h = c(-dffitsc, dffitsc), lty = 2, col = "red")
plot(dcook, type = "h", xlab = "Observação", ylab = "Distância de Cook",
     col = "blue")
abline(h = dcookc, lty = 2, col = "red")
```

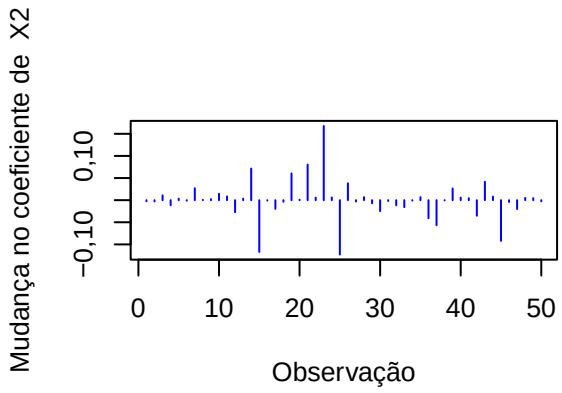


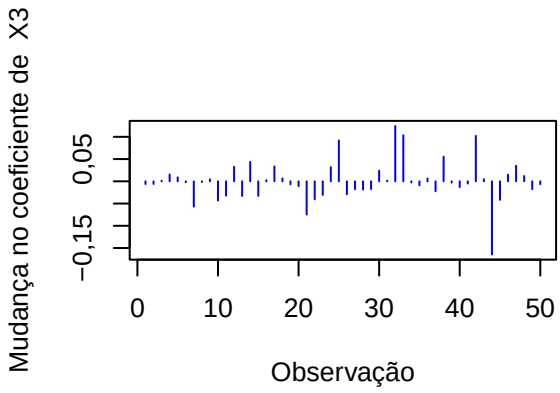



```
par(mfrow = c(2, 2))
for (j in 1:4) {
   plot(lm.influence(m1)$coef[, j], type = "h", xlab = "Observação",
      ylab = paste("Mudança no coeficiente de ",
      names(coef(m1))[j]), col = "blue")
}
```









Nota 10. Identifique as observações mais influentes nos gráficos abaixo. Tente apontar alguma diferença marcante delas em relação às demais.

Nota 11. Efetue o teste da hipótese H: $\beta_{X2} = \beta_{X3} = 0$. Caso seja possível simplificar o modelo, refaça o exemplo com o modelo mais simples.