SME0820 - Modelos de Regressão e Aprendizado Supervisionado I - Exercício 2

Brenda da Silva Muniz 11811603 Francisco Rosa Dias de Miranda 4402962
Heitor Carvalho Pinheiro 11833351 Mônica Amaral Novelli 11810453

Novembro de 2021

Este trabalho tem como objetivo ajustar um modelo de regressão linear múltipla a um conjunto de dados.

Conjunto de dados

O dataset contém dados de um experimento para determinar **pressão**, **temperatura**, **fluxo de CO2**, **umidade** e **tamanho da partícula de amendoim** sob o **rendimento total de aceite por lote de amendoim**. [rendimento (y)].

Significância: 97%

```
dados <- read_csv("dados/data-table-B7.csv", locale = locale(decimal_mark = ","))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   x1 = col_double(),
##   x2 = col_double(),
##   x3 = col_double(),
##   x4 = col_double(),
##   x5 = col_double(),
##   y = col_double()
## )
```

```
dim(dados)
```

```
## [1] 16  6
```

```
# Renomeando as colunas
names(dados) <- c("Pressao", "Temp", "FluxoCO2", "Umidade", "Tamanho", "y")

head(dados)
```

```
## # A tibble: 6 x 6
##   Pressao  Temp FluxoCO2 Umidade Tamanho     y
##     <dbl> <dbl>    <dbl>   <dbl>   <dbl> <dbl>
## 1     415    25        5      40    1.28    63
```

```
## 2     550    25        5      40    4.05    21
## 3     415    95        5      40    4.05    36
## 4     550    95        5      40    1.28    99
## 5     415    25       15      40    4.05    24
## 6     550    25       15      40    1.28    66
```

Temos cinco covariáveis quantitativas e a coluna y corresponde a nossa variável preditora que determina **o rendimento total de azeite por lote de amendoim**

1. Análise Descritiva dos dados

Para facilitar nosso trabalho em termos computacionais, vamos nomear nossas variáveis e concatená-las num data frame, sendo:

- Y : Rendimento total de azeite por lote de amendoim*.
- X_1 : Pressão
- X_2 : Temperatura
- X_3 : Fluxo de CO2
- X_4 : Umidade
- X_5 : Tamanho

```
x1 <- dados$Pressao
x2 <- dados$Temp
x3 <- dados$FluxoCO2
x4 <- dados$Umidade
x5 <- dados$Tamanho
y <- dados$y

tabela01 <- data.frame(cbind(x1, x2, x3, x4, x5, y))
```

Devido a complexidade das fórmulas envolvidas para realizarmos uma regressão linear múltima, utilizaremos uma abordagem matricial. Desse modo, poderemos escrever

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_k x_{ki} + \varepsilon_i, i = 1, ..., 5$$

com:

$$\begin{cases} y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{21} + ... + \beta_k x_{k1} + \varepsilon_1 \\ y_2 = \beta_0 + \beta_1 x_{12} + \beta_2 x_{22} + ... + \beta_k x_{k2} + \varepsilon_2 \\ y_3 = \beta_0 + \beta_1 x_{13} + \beta_2 x_{23} + ... + \beta_k x_{k3} + \varepsilon_3 \\ y_4 = \beta_0 + \beta_1 x_{14} + \beta_2 x_{24} + ... + \beta_k x_{k4} + \varepsilon_4 \\ y_5 = \beta_0 + \beta_1 x_{15} + \beta_2 x_{25} + ... + \beta_k x_{k5} + \varepsilon_5 \end{cases}$$

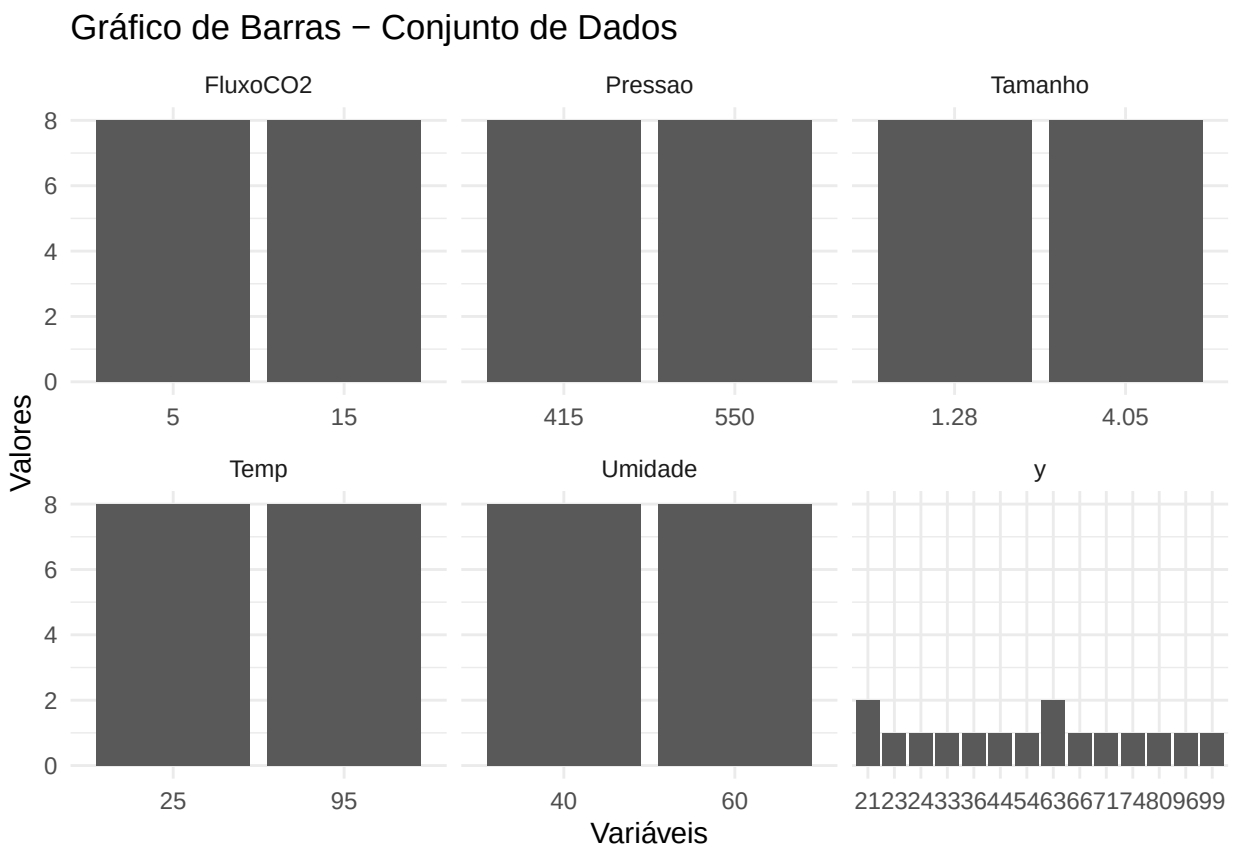
Assim, alocamos essas equações em dois vetores colunas (5x1), fazendo:

```r
n <- length(dados$y)

X <- matrix(c(rep(1, n), x1, x2, x3, x4, x5), ncol = 6, nrow = n, byrow = FALSE)

Y <- matrix(y, ncol = 1, nrow = n)

k <- ncol(X) - 1

p <- k + 1
```

Uma vez definidos tais pontos, podemos iniciar nossa análise descritiva dos dados, fazendo a montagem dos gráficos para explorar o comportamento das variáveis de nosso dataset.

Gráficos de barras

```r
dados %>%
  pivot_longer(cols = everything()) %>%
  ggplot() +
  geom_bar(aes(x = as_factor(value)), stat = "count") +
  facet_wrap(~name, scales = "free_x") +
  labs(
    x = "Variáveis",
    y = "Valores",
    title = "Gráfico de Barras - Conjunto de Dados"
  ) +
  theme_minimal()
```



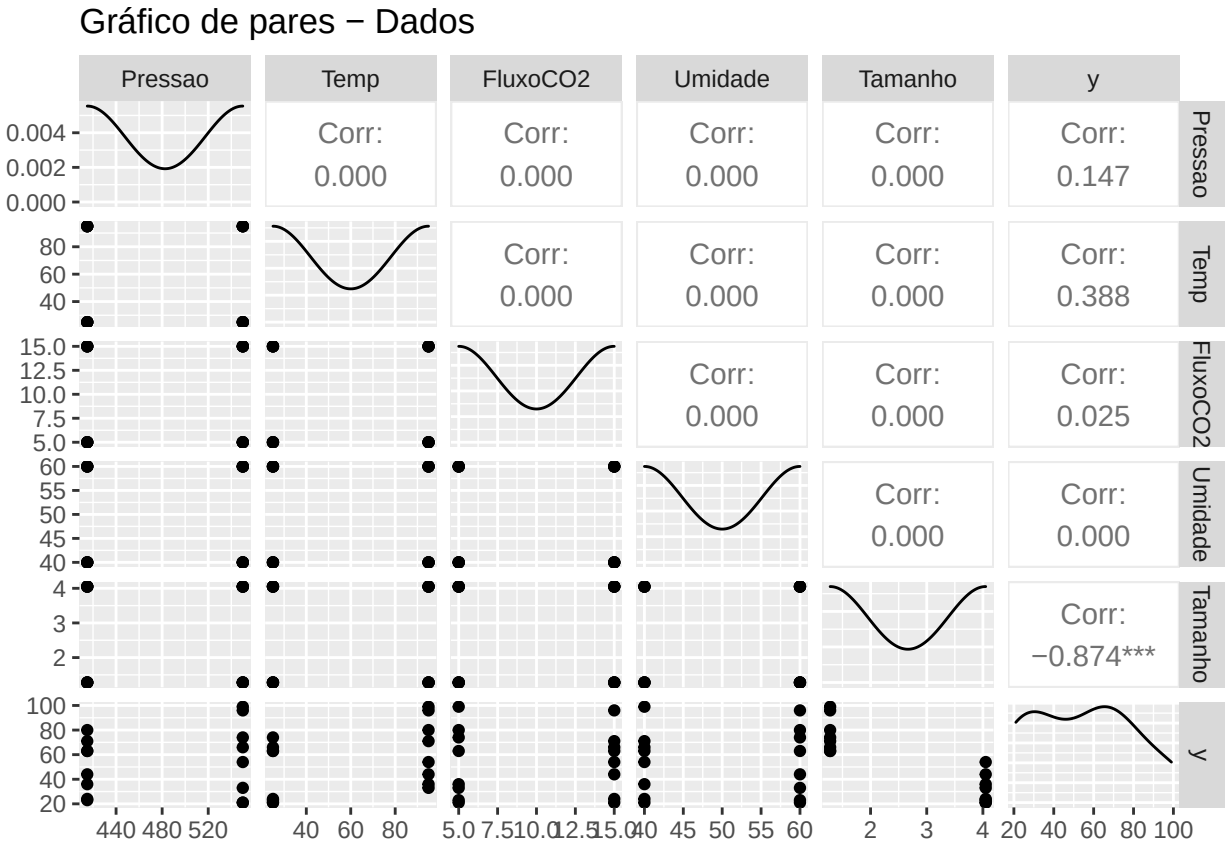
A partir dos gráficos de barras, podemos ver que nossas cinco covariáveis, apesar de serem quantitativas, assumem apenas dois valores, com a mesma proporção. A única variável que assume mais valores do que isso é y, que apararenta ter uma distribuição quase uniforme.

Outros gráficos que comparam as relações entre nossas variáveis é o gráfico de coordenadas paralelas e nossa matriz de correlação, ambos também explicitando a falta de correlação entre as covariáveis.

Correlação de Pearson entre as covariáveis e y

Fazendo uso das correlações, podemos dispor graficamente uma matriz de gráficos para expor as relações entre as variáveis, de modo que teremos densidades de frequência nas diagonais, gráficos de dispersão no painel triangular inferior e coeficientes das correlações no superior, de modo que o tamanho dos números é condicionado ao valor da correlação.

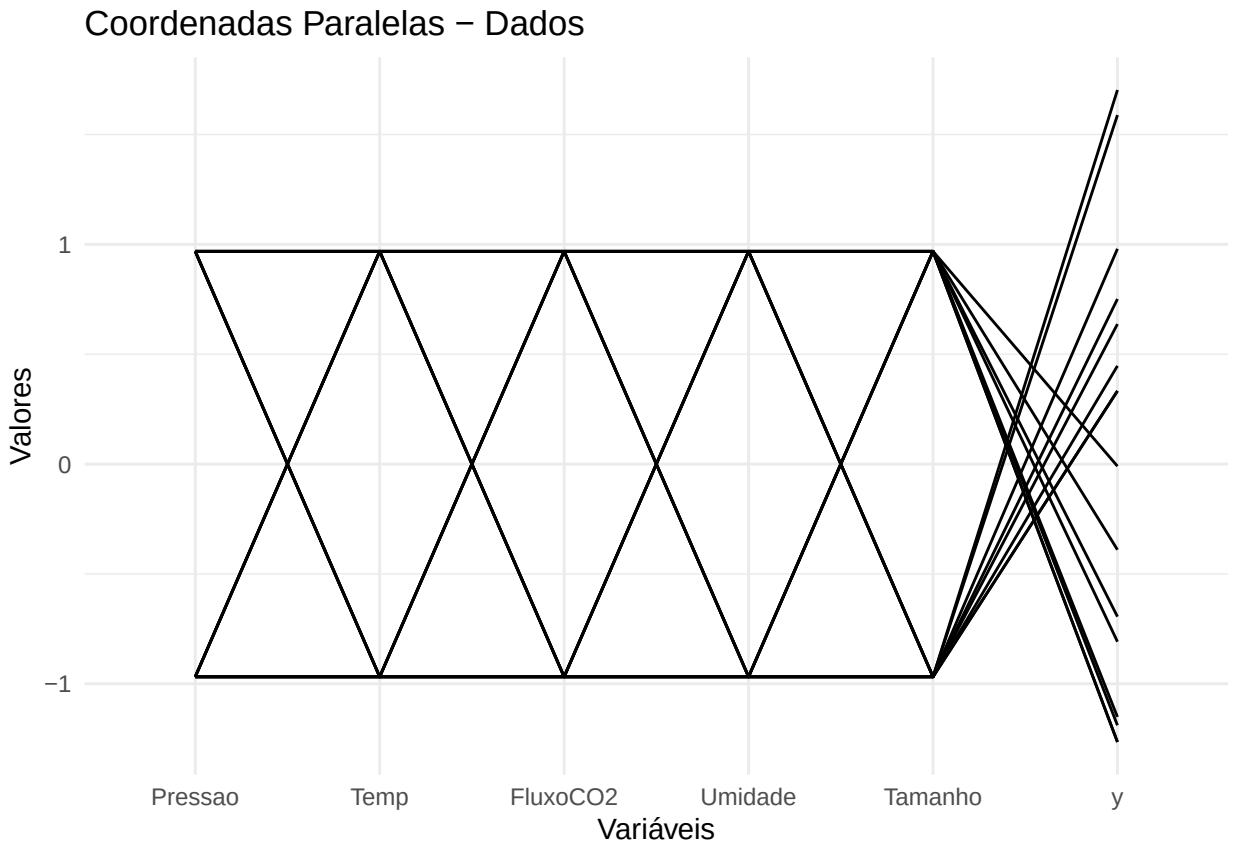
```
ggpairs(dados) + ggtitle("Gráfico de pares - Dados")
```



No gráfico de pares acima, podemos observar as correlações (ou ausência delas) de todas as covariáveis entre si e com a variável preditora. Analisando esses resultados, vemos que nenhuma das covariáveis se correlacionam entre si. Além disso, a maioria apresenta uma correlação muito baixa com a variável preditora - com exceção de x5 (Tamanho) com y.

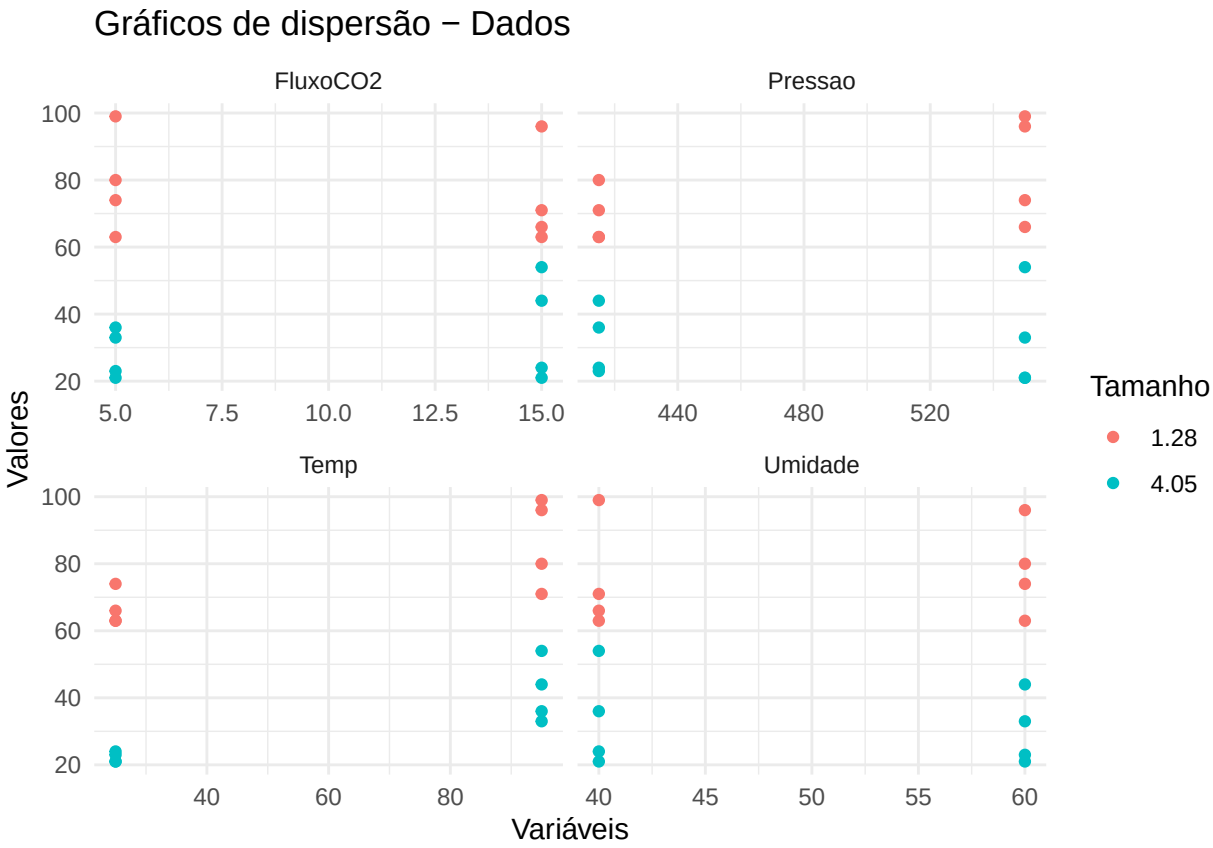
Essa ausência de correlação pode ser explicada pelo comportamento em "X" da maior parte das covariáveis, que também pode ser notado através do gráfico de coordenadas paralelas:

```
ggparcoord(dados) + labs(
  x = "Variáveis",
  y = "Valores",
  title = "Coordenadas Paralelas - Dados"
) +
  theme_minimal()
```



Definindo a covariável Tamanho como mapeamento para cor, podemos dispor outra versão dos gráficos de pares:

```
dados %>%
  pivot_longer(!c(Tamanho, y)) %>%
  ggplot(aes(y = y, color = as_factor(Tamanho))) +
  geom_point(aes(x = value)) +
  facet_wrap(~name, scales = "free_x") +
  labs(
    x = "Variáveis",
    y = "Valores",
    title = "Gráficos de dispersão - Dados",
    color = "Tamanho"
  ) +
  theme_minimal()
```



Note como a covariável Tamanho foi capaz de separar bem as variáveis no eixo y, enquanto que o mesmo feito não foi alcançado no eixo x. Temos aqui fortes indícios de independência entre as covariáveis, e o melhor modelo talvez não seja o que contenha todas elas, como veremos mais adiante.

2. Ajuste do modelo de Regressão Linear Múltipla

Ajustaremos um modelo de regressão utilizando todas as covariáveis presentes em nosso conjunto de dados.

Definimos um modelo de regressão linear múltipla com k covariáveis através da fórmula $Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_k X_k + \xi$, onde $\beta_j, j = 1, \ldots, k$ são os coeficientes de regressão e representam a mudança esperada na variável resposta Y por unidade de mudança em X_i quando todas as outras covariáveis $X_i (i \neq j)$ são mantidas constantes.

Podemos também representar o modelo da seguinte forma:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\xi}$$

Onde:

$\boldsymbol{Y}$ = Vetor de $(n \times 1)$ observações;

$\boldsymbol{X}$ = Matriz $(n \times p)$ de covariáveis;

$\boldsymbol{\beta}$ = Vetor $P \times 1$ de coeficientes de regressão;

ξ = Vetor n $\times$ 1 de erros aleatórios;

$p = k + 1$.

```r
# Definindo os betas do modelo de regressão múltipla
betas <- solve(t(X) %*% X) %*% t(X) %*% Y
betas
```

```
##               [,1]
## [1,]  5.207905e+01
## [2,]  5.555556e-02
## [3,]  2.821429e-01
## [4,]  1.250000e-01
## [5,] -1.554312e-15
## [6,] -1.606498e+01
```

```r
# Definindo a matrix C_jj
C_jj <- solve(t(X) %*% X)
C_jj
```

```
##              [,1]          [,2]          [,3]          [,4]          [,5]
## [1,]  5.483580991 -6.618656e-03 -3.061224e-03 -2.500000e-02 -3.125000e-02
## [2,] -0.006618656  1.371742e-05  2.449395e-20 -1.693022e-19 -1.469637e-20
## [3,] -0.003061224  7.965567e-21  5.102041e-05 -7.830471e-20 -6.797284e-21
## [4,] -0.025000000 -4.206704e-19 -2.081668e-19  2.500000e-03 -2.636780e-18
## [5,] -0.031250000 -1.615461e-19  2.890702e-34 -2.220446e-18  6.250000e-04
## [6,] -0.086831576 -2.232725e-18 -6.782534e-19 -6.169755e-18 -9.338984e-18
##               [,6]
## [1,] -8.683158e-02
## [2,] -8.271877e-19
## [3,] -3.825863e-19
## [4,] -3.124455e-18
## [5,] -3.905569e-18
## [6,]  3.258220e-02
```

```r
# Definindo uma matrix para os betas
betas <- matrix(data = betas, nrow = length(betas), ncol = 1, byrow = FALSE)
rownames(betas) <- c("beta0", "beta1", "beta2", "beta3", "beta4", "beta5")
betas
```

```
##                [,1]
## beta0  5.207905e+01
## beta1  5.555556e-02
## beta2  2.821429e-01
## beta3  1.250000e-01
## beta4 -1.554312e-15
## beta5 -1.606498e+01
```

```r
# Modelo do R
lm(formula = y ~ ., data = dados)
```

```
##
## Call:
## lm(formula = y ~ ., data = dados)
##
```

```
## Coefficients:
## (Intercept)      Pressao         Temp     FluxoCO2      Umidade      Tamanho
##   5.208e+01    5.556e-02    2.821e-01    1.250e-01    8.774e-17   -1.606e+01
```

Em relação à diferença entre o EMV e o estimador de mínimos quadrados de beta é, basicamente, que, estando sob a suposição de erro normal, como normalmente assumimos em regressão linear, o estimador de mínimos quadrados e a máxima verossimilhança são os mesmos. Ou seja, minimizar o erro ao quadrado é equivalente a maximizar a probabilidade caso as seguintes condições de regularidade sejam cumpridas:

1. O modelo de regressão é linear nos parâmetros,

ou seja, o modelo na população pode ser escrito como $y = \beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_k x_k+\varepsilon$, em que $\beta_0, \beta_1, ..., \beta_k$ são parâmetros desconhecidos e ε é o termo de erro aleatório não observável.

2. Amostragem Aleatória

ou seja, possuimos uma amostra aleatória de n observações $(x_{1i}, x_{2i}, ..., x_{ki}, y_i), i = 1, 2, ..., n$, desse modelo de regressão descrito acima - linear. - Ausência de Colinearidade Perfeita. ou seja, temos que, como nenhum regressor é constante; não teremos uma relação linear perfeita entre os mesmos.

3. Média Condicional Zero ou seja, valor esperado do termo de erro aleatório, ε, condicionado na matriz de explicação X, tem que ser igual a zero. Logo, $E(\varepsilon|X) = 0$. Em um modelo linear, se os erros pertencem a uma distribuição normal, os estimadores de mínimos quadrados também são os estimadores de probabilidade máxima.

3. Estimação de σ^2

Temos que:

$$SQ_{res} = \sum_{i=1}^{n}(Y_i - \hat{Y})^2 = (\boldsymbol{y} - \hat{\boldsymbol{y}})^T(\boldsymbol{y} - \hat{\boldsymbol{y}}) =$$

$$(\boldsymbol{y} - X\hat{\beta})^T(\boldsymbol{y} - X\hat{\beta}) = (\boldsymbol{y}^T - X^T\hat{\beta}^T)(\boldsymbol{y} - X\hat{\beta}) =$$

$$\boldsymbol{y^T y} - \boldsymbol{Y^T} X\hat{\beta} - \hat{\beta}^T X^T \boldsymbol{y} + \hat{\beta}^T X^T X\hat{\beta} =$$

$$\boldsymbol{y^T y} - 2\hat{\beta}^T X^T \boldsymbol{y} + \hat{\beta}^T X^T X\hat{\beta}$$

A partir disso, considerando que:
$\hat{\beta} = (X^T X)^{-1} X^T \boldsymbol{y}$; $(X^T X)\hat{\beta} = X^T Y$ e, portanto,

$$SQ_{res} = \boldsymbol{y^T y} - 2\hat{\beta}^T X^T \boldsymbol{y} + \hat{\beta}^T X^T \boldsymbol{y}$$

resultando em:

$$SQ_{res} = \boldsymbol{y}^T \boldsymbol{y} - \widehat{\boldsymbol{\beta}}^T X^T \boldsymbol{y}$$

Com isso,

```r
SQRes <- (t(Y) %*% Y) - (t(betas) %*% t(X) %*% Y)
SQRes
```

```
##          [,1]
## [1,] 650.5
```

Agora, tomando:

$$SQ_{res} = \boldsymbol{y}^T\boldsymbol{y} - \widehat{\boldsymbol{\beta}}^T X^T\boldsymbol{y}$$

podemos fazer:

$$SQ_{res} = \boldsymbol{y}^T\boldsymbol{y} - ((X^TX)^{-1}X^T\boldsymbol{y})^T X^T\boldsymbol{y} =$$

$$\boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^TX(X^TX)^{-1}X^T\boldsymbol{y} = \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^TH\boldsymbol{y}$$

Dessa forma:

$$E(SQ_{res}) = E(\boldsymbol{y}^T\boldsymbol{y}) - E(\boldsymbol{y}^TH\boldsymbol{y})$$

- Desenvolvendo $E(\boldsymbol{y}^T\boldsymbol{y})$:

$$E(\boldsymbol{y}^T\boldsymbol{y}) = E(\boldsymbol{y}^TI\boldsymbol{y}) = tr(I\sigma^2I) + (X\beta)^TIX\beta =$$
$$\sigma^2tr(I) + \beta^TX^TX\beta = n\sigma^2 + \beta^TX^TX\beta$$

- Desenvolvendo $E(\boldsymbol{y}^TH\boldsymbol{y})$:

$$E(\boldsymbol{y}^TH\boldsymbol{y}) = tr(H\sigma^2I) + (X\beta)^TX(X^TX)^{-1}X^TX\beta =$$

$$\sigma^2tr(H) + \beta^TX^TX(X^TX)^{-1}X^TX\beta = \sigma^2p + \beta^TX^TX\beta$$

Realizando tais substituições, chegamos em:

$$E(SQ_{res}) = n\sigma^2 + \beta^TX^TX\beta - (\sigma^2p + \beta^TX^TX\beta) = (n-p)\sigma^2$$

Logo, $\widehat{\sigma}^2 = \dfrac{SQres}{n-p} = QM_{res}$ é um estimador não viciado para sigmaˆ2.

Seu cálculo se dá por:

```r
p <- ncol(X)

# estimando o sigma^2
sigma2 <- SQRes / (n - p)
sigma2
```

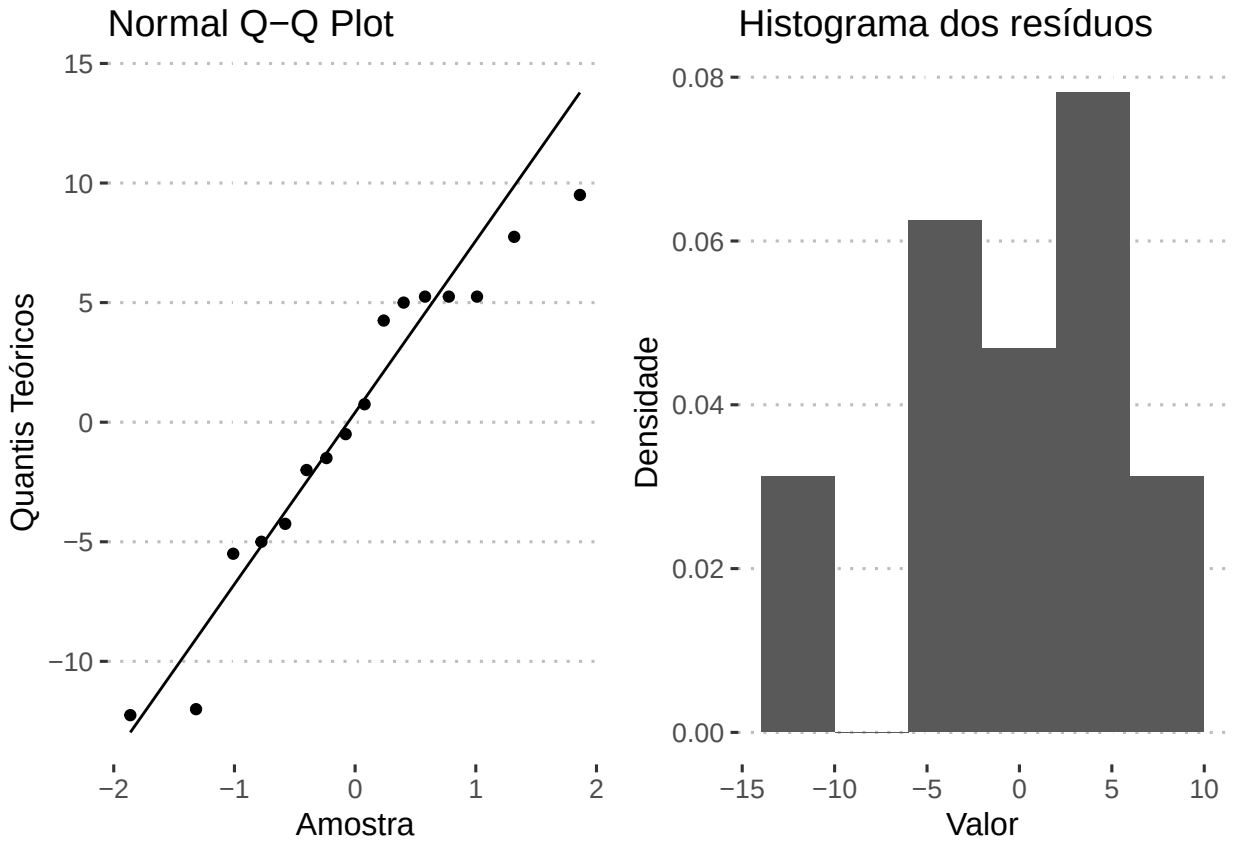
```
##          [,1]
## [1,] 65.05
```

4. ANOVA

Agora que ajustamos um modelo inicial, é necessário que verifiquemos se ele é adequado em explicar a variabilidade de nossa amostra. Vamos assumir que $\xi \sim N_n(0, \sigma^2 I)$. Precisamos agora verificar nossa suposição graficamente:

```r
# Obtendo uma estimativa para Y a partir do modelo ajustado
Y_est <- X %*% betas

# Cálculo dos resíduos
res <- Y - Y_est
```

```r
p <- ggplot(tibble(res), aes(sample = res)) +
  stat_qq() +
  stat_qq_line() +
  labs(
    x = "Amostra",
    y = "Quantis Teóricos",
    title = "Normal Q-Q Plot"
  ) +
  theme_pubclean()

q <- ggplot(tibble(res), aes(res)) +
  geom_histogram(aes(y = ..density..), binwidth = 4, stat = "bin") +
  labs(
    title = "Histograma dos resíduos",
    y = "Densidade",
    x = "Valor"
  ) +
  theme_pubclean()

grid.arrange(p, q, ncol = 2)
```



O Q-Q Plot nos mostra o quanto os resíduos estão distantes do esperado dado que os dados têm distribuição normal, enquanto que o histograma dos resíduos nos fornece aproximações a respeito da distribuição de ξ.

A partir dos gráficos acima, podemos notar que os resíduos não estão muito afastados dos quantis teóricos, embora sua distribuição seja ligeiramente assimétrica, conforme constatado no histograma. Podemos também utilizar o teste de Shapiro-Wilk para verificar a normalidade dos dados.

```
pander(shapiro.test(res),
  style = "rmarkdown",
  caption = "Teste de normalidade Shapiro-Wilk para os resíduos"
)
```

Table 1: Teste de normalidade Shapiro-Wilk para os resíduos

Test statistic	P value
0.9334	0.2754

O teste acima confirma nossa suposição de que os resíduos têm distribuição Normal, pois, para um nível de significância de 97%, o valor-p obtido, 0,2754, não rejeita a hipótese nula, de normalidade dos dados.

Além disso, para determinar matematicamente se existe uma relação linear entre a variável resposta $\boldsymbol{Y}$ e qualquer as outras covariáveis $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k$, é possível utilizar o teste **ANOVA**. Nele, queremos testar:

H_0: Nenhuma das variáveis contribui significativamente ao modelo, versus:

H_a: Pelo menos uma das covariáveis contribui significativamente ao modelo.

Tabela *ANOVA*:

Table 2: Tabela ANOVA

F.V.	G.L	S.Q.	Q.M.	F
Regressão	k	SQ_{reg}	$QMreg = \frac{SQreg}{k}$	$F = \frac{QMreg}{QMres}$
Resíduo	$n-p$	SQ_{res}	$QMres = \frac{SQres}{n-p}$	
Total	$n-1$	SQ_{total}	QM_{Total}	

```
# Soma dos quadrados dos residuos
(SQRes <- t(Y - Y_est) %*% (Y - Y_est))
```

```
##          [,1]
## [1,] 650.5
```

$$SQ_{Reg} = SQ_{Total} - SQ_{Res} = \frac{1}{n}\sum_{i=1}^{n} y_i^2 - (Y - \hat{Y})^T \cdot (Y - \hat{Y}) =$$

$$\beta^T \cdot X^T \cdot Y - \frac{1}{n}(u^T \cdot Y)^2$$

```
# Soma dos quadrados totais
u <- c(rep(1, n))
(SQTotal <- t(Y) %*% Y - ((t(u) %*% Y)^2) / n)
```

```
##          [,1]
## [1,] 10363
```

```
# Soma dos quadrados da regressao
(SQReg <- SQTotal - SQRes)
```

```
##           [,1]
## [1,] 9712.5
```

```
# Calculando a anova
k <- 2 # covariaveis utilizadas
p <- k + 1


gl_sqreg <- k
QMReg <- SQReg / gl_sqreg

gl_sqres <- n - p
QMRes <- SQRes / gl_sqres

gl_sqtotal <- n - 1
```

```
# calculando a estatistica F
(F_0 <- QMReg / QMRes)
```

```
##          [,1]
## [1,] 97.05035
```

```
(QMTotal <- QMRes + QMReg)
```

```
##          [,1]
## [1,] 4906.288
```

Apresentamos os resultados obtidos na forma de tabela.

Table 3: Tabela ANOVA para o modelo ajustado.

F.V.	G.L	S.Q.	Q.M.	F
Regressão	2	9712.5	4856.25	97.05035
Resíduo	13	650.5	50.03846	
Total	15	10363	4906.288	

Como nosso estimador $F \sim F(k,\ n-k-1)$, podemos obter os quantis com o auxílio do R:

```
alpha <- 0.03
(RR <- qf(alpha, df1 = k, df2 = n - k - 1, lower.tail = F))
```

```
## [1] 4.648139
```

Rejeitamos H_0 se $F_0 > F_{crit}$, sendo F_{crit} o quantil teorico da distribuicao F com k e $n-p$ graus de liberdade.

```
if (RR < F_0) {
  cat("Rejeita-se H0")
}
```

```
## Rejeita-se H0
```

Dessa forma, podemos concluir com 97% de confiança que pelo menos uma das variáveis contribui significativamente ao modelo.

5. R^2 e $R^2 ajustado$

Agora que sabemos que nosso modelo de regressão linear múltipla é adequado, estamos interessados em medir quais covariáveis têm maior contribuição em explicar a variabilidade de nossa amostra.

Uma das formas de fazer isso, é utilizando o *coeficiente de determinação*, também conhecido como R^2, definido como:

$$R^2 = \frac{SQ_{Reg}}{SQ_{Total}}$$

R^2 é útil para quantificar o percentual da variabilidade dos dados explicado pelo modelo, contudo, como ele tende a aumentar conforme se adicionam mais covariáveis, sendo elas explicativas ou não. Uma forma de contornar esse impasse é através da utilização do R^2 ajustado:

$$R^2_{adj} = 1 - \frac{QM_{Res}}{QM_{Total}}$$

Para evitar duplicar o código utilizado para ajuste dos modelos, criamos uma função com este intuito:

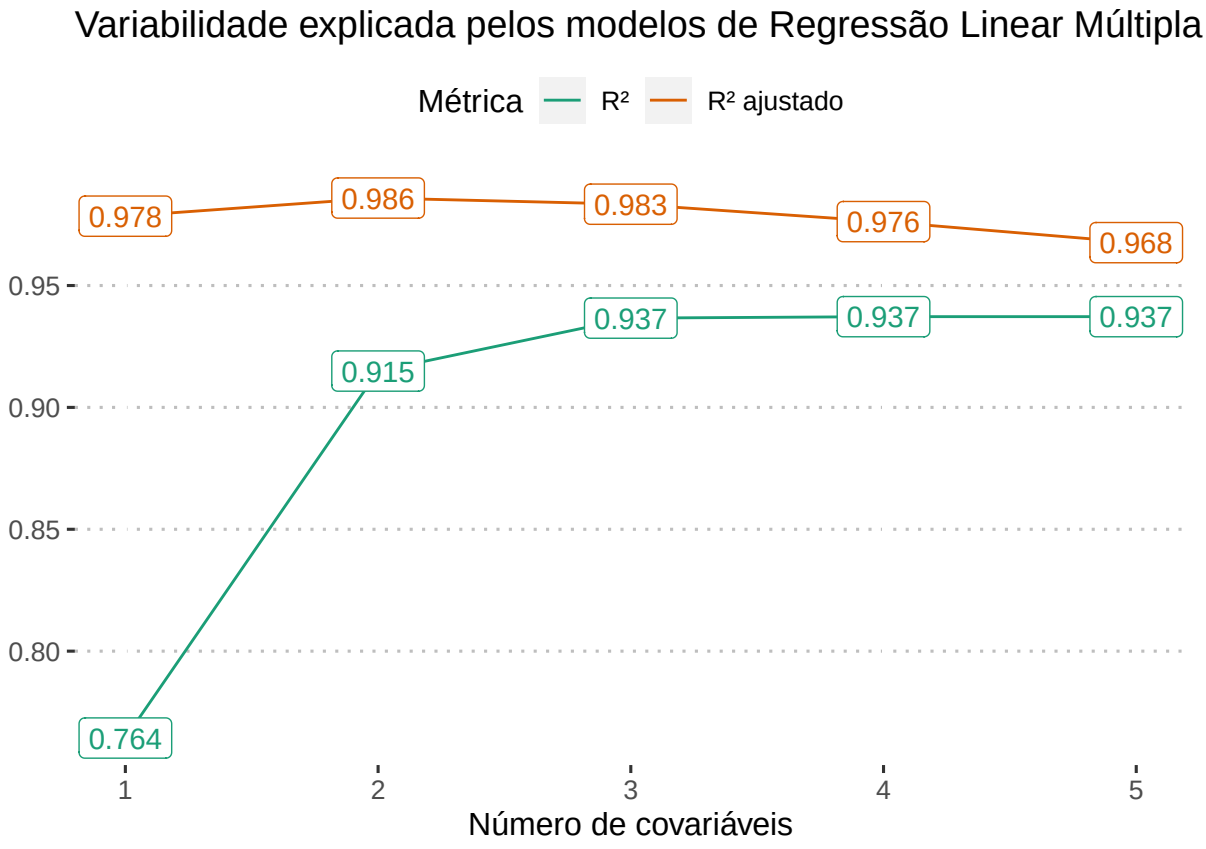
```
lm.anova <- function(x, y) {
  x <- as.matrix(x)
  y <- as.matrix(y)

  n <- length(y)
  k <- ncol(x) # covariaveis utilizadas
  p <- k + 1

  X <- matrix(c(rep(1, n), x), ncol = p, nrow = n, byrow = FALSE)
  Y <- matrix(y, ncol = 1, nrow = n)

  betas <- solve(t(X) %*% X) %*% t(X) %*% Y

  Y_est <- X %*% betas

  res <- Y - Y_est

  SQRes <- t(Y - Y_est) %*% (Y - Y_est)

  u <- c(rep(1, n))
  SQTotal <- t(Y) %*% Y - ((t(u) %*% Y)^2) / n

  SQReg <- SQTotal - SQRes

  gl_sqreg <- k
  QMReg <- SQReg / gl_sqreg
  gl_sqres <- n - p
  QMRes <- SQRes / gl_sqres
  SQTotal <- t(Y) %*% Y - ((t(u) %*% Y)^2) / n
  gl_sqtotal <- n - 1

  # calculando a estatistica F
  F_0 <- QMReg / QMRes

  # calculo de R2
  R2 <- SQReg / SQTotal

  QMTotal <- QMRes + QMReg
  # R2 ajustado
  R2adj <- 1 - QMRes / QMTotal
  # retorna uma lista com objetos relevantes ao modelo
  list(
    R2 = R2, R2adj = R2adj, betas = betas, Y_est = Y_est,
    anov = list(SQreg = SQReg, SQRes = SQRes, QMreg = QMReg, F_0 = F_0)
  )
}
```

Vamos criar uma lista com os cinco atributos presentes em nosso conjunto de dados, e ajustaremos modelos incrementalmente com as seguintes variáveis:

```r
feat <- c("Tamanho", "Temp", "Pressao", "FluxoCO2", "Umidade")
stepmodel <- 1:5 %>% map(~ head(feat, .x))
```

```r
# Calcula R2 e R2adj para os cinco modelos
r2 <- stepmodel %>%
  map_dfr(~ lm.anova(
    x = dados %>%
      select(all_of(.x)),
    y = dados$y
  )[c("R2adj", "R2")]) %>%
  cbind(x = 1:5)
```

```r
# gráfico de R2 e R2adj
r2 %>%
  pivot_longer(cols = !x) %>%
  ggplot(aes(x = x, y = value, color = name, label = round(value, 3))) +
  geom_line() +
  labs(
    title = "Variabilidade explicada pelos modelos de Regressão Linear Múltipla",
    x = "Número de covariáveis",
    y = "",
    color = "Métrica"
  ) +
  scale_color_brewer(palette = "Dark2", labels = c("R²", "R² ajustado")) +
  geom_label(show.legend = FALSE) +
  theme_pubclean()
```



Diferentemente do esperado, o valor de R^2 não aumentou com a inclusão de mais variáveis, e o valor de R^2_{adj} apenas diminuiu após a inclusão da terceira variável. Temos aqui fortes indícios que nem todas as variáveis são significativas ao modelo.

O modelo que foi capaz de melhor explicar a fonte de variabilidade dos dados de acordo com o critério de maior R^2_{adj} foi o modelo com apenas duas covariáveis, que apresentou $R^2_{adj} = 0.986$.

6. Testes nos Coeficientes de Regressão Individualmente

Usamos o teste t-parcial para saber se realmente todas as covariáveis são diferentes de zero. Assim, testamos cada um dos β_i, medindo a contribuição de uma determinada covariável considerando que outras co-variáveis estão no modelo. Definido o teste como:

$H_0 : \beta_j = 0$ versus $H_1 : \beta_j \neq 0$

Como consequência, se H_0 for rejeitado, X_i é importante dado que as outras covariáveis estão no modelo também.

Vale lembrar, ainda, que esse é um teste marginal ou parcial, pois como dito anteriormente $\hat{\beta}_{i-1}$ depende de todas as outras covariáveis.

Sendo:

```
alpha <- 0.03
beta0_est <- betas[1, 1]
beta1_est <- betas[2, 1]
beta2_est <- betas[3, 1]
beta3_est <- betas[4, 1]
beta4_est <- betas[5, 1]
beta5_est <- betas[6, 1]
```

```
t1 <- qt(alpha / 2, n - p)
t2 <- qt(1 - alpha / 2, n - p)
cbind(t1, t2)
```

```
##             t1       t2
## [1,] -2.435845 2.435845
```

Rejeitamos H_0 se $t < t_{(\alpha/2;n-p)}$ ou $t > t_{(1-\alpha/2;n-p)}$.

```
t_b1 <- beta1_est / (sqrt(QMRes * solve(t(X) %*% X)[2, 2]))
t_b1
```

Testando para β_1

```
##          [,1]
## [1,] 2.120505
```

```r
if (t_b1 < t1 || t_b1 > t2) {
  cat("Rejeita-se H0")
}
```

```r
t_b2 <- beta2_est / (sqrt(QMRes * solve(t(X) %*% X)[3, 3]))
t_b2
```

Testando para β_2

```
##          [,1]
## [1,] 5.583996
```

```r
if (t_b2 < t1 || t_b2 > t2) {
  cat("Rejeita-se H0")
}
```

```
## Rejeita-se H0
```

```r
t_b3 <- beta3_est / (sqrt(QMRes * solve(t(X) %*% X)[4, 4]))
t_b3
```

Testando para β_3

```
##           [,1]
## [1,] 0.3534175
```

```r
if (t_b3 < t1 || t_b3 > t2) {
  cat("Rejeita-se H0")
}
```

```r
t_b4 <- beta4_est / (sqrt(QMRes * solve(t(X) %*% X)[5, 5]))
t_b4
```

Testando para β_4

```
##               [,1]
## [1,] -8.789138e-15
```

```r
if (t_b4 < t1 || t_b4 > t2) {
  cat("Rejeita-se H0")
}
```

```r
t_b5 <- beta5_est / (sqrt(QMRes * solve(t(X) %*% X)[6, 6]))
t_b5
```

Testando para β_5

```
##           [,1]
## [1,] -12.58166
```

```r
if (t_b5 < t1 || t_b5 > t2) {
  cat("Rejeita-se H0")
}
```

```
## Rejeita-se H0
```

Portanto, seguindo a regra de decisão apresentada, concluímos que somente β_2 e β_5 rejeitam H_0. Logo, Temperatura e Tamanho são covariáveis significativas para o modelo, dado que já temos pelo menos uma variável significativa no modelo conforme constatado no teste *ANOVA* anteriormente.

7. Teste um Subconjunto de Coeficientes

Utilizando o critério dos testes t-parciais e R^2_{adj} calculados anteriormente, selecionamos o melhor modelo como sendo:

$$Y = \beta_0 + \beta_2 x_2 + \beta_5 x_5$$

Estimando os Betas para o melhor modelo

```r
x <- dados %>% select(Temp, Tamanho)
y <- dados$y
modelo <- lm.anova(x, y)
modelo$betas
```

```
##             [,1]
## [1,]  80.1346055
## [2,]   0.2821429
## [3,] -16.0649819
```

Estimando σ^2 para o melhor modelo

```r
gl <- n - 3
modelo$anov$SQRes / gl
```

```
##          [,1]
## [1,] 67.82692
```

Realizando o teste de um subconjunto de coeficientes

```r
F_testeparcial <- modelo$anov$F_0
```

```r
alpha <- 0.03
gl_modredu <- 2
RR <- qf(alpha, df1 = gl_modredu, df2 = n - gl_modredu - 1, lower.tail = F)
RR
```

```
## [1] 4.648139
```

Seguindo a regra, se $F > F_{\text{Tabelado}}$ rejeitamos H_0.

```r
if (RR < F_testeparcial) {
  cat("Rejeita-se H0")
}
```

```
## Rejeita-se H0
```

Como foi rejeitada H_0 temos que o subconjunto de coeficientes testados (variáveis Temperatura e Tamanho) contribui significativamente para o modelo. Vale ressaltar que este teste é análogo ao teste feito anteriormente usando a estatística t.

Apresentamos agora a equação final do modelo reduzido:

$$Y = 80.134 + 0.282x_2 - 16.065x_5$$

Interpretação dos coeficientes:

- β_0: Quando todos os x_i são iguais a zero, o valor esperado de y é de 80,134.
- β_2: Em média, para cada aumento de 1 ponto na Temperatura, esperamos um aumento de 0,282 em y, com todo o resto mantido constante.
- β_5: Em média, a cada aumento de 1 ponto no Tamanho, é esperado um descréscimo de 16,065 unidades em y, com todo o resto mantido constante.

8. IC Para Coeficientes de Regressão

Dado que o melhor modelo que encontramos foi aquele com as variáveis, **Temperatura (Temp)** e **Tamanho**, estimaremos o intervalo de confiança para os coeficientes $\beta_j = \beta_2, \beta_5$ com um coeficiente de confiança de 97%

Definimos o Índice de confiança $\gamma = (100 - \alpha)\%$, também chamado de *coeficiente de confiança*

```r
# Definindo o alpha e o t crítico
alpha <- 0.03
# para o melhor modelo temos que k =2
k <- 2
p <- k + 1
t1 <- qt(alpha / 2, n - p)
t2 <- qt(1 - alpha / 2, n - p)
```

Para β_2 temos que:

$97\%; \hat{\beta}_2 = 0,2821; QM_{Res} = 65,05; (X^T X)^{-1}_{33} = 5,1020e^{-}5; T_{(0,975;22)} = 2,4358$

```r
# Testando para o beta2
dp_b2 <- sqrt(sigma2 * diag(C_jj)[3])
b2_lim_inf <- betas[3] - t2 * dp_b2
b2_lim_sup <- betas[3] - t1 * dp_b2
IC_b2 <- cbind(b2_lim_inf, b2_lim_sup)
```

Para β_5 temos que:

$97\%; \hat{\beta}_5 = -16,065; QM_{Res} = 65,05; (X^TX)^{-1}_{66} = 0,0326; T_{(0,975;22)} = 2,4358$

```r
# Testando para beta5
dp_b5 <- sqrt(sigma2 * diag(C_jj)[6])
b5_lim_inf <- betas[6] - t2 * dp_b5
b5_lim_sup <- betas[6] - t1 * dp_b5
IC_b5 <- cbind(b5_lim_inf, b5_lim_sup)
```

```r
# Estruturando em uma tabela
IC_tab <- rbind(IC_b2, IC_b5)
row.names(IC_tab) <- c("beta2", "beta5")
colnames(IC_tab) <- c("IC (0.015)", "IC (0.985)")
IC_tab %>% kable(caption = "Intervalos de Confiança para os coeficientes do modelo")
```

Table 4: Intervalos de Confiança para os coeficientes do modelo

	IC (0.015)	IC (0.985)
beta2	0.1418145	0.4224712
beta5	-19.6111848	-12.5187791

A tabela acima pode ser interpretada da seguinte maneira: Temos 97% de confiança de que o real valor do parâmetro β_j está contido nos intervalos acima. Outro interpretação seria, se realizássemos esse experimento diversas vezes, concluiríamos que em 97% das vezes o real valor de β_j estaria dentro do intervalo.

9. IC para E(Y)

Podemos também construir intervalos de confiança para pontos específicos do nosso conjunto de dados. Seja $\boldsymbol{x}_0 = [1, x_{01}, x_{02}, \ldots, x_{0k}]$, podemos obter uma estimativa de $\hat{y}$ a partir da seguinte equação:

$$\hat{y_0} = x_0^T \hat{\beta}$$

Em seguida, estimamos a variância, ou erro padrão, de $\hat{y_0}$ através da equação:

$$Var(\hat{y_0}) = \hat{\sigma}^2 x_0^T (X^TX)^{-1} x_0$$

Desse modo, construímos nosso intervalo de confiança para a média de y no ponto x_0, conforme a seguinte equação:

$$\hat{y_0} - t_{\alpha/2,n-p}\sqrt{\hat{\sigma}^2 x_0^T (X^TX)^{-1} x_0} \leq E(y|x_0) \leq \hat{y_0} + t_{\alpha/2,n-p}\sqrt{\hat{\sigma}^2 x_0^T (X^TX)^{-1} x_0}$$

A seguir, calculamos o Intervalo de Confiança (IC) para a média , dado um x_0 - ou seja, dado uma observação do nosso conjunto de dados. De início, iremos calcular para o modelo com todas as covariáveis e, em seguida,

para o modelo com as covariáveis mais significativas. A fim de comparar os diferentes modelos e seus IC , utilizaremos como métrica de avaliação o **Erro Quadrático Médio (MSE)** e o **Coeficiente de Determinação R^2 Ajustado**

$$MSE = \sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}$$

$$R^2_{aj} = 1 - \frac{SQ_{Res}/(n-p)}{SQ_T/(n-1)}$$

IC de E(Y), considerando todas as covariáveis no modelo

```
sigma2 <- SQRes / (n - p)
t3 <- qt(1 - alpha / 2, n - p)



mean_IC <- data.frame(low_b = numeric(0), upper_b = numeric(0))
y_pred <- data.frame(value = numeric(0))

for (row in 1:nrow(X)) {
  x0 <- X[row, ]
  y0 <- x0 %*% betas
  se <- sqrt(sigma2 %*% t(x0) %*% C_jj %*% x0)
  low_b <- y0 - t3 * se
  upper_b <- y0 + t3 * se
  tab <- cbind(low_b, upper_b)
  mean_IC <- rbind(mean_IC, tab)
  y_pred <- rbind(y_pred, y0)
}

colnames(mean_IC) <- c("Lower Bound", "Upper Bound")
len_mean_IC <- mean_IC$`Upper Bound` - mean_IC$`Lower Bound`
mean_IC <- cbind(mean_IC, len_mean_IC)
mean_IC <- cbind(mean_IC, y_pred)
mean_IC <- cbind(mean_IC, dados$y)

colnames(mean_IC) <- c("Lower Bound", "Upper Bound", "Interval Length", "Predicted Value", "Real Value")

error <- abs(mean_IC$`Predicted Value` - mean_IC$`Real Value`)
mean_IC <- cbind(mean_IC, error)

colnames(mean_IC) <- c("Lower Bound", "Upper Bound", "Interval Length", "Predicted Value", "Real Value"

mse <- mean(mean_IC$`Absolute Error`**2)

mean_IC %>% kable(caption = "**Estimação do IC de E(Y) para cada observação do conjunto de dados conside
```

Table 5: **Estimação do IC de E(Y) para cada observação do conjunto de dados considerando todas as covariáveis** (MSE = 40.65, R^2 = 0.90)

Lower Bound	Upper Bound	Interval Length	Predicted Value	Real Value	Absolute Error
51.698425	72.80158	21.10315	62.25	63	0.75
14.698425	35.80158	21.10315	25.25	21	4.25
26.948425	48.05158	21.10315	37.50	36	1.50
78.948425	100.05158	21.10315	89.50	99	9.50
8.448425	29.55158	21.10315	19.00	24	5.00
60.448425	81.55158	21.10315	71.00	66	5.00
72.698425	93.80158	21.10315	83.25	71	12.25
35.698425	56.80158	21.10315	46.25	54	7.75
7.198425	28.30158	21.10315	17.75	23	5.25
59.198425	80.30158	21.10315	69.75	74	4.25
71.448425	92.55158	21.10315	82.00	80	2.00
34.448425	55.55158	21.10315	45.00	33	12.00
52.948425	74.05158	21.10315	63.50	63	0.50
15.948425	37.05158	21.10315	26.50	21	5.50
28.198425	49.30158	21.10315	38.75	44	5.25
80.198425	101.30158	21.10315	90.75	96	5.25

O resultado da tabela acima nos diz que, dado qualquer observação do nosso conjunto dados, podemos afirmar com 97% de confiança que o verdadeiro valor de $y = rendimento\ de\ azeite\ por\ lote\ de\ amendoim|x_0$ está contido neste intervalo, considerando que o nosso modelo leva em conta todas as covariáveis x_i.

Entretanto, sabemos que o melhor modelo encontrado foi aquele com as covariáveis **Pressão, Temperatura** e **Tamanho**. Considerando apenas essas variáveis explicatórias, obteremos os seguintes intervalos de confiança:

```
# O melhor modelo teve os valores dos betas modificados
# Redefinindo a matrix X
X_2 <- X[, c(1, 3, 6)]
# Redefinindo a matrix C_jj
C_jj2 <- solve(t(X_2) %*% X_2)

betas2 <- solve(t(X_2) %*% X_2) %*% t(X_2) %*% Y
betas2
```

```
##             [,1]
## [1,]  80.1346055
## [2,]   0.2821429
## [3,] -16.0649819
```

```
mean_IC2 <- data.frame(low_b = numeric(0), upper_b = numeric(0))
y_pred2 <- data.frame(value = numeric(0))

# recalculando a estatística teste t
# agora nosso k =4
t4 <- qt(1 - alpha / 2, n - 3)
sigma <- SQRes / (n - 3)
```

```r
for (row in 1:nrow(X)) {
  x0 <- X[row, c(1, 3, 6)]
  y0 <- x0 %*% t(t(betas2))
  se <- sqrt(sigma2 %*% t(x0) %*% C_jj2 %*% x0)
  low_b <- y0 - t4 * se
  upper_b <- y0 + t4 * se
  tab <- cbind(low_b, upper_b)
  mean_IC2 <- rbind(mean_IC2, tab)
  y_pred2 <- rbind(y_pred2, y0)
}

colnames(mean_IC2) <- c("Lower Bound", "Upper Bound")
len_mean_IC2 <- mean_IC2$`Upper Bound` - mean_IC2$`Lower Bound`
mean_IC2 <- cbind(mean_IC2, len_mean_IC2)
mean_IC2 <- cbind(mean_IC2, y_pred2)
mean_IC2 <- cbind(mean_IC2, dados$y)

colnames(mean_IC2) <- c("Lower Bound", "Upper Bound", "Interval Length", "Predicted Value", "Real Value

error <- abs(mean_IC2$`Predicted Value` - mean_IC2$`Real Value`)
mean_IC2 <- cbind(mean_IC2, error)

colnames(mean_IC2) <- c("Lower Bound", "Upper Bound", "Interval Length", "Predicted Value", "Real Value

mse2 <- mean(mean_IC2$`Absolute Error`**2)

mean_IC2 %>% kable(caption = "**Estimação do IC para cada observação do conjunto de dados considerando
```

Table 6: **Estimação do IC para cada observação do conjunto de dados considerando apenas as covariáveis significativas** (MSE = 55.10, R^2 = 0.985)

Lower Bound	Upper Bound	Interval Length	Predicted Value	Real Value	Absolute Error
59.16391	74.08609	14.92218	66.625	63	3.625
14.66391	29.58609	14.92218	22.125	21	1.125
34.41391	49.33609	14.92218	41.875	36	5.875
78.91391	93.83609	14.92218	86.375	99	12.625
14.66391	29.58609	14.92218	22.125	24	1.875
59.16391	74.08609	14.92218	66.625	66	0.625
78.91391	93.83609	14.92218	86.375	71	15.375
34.41391	49.33609	14.92218	41.875	54	12.125
14.66391	29.58609	14.92218	22.125	23	0.875
59.16391	74.08609	14.92218	66.625	74	7.375
78.91391	93.83609	14.92218	86.375	80	6.375
34.41391	49.33609	14.92218	41.875	33	8.875
59.16391	74.08609	14.92218	66.625	63	3.625
14.66391	29.58609	14.92218	22.125	21	1.125
34.41391	49.33609	14.92218	41.875	44	2.125
78.91391	93.83609	14.92218	86.375	96	9.625

Do mesmo modo que a tabela anterior, podemos afirmar com 97% de confiança que o verdadeiro valor de

$y = rendimento\ de\ azeite\ por\ lote\ de\ amendoim|x_0$ está contido neste intervalo, dado que o nosso modelo possui as covariáveis mais significativas.

Vale notar que é sempre preferível ter intervalos de confiança menores, uma vez que eles nos dão uma melhor ideia da magnitude da variável dependente y. Porém, nosso melhor modelo apresentou intervalos de confiança muito maiores que o modelo completo. Isso ocorre pois o modelo completo apresenta uma **erro padrão**, tal que: $Var(\hat{y_0}) = \hat{\sigma}^2 x_0^T (X^T X)^{-1} x_0 \approx 3.063$ enquanto que o melhor modelo, com apenas 3 covariáveis, apresenta um **erro padrão**: $Var(\hat{y_0}) = \hat{\sigma}^2 x_0^T (X^T X)^{-1} x_0 \approx 4.33$. Essa diferença é responsável pelo aumento do comprimento do IC. Ainda sim, vale lembrar que o R^2_{aj} do melhor modelo é maior que o do modelo completo, o que indica que esse modelo é, de fato, melhor.

Esse fato ilustra um conceito interessante em modelos preditivos e aprendizado de máquina: **o dilema viés-variância**. Esse dilema estabelece que a diminuição da variância de um modelo sempre ocorre às custas do aumento do viés e, vice-versa.

Podemos definir **Viés** como a diferença entre o valor real e o predito. Para múltiplas observações, computamos a média desta diferença:

$$Vies = E(\hat{y_i} - y_i)$$

E a **Variância** como a média dos desvios quadrados dos valores preditos $\hat{y_i}$ em relação à média dos valores preditos $E(\hat{y_i})$:

$$Var(\hat{y_i}) = \frac{\sum_{i=1}^{n} (\hat{y_i} - E(\hat{y_i}))^2}{n}$$

Pode-se mostrar que o MSE pode ser decomposto de forma a gerar a seguinte equação:

$$MSE = Viés^2 + Variância + Ruído$$

Table 7: Tabela Viés X Variância

	Viés	Variância
Melhor modelo	0	58.78
Modelo completo	0	43.37

Percebemos na tabela acima que ambos os modelos apresentaram $Viés = 0$, o que provavelmente ocorreu devido ao pequeno núemro de observações, mas ainda sim indica que nosso modelo não está enviesado a um modelo em particular. Vale notar também que o **melhor modelo**, mesmo apresentando maior variância é capaz de explicar - a partir do R_{aj} - 98% da variabilidade, enquanto que o **modelo completo**, explicava cerca de 90%, apresentando uma variabilidade bem menor, fato esse que, dado esse conjunto de dados, seria um indicativo de *sobreajuste*.