

SME0816 - Planejamento de Experimentos I

Profa. Cibele Russo

(Referências: Montgomery (2012), Notas de aula de Roseli Leandro; Clarice Demétrio; Marinho Andrade)

- Quatro projetos de circuito de computador
- Variável resposta: número de ruídos

Entrada de datos

```
y<-c(19, 20, 19, 30, 8,  
      80, 61, 73, 56, 80,  
      47, 26, 25, 35, 50,  
      95, 46, 83, 78, 97)  
trat <- c(rep(1,5),rep(2,5),rep(3,5),rep(4,5))  
trat <- as.factor(trat)  
datos <- data.frame(trat,y)  
str(datos)
```

```
## 'data.frame':    20 obs. of  2 variables:  
## $ trat: Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 2  
## $ y   : num  19 20 19 30 8 80 61 73 56 80 ...
```

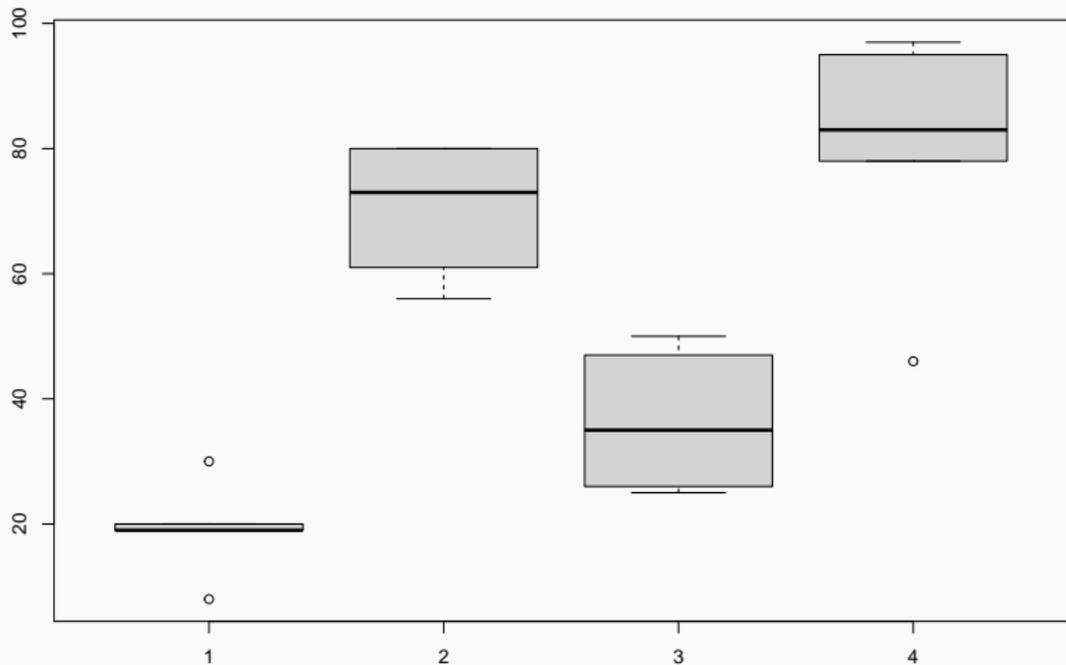
Dados

dados

##	trat	y
## 1	1	19
## 2	1	20
## 3	1	19
## 4	1	30
## 5	1	8
## 6	2	80
## 7	2	61
## 8	2	73
## 9	2	56
## 10	2	80
## 11	3	47
## 12	3	26
## 13	3	25

Boxplot

```
boxplot(split(dados$y,dados$trat))
```



Variabilidad

```
(varx <- tapply(dados$y,dados$trat,var))
```

```
##      1      2      3      4
```

```
## 60.7 121.5 134.3 420.7
```

```
max(varx)/min(varx)
```

```
## [1] 6.930807
```

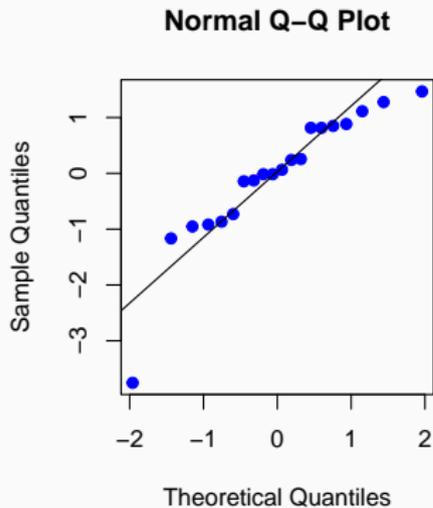
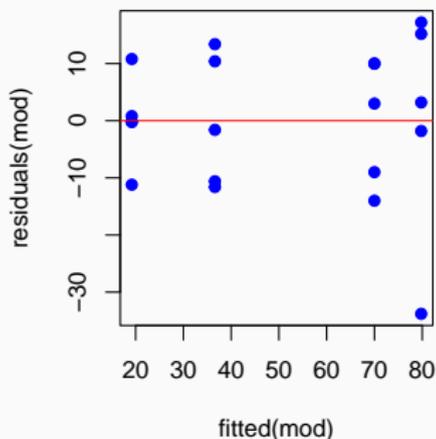
Análise dos resíduos

```
mod <- lm(y ~ trat,data=dados)
```

```
res <- rstudent(mod)
```

Análise gráfica

```
par(mfrow=c(1,2),pty="s")  
plot(fitted(mod),residuals(mod),pch=19,col="blue")  
abline(h=0,col="red")  
qqnorm(res,col="blue",pch=19); qqline(res)
```



Half-normal plot

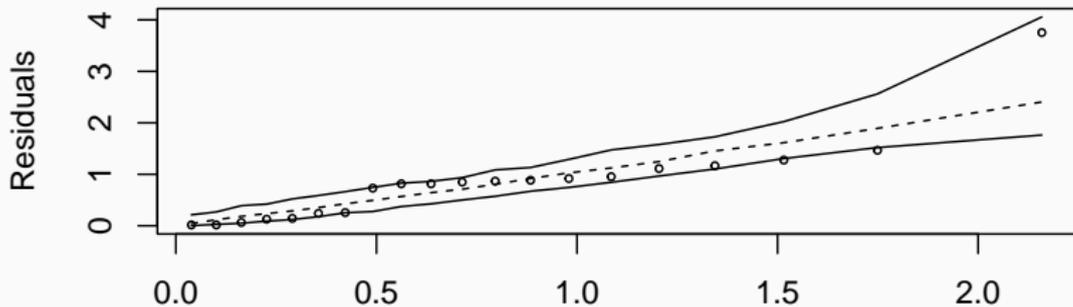
```
# half-normal plot with simulated envelope
```

```
library(hnp)
```

```
## Loading required package: MASS
```

```
hnp(mod)
```

```
## Gaussian model (lm object)
```



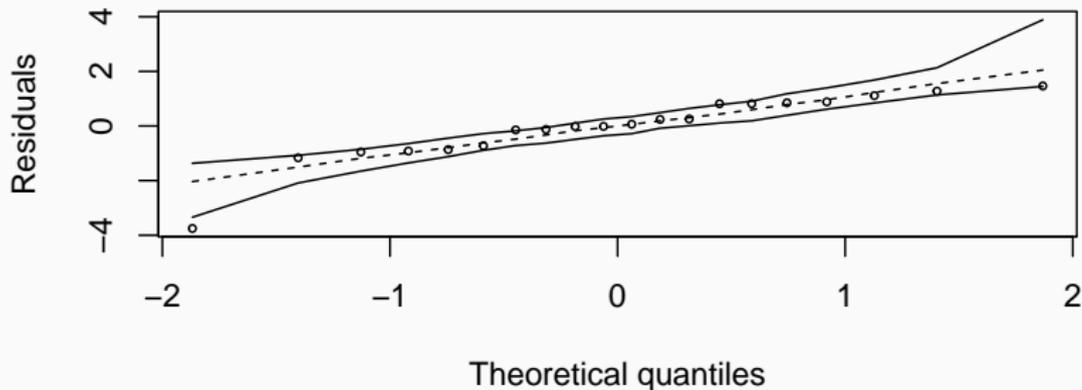
Normal plot

```
# normal plot with simulated envelope
```

```
library(hnp)
```

```
hnp(mod, half = F)
```

```
## Gaussian model (lm object)
```



Normalidade dos erros

```
shapiro.test(rstudent(mod))
```

```
##
```

```
## Shapiro-Wilk normality test
```

```
##
```

```
## data:  rstudent(mod)
```

```
## W = 0.8682, p-value = 0.01093
```

Homocedasticidade de variâncias

```
bartlett.test(y ~ trat, data=dados)
```

```
##
```

```
## Bartlett test of homogeneity of variances
```

```
##
```

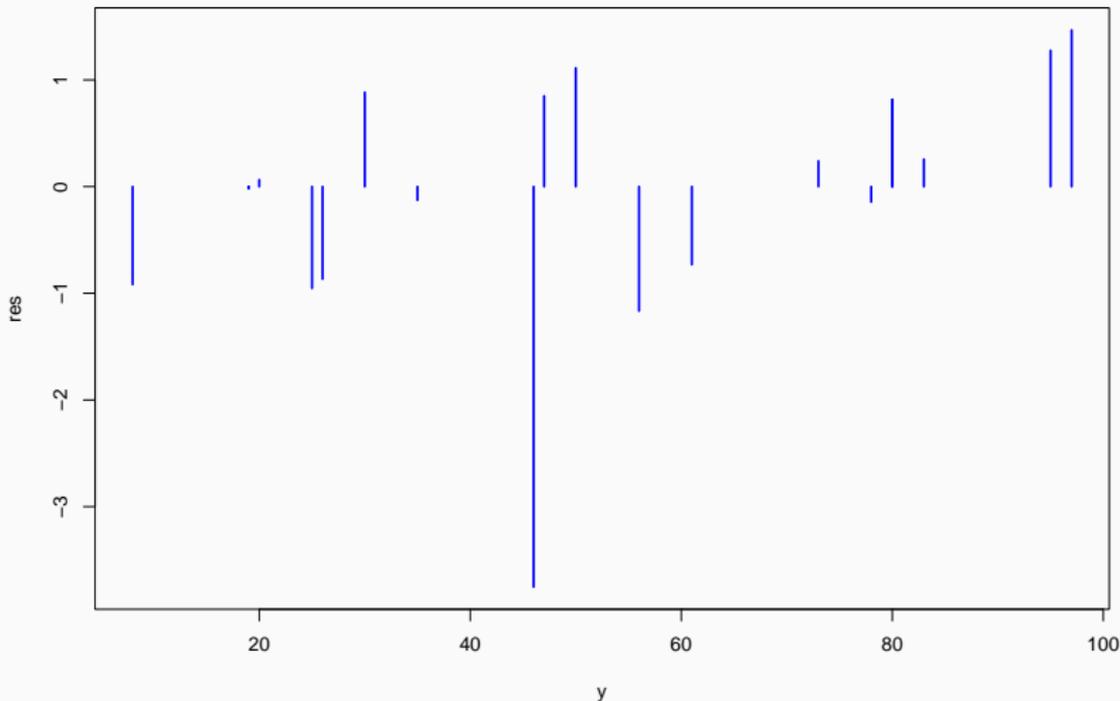
```
## data:  y by trat
```

```
## Bartlett's K-squared = 3.6893, df = 3, p-value = 0.297
```

- Os pressupostos do modelo não foram satisfeitos
- A hipótese de normalidade dos dados foi rejeitada
- Verificou-se a existência de um outlier.

Observe no tratamento 4 o resíduo da obs. 46 é -3.7530

```
plot(y, res, type='h', lwd=2, col="blue")
```



```
tapply(y, trat, summary)
```

```
## $`1`
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      8.0    19.0    19.0    19.2   20.0    30.0
```

```
##
```

```
## $`2`
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      56     61     73     70     80     80
```

```
##
```

```
## $`3`
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      25.0   26.0   35.0   36.6   47.0   50.0
```

```
##
```

```
## $`4`
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      46.0   78.0   83.0   79.8   95.0   97.0
```

Retirando-se a observação 46.

```
y.m <- c(19, 20, 19, 30, 8,  
         80, 61, 73, 56, 80,  
         47, 26, 25, 35, 50,  
         95, 83, 78, 97)
```

```
trat.m <- c(rep(1,5),rep(2,5),rep(3,5),rep(4,4))  
trat.m <- as.factor(trat.m)  
str(dados)
```

```
## 'data.frame':    20 obs. of  2 variables:  
## $ trat: Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 2  
## $ y   : num  19 20 19 30 8 80 61 73 56 80 ...
```

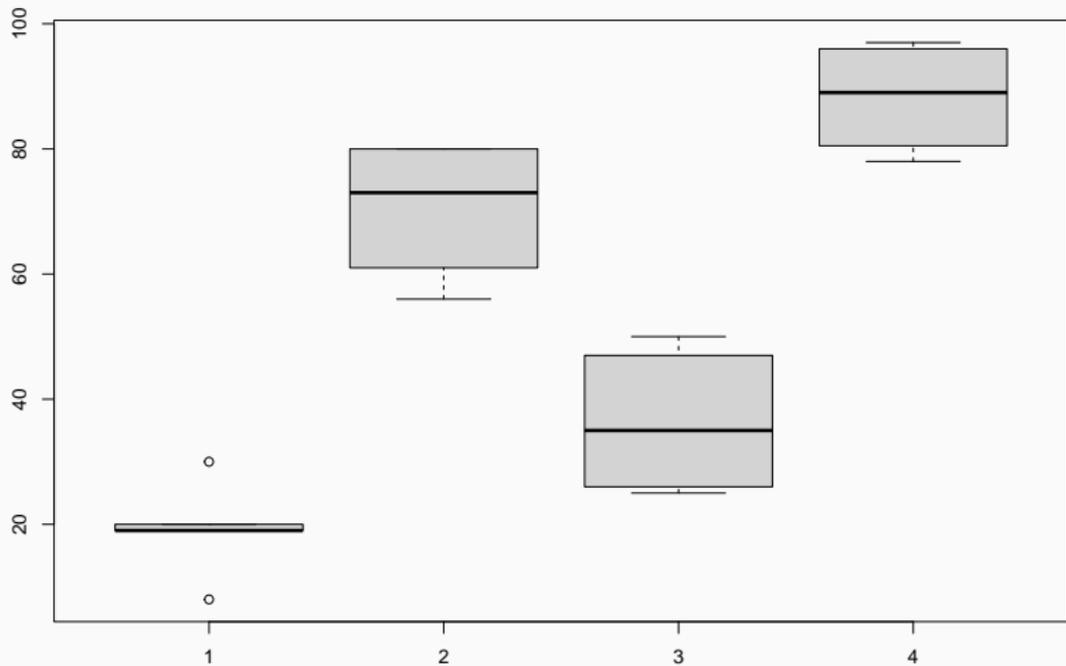
```
dados.novo <- data.frame(trat.m=trat.m,y.m=y.m)
```

Observe que o tratamento 4 passou a ter 4 repetições

```
t(dados.novo)
```

```
##           [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## trat.m "1"  "1"  "1"  "1"  "1"  "2"  "2"  "2"  "2"  "2"
## y.m    "19" "20" "19" "30" " 8" "80" "61" "73" "56" "80"
##           [,14] [,15] [,16] [,17] [,18] [,19]
## trat.m "3"    "3"    "4"    "4"    "4"    "4"
## y.m    "35"   "50"   "95"   "83"   "78"   "97"
```

```
boxplot(split(dados.novo$y.m,dados.novo$trat.m))
```



```
tapply(dados$y,dados$trat,var)
```

```
##      1      2      3      4  
## 60.7 121.5 134.3 420.7
```

```
(var.novo <- tapply(dados.novo$y.m,dados.novo$trat,var))
```

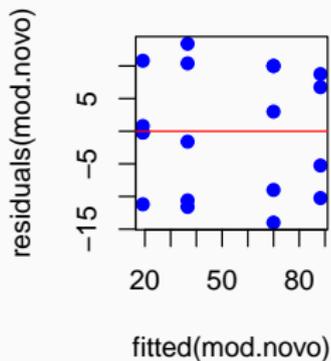
```
##           1           2           3           4  
## 60.70000 121.50000 134.30000 84.91667
```

```
max(var.novo)/min(var.novo)
```

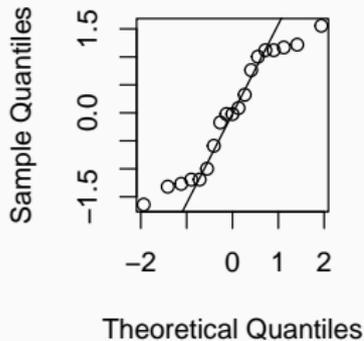
```
## [1] 2.212521
```

```
mod.novo <- lm(y.m ~ trat.m,data=dados.novo)
res.novo <- rstudent(mod.novo)
```

```
par(mfrow=c(1,2),pty="s")
plot(fitted(mod.novo),residuals(mod.novo),
     pch=19,col="blue")
abline(h=0,col="red")
qqnorm(res.novo)
qqline(res.novo)
```



Normal Q-Q Plot

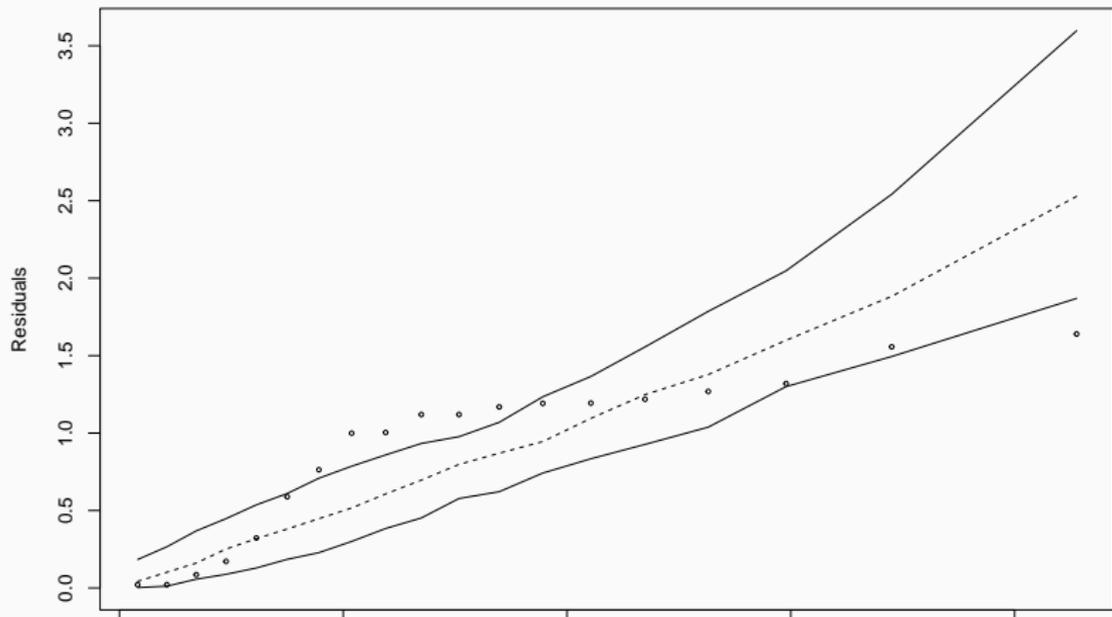


```
library(hnp)
```

```
# half-normal plot with simulated envelope
```

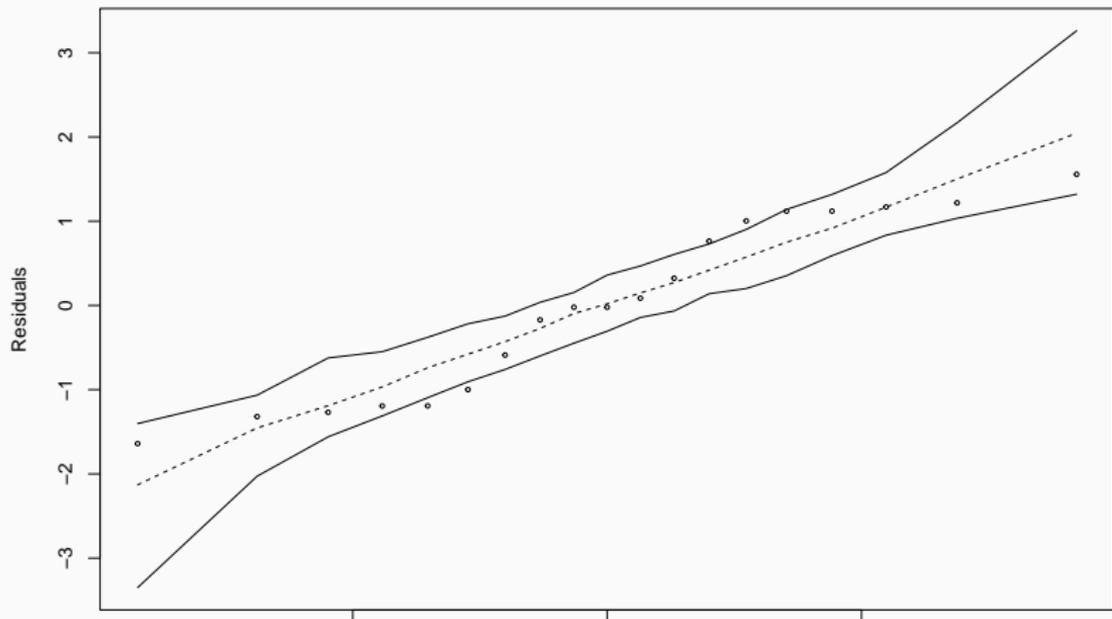
```
hnp(mod.novo)
```

```
## Gaussian model (lm object)
```



```
library(hnp)
# normal plot with simulated envelope
hnp(mod.novo, half = F)

## Gaussian model (lm object)
```



Normalidade

```
shapiro.test(rstudent(mod.novo))
```

```
##
```

```
## Shapiro-Wilk normality test
```

```
##
```

```
## data:  rstudent(mod.novo)
```

```
## W = 0.91525, p-value = 0.09245
```

Homogeneidade de variâncias

```
bartlett.test(y.m ~ trat.m, data=dados.novo)
```

```
##
```

```
## Bartlett test of homogeneity of variances
```

```
##
```

```
## data:  y.m by trat.m
```

```
## Bartlett's K-squared = 0.66029, df = 3, p-value = 0.8825
```

- As suposições básicas do modelo foram atendidas

Teste F

```
anova(mod.novo)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: y.m
```

```
##           Df  Sum Sq Mean Sq F value    Pr(>F)
```

```
## trat.m      3 13439.4  4479.8  44.187 1.106e-07 ***
```

```
## Residuals 15  1520.8   101.4
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Vamos interpretar o quadro da ANOVA

```
anova(mod)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: y
```

```
##           Df  Sum Sq Mean Sq F value    Pr(>F)
```

```
## trat       3 12042.0  4014.0    21.78 6.797e-06 ***
```

```
## Residuals 16  2948.8   184.3
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Verifica-se que existe efeito de tratamento, ou ainda, que existe diferenças estatisticamente significantes ao nível $\alpha=5\%$ em pelo menos um contraste de médias (valor $P = 1.106e-07 < 0,05$).

Objetivo do experimento

- Testar se existe efeito de tratamento.

Ou equivalentemente,

- Testar se existe diferença entre médias de tratamentos.

Modelo 1: Modelo de efeito de tratamentos

- Para testar se existe efeito de tratamento.

$$y_{ij} = \mu + \tau_i + \epsilon_{ij}$$

$$\begin{cases} H_0 : \tau_1 = \tau_2 = \dots = \tau_a = 0 \\ H_1 : \tau_i \neq 0 \text{ para algum } i \end{cases}$$

Modelo 2: Modelo de médias de tratamentos

$$\mu_i = \mu + \tau_i \implies \tau_i = \mu_i - \mu$$

- Para testar se existe diferença entre médias de tratamentos.

$$y_{ij} = \mu_i + \epsilon_{ij}$$

$$\left\{ \begin{array}{l} H_0 : \mu_1 = \mu_2 = \dots = \mu_a = \mu \\ H_1 : \mu_i \neq \mu_j \text{ para algum } i \neq j \end{array} \right.$$

Modelo de efeito de tratamentos

$$y_{ij} = \mu + \tau_i + \epsilon_{ij}$$

- y_{ij} é o valor observado no i -ésimo nível do tratamento, na repetição j , $i = 1, 2, \dots, a$, $j = 1, 2, \dots, k$.
- μ é uma constante inerente aos dados;
- τ_i é o efeito do tratamento i ;
- ϵ_{ij} é um erro aleatório que inclui todas as outras causas de variação do experimento.
- assume-se que $\epsilon_{ij} \sim N(0, \sigma^2)$, independentes ou seja, $E(Y_{ij}) = \mu + \tau_i$.

$$y_{ij} = \mu_i + \epsilon_{ij},$$

- y_{ij} é o valor observado no i -ésimo nível do tratamento, na repetição j , $i = 1, 2, \dots, a$, $j = 1, 2, \dots, k$.
- μ_i é a média do i -ésimo nível do tratamento, que poderá ser a média geral, ou a média de um dos tratamentos. Depende da restrição utilizada.
- ϵ_{ij} é um erro aleatório que inclui todas as outras causas de variação do experimento.
- assume-se que $\epsilon_{ij} \sim N(0, \sigma^2)$, independentes ou seja, $E(Y_{ij}) = \mu_i$.

Como instalar o experimento

Croqui na agricultura

- quatro tratamentos, $a = 4$, com $k = 5$ repetições

```
trat <- c("A","B","C","D")
```

```
rep(trat,each=4)
```

```
## [1] "A" "A" "A" "A" "B" "B" "B" "B" "C" "C" "C" "C" "D"
```

```
set.seed(13); x <- sample(rep(trat,each=5))
```

Croqui

```
croqui <- data.frame(Parcela = 1:20, trat.parcela = x )  
croqui
```

```
##   Parcela trat.parcela  
## 1      1           A  
## 2      2           A  
## 3      3           B  
## 4      4           C  
## 5      5           B  
## 6      6           D  
## 7      7           A  
## 8      8           D  
## 9      9           C  
## 10    10          B  
## 11    11          D  
## 12    12          A  
## 13    13          C  
## 14    14          A  
## 15    15          D  
## 16    16          C  
## 17    17          B  
## 18    18          B  
## 19    19          D  
## 20    20          C
```

O croqui de campo (Código)

```
require(dae)
k <- 5 # Número de repetições
a <- 4 # Número de tratamentos
n <- a*k # Número de unidades observacionais
(DIC.unid.obs <- list(Plot = n))
(tratamento <- factor(rep(c(1:a), each=k),
labels=c("A", "B", "C", "D")))
DIC.croqui <- designRandomize(recipient = DIC.unid.obs,
allocated = tratamento, seed=300)
DIC.croqui # dê as duas últimas colunas para o pesquisador

DIC.layout <- cbind(fac.gen(list(rows = 4,
columns = 5)),DIC.croqui)

designGGPlot(DIC.layout,
labels = "tratamento",
label.size = 5,
row.factors = "rows",
column.factors = "columns",
blockdefinition = cbind(4,5),
title = "",
xlab = "",
ylab = "",
colour= c("lightblue","lightgreen",
"lightgrey",
"lightpink"))
```

```
## Loading required package: dae

## Loading required package: ggplot2

## $Plot
## [1] 20

## [1] A A A A A B B B B B C C C C C D D D D D
## Levels: A B C D

##      Plot tratamento
## 1      1            B
## 2      2            C
## 3      3            B
## 4      4            B
## 5      5            D
## 6      6            C
## 7      7            D
## 8      8            C
## 9      9            D
## 10     10           A
## 11     11           A
## 12     12           B
## 13     13           A
## 14     14           C
## 15     15           A
## 16     16           D
## 17     17           B
## 18     18           A
## 19     19           C
## 20     20           D
```

Croqui

	1	2	3	4	5
1	B	C	B	B	D
2	C	D	C	D	A
3	A	B	A	C	A
4	D	B	A	C	D

Como instalar o experimento

Sequência na indústria

```
a <- 4      # número de tratamentos
k <- 5      # número de repetições
n <- a*k    # número de unidades observacionais
niveis.potencia <- as.factor(c(160,180,200,220))
potencia <- rep(niveis.potencia,each=5)
set.seed(13); x <- sample(1:n)
esquema <- data.frame(sequencia = 1:n,
                      potencia = potencia,
                      seq.inst=x, pot.inst=potencia[x])
```

Esquema

esquema

##	sequencia	potencia	seq.inst	pot.inst
## 1	1	160	3	160
## 2	2	160	5	160
## 3	3	160	10	180
## 4	4	160	13	200
## 5	5	160	6	180
## 6	6	180	16	220
## 7	7	180	4	160
## 8	8	180	17	220
## 9	9	180	12	200
## 10	10	180	8	180
## 11	11	200	20	220
## 12	12	200	1	160
## 13	13	200	11	200
## 14	14	200	2	160
## 15	15	200	19	220
## 16	16	220	14	200
## 17	17	220	9	180
## 18	18	220	7	180
## 19	19	220	18	220
## 20	20	220	15	200

Modelo de efeito fixo

- Os tratamentos foram especificamente escolhidos pelo experimentador;
- Nesse caso, deseja-se testar hipóteses sobre a média dos tratamentos dos tratamentos escolhidos;
- As conclusões serão aplicadas somente para os níveis de fatores considerados no experimento;
- As conclusões não podem ser estendidas para tratamentos que não foram testados;

- Pode ser de interesse estimar $(\mu, \tau_1, \dots, \tau_a, \sigma^2)$, ou $(\mu_1, \dots, \mu_a, \sigma^2)$
- $E(Y_{ij}) = E(\mu + \tau_i + e_{ij}) = \mu + \tau_i = \mu_i$
- $Var(Y_{ij}) = Var(\mu + \tau_i + e_{ij}) = Var(e_{ij}) = \sigma^2$

Modelo de efeito aleatório ou modelo de componentes de variância

- Os a tratamentos podem ser uma amostra de uma população grande de tratamentos;
- Nesse caso as conclusões podem ser estendidas para todos os tratamentos da população, independentemente de eles terem sido incluídos ou não na análise;
- Nesse modelo τ_i é uma variável aleatória;
- Os testes de hipóteses são sobre a variabilidade dos τ_i ,
- As estimativas serão para a sua variabilidade;
- Ver Capítulo 13 - Montgomery.

Análise da variância do modelo de efeito fixo

Table 1: Apresentação típica de experimentos com um fator

Nível do tratamento	Observações				Total	Médias
1	y_{11}	y_{12}	\dots	y_{1k}	$y_{1.}$	$\bar{y}_{1.}$
2	y_{21}	y_{22}	\dots	y_{2k}	$y_{2.}$	$\bar{y}_{2.}$
\vdots	\vdots	\vdots	\dots	\vdots	\vdots	\vdots
a	y_{a1}	y_{a2}	\dots	y_{ak}	$y_{a.}$	$\bar{y}_{a.}$
					$y_{..}$	$\bar{y}_{..}$

Dados balanceados

- a é o número de tratamentos;
- n é o número de repetições por tratamento (em Montgomery $k = n$);
- Média de tratamentos $\overline{y_{i.}}$

$$y_{i.} = \sum_{j=1}^k y_{ij}, \quad \overline{y_{i.}} = \frac{y_{i.}}{k}, i = 1, \dots, k$$

- Média geral $\overline{y_{..}}$

$$\overline{y_{..}} = \sum_{i=1}^a \sum_{j=1}^k y_{ij}, \quad \overline{y} = \frac{y_{..}}{ak}$$

- n é o número de unidades experimentais ou parcelas, $n = ak$

Decomposição da soma de quadrados total

- A análise da variância ou da variação é um processo aritmético, que consiste em decompor a variação total em fontes ou causas de variação.
- A variação total ou variabilidade total, denotada por $SQT = SQ_{Total} = SS_T$, é dada pela soma de quadrados total corrigida:

$$SQT = \sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{..})^2$$

em que $y_{ij} - \bar{y}_{..}$ mede o desvio de uma observação em relação a média amostral geral, sem levar em conta a que tratamento pertence a observação.

Variância dos valores observados em relação à média (sem considerar os tratamentos)

$$\frac{SS_T}{n-1} = \frac{SS_T}{ak-1} = \frac{\sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{..})^2}{ak-1}$$

- $SS_T = SQT = SQT_{Total}$.

Na decomposição

$$SQT = \sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{..})^2 = \sum_{i=1}^a \sum_{j=1}^k (y_{ij} + \bar{y}_{i.} - \bar{y}_{i.} - \bar{y}_{..})^2,$$

reagrupando os termos temos

$$SQT = \sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{..})^2 = \sum_{i=1}^a \sum_{j=1}^k [(\bar{y}_{i.} - \bar{y}_{..}) + (y_{ij} - \bar{y}_{i.})]^2.$$

Considerando-se que

$$\sum_{j=1}^k (y_{ij} - \bar{y}_{i.}) = (y_{i.} - k \frac{\sum y_{i.}}{k}) = 0,$$

tem-se:

$$SQT = \sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{..})^2 = k \sum_{i=1}^a (\bar{y}_{i.} - \bar{y}_{..})^2 + \sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{i.})^2$$

$$SQT = k \underbrace{\sum_{i=1}^a (\bar{y}_{i.} - \bar{y}_{..})^2}_{SQTrat} + \underbrace{\sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{i.})^2}_{SQR}$$

Assim,

$$SQT = SQTrat + SQRes$$

Ou

$$SQRes = SQT - SQTrat$$

- $SQRes = SQR$ é a soma de quadrados do resíduo
- $SQTrat$ é a soma de quadrados de tratamentos

A soma de quadrados total corrigida obtida foi descomposta na soma de quadrados das diferenças entre as médias de tratamentos e a média amostra geral

$$SQT = k \underbrace{\sum_{i=1}^a (\bar{y}_i - \bar{y}_{..})^2}_{\text{Entre Trat}} + \underbrace{\sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_i)^2}_{\text{Dentro de trat}}$$

Tabela da ANOVA

Table 2: Tabela da análise de variância para um fator - Efeito Fixo

C. V.	gl	SQ	QM	F_{obs}
Entre Trat.	$a - 1$	SQTrat	$QMTrat = \frac{SQTrat}{a-1}$	$F_{obs} = \frac{QMTrat}{QMRes}$
Dentro Trat.	$n - a$	SQRes	$QMRes = \frac{SQRes}{n-a}$	
Total	$n - 1$	SST		

$$\text{Correção} = C = \frac{y_{..}^2}{n}$$

$$SQTotal = \sum_{i=1}^a \sum_{j=1}^k y_{ij}^2 - C$$

$$SQTrat = \sum_{i=1}^a y_{i.}^2 - C$$

$$SQRes = SQTotal - SQTrat$$

- Variância dos valores observados independentemente do tratamento.

$$\frac{SS_T}{n-1} = \frac{SS_T}{ak-1} = \frac{\sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \bar{y}_{..})^2}{ak-1}$$

- $SS_T = SQT = SQT_{Total}$.

Exemplo de aplicação (Montgomery)

EXAMPLE 3.1 The Plasma Etching Experiment

To illustrate the analysis of variance, return to the first example discussed in Section 3.1. Recall that the engineer is interested in determining if the RF power setting affects the etch rate, and she has run a completely randomized experiment with four levels of RF power and five replicates. For convenience, we repeat here the data from Table 3.1:

RF Power (W)	Observed Etch Rate ($\text{\AA}/\text{min}$)					Totals	Averages
	1	2	3	4	5	y_i	\bar{y}_i
160	575	542	530	539	570	2756	551.2
180	565	593	590	579	610	2937	587.4
200	600	651	610	637	629	3127	625.4
220	725	700	715	685	710	3535	707.0
						$y_{..} = 12,355$	$\bar{y}_{..} = 617.75$

$$\begin{aligned}SS_T &= \sum_{i=1}^4 \sum_{j=1}^5 y_{ij}^2 - \frac{y_{..}^2}{N} \\&= (575)^2 + (542)^2 + \cdots + (710)^2 - \frac{(12,355)^2}{20} \\&= 72,209.75\end{aligned}$$

$$\begin{aligned}SS_{\text{Treatments}} &= \frac{1}{n} \sum_{i=1}^4 y_i^2 - \frac{y_{..}^2}{N} \\&= \frac{1}{5} [(2756)^2 + \cdots + (3535)^2] - \frac{(12,355)^2}{20} \\&= 66,870.55\end{aligned}$$

We will use the analysis of variance to test $H_0: \mu_1 = \mu_2 = \mu_3 = \mu_4$ against the alternative H_1 : some means are different. The sums of squares required are computed using Equations 3.8, 3.9, and 3.10 as follows:

$$\begin{aligned}SS_E &= SS_T - SS_{\text{Treatments}} \\&= 72,209.75 - 66,870.55 = 5339.20\end{aligned}$$

Usually, these calculations would be performed on a computer, using a software package with the capability to analyze data from designed experiments.

The ANOVA is summarized in Table 3.4. Note that the RF power or between-treatment mean square (22,290.18) is many times larger than the within-treatment or error mean square (333.70). This indicates that it is unlikely that the treatment means are equal. More formally, we

Entrada dos dados

```
require(readxl)
```

```
## Loading required package: readxl
```

```
dados <- read_xlsx("plasma_pag_62_ed7.xlsx" ,sheet=1)
```

Dados

dados

```
## # A tibble: 20 x 2
##   Potencia Etch.rate
##   <dbl>     <dbl>
## 1     160     575
## 2     180     565
## 3     200     600
## 4     220     725
## 5     160     542
## 6     180     593
## 7     200     651
## 8     220     700
## 9     160     530
## 10    180     590
## 11    200     610
## 12    220     715
## 13    160     539
## 14    180     579
## 15    200     637
## 16    220     685
## 17    160     570
## 18    180     610
## 19    200     629
## 20    220     710
```

Sobre os dados

```
head(dados,2)
```

```
## # A tibble: 2 x 2
##   Potencia Etch.rate
##   <dbl>     <dbl>
## 1     160       575
## 2     180       565
```

```
tail(dados,2)
```

```
## # A tibble: 2 x 2
##   Potencia Etch.rate
##   <dbl>     <dbl>
## 1     200       629
## 2     220       710
```

Sobre os dados

```
names(dados)
```

```
## [1] "Potencia" "Etch.rate"
```

```
str(dados)
```

```
## tibble [20 x 2] (S3: tbl_df/tbl/data.frame)
```

```
## $ Potencia : num [1:20] 160 180 200 220 160 180 200 220
```

```
## $ Etch.rate: num [1:20] 575 565 600 725 542 593 651 700
```

O tratamento deve ser transformado em fator

```
dados$Potencia <- as.factor(dados$Potencia)
str(dados)
```

```
## tibble [20 x 2] (S3: tbl_df/tbl/data.frame)
##  $ Potencia : Factor w/ 4 levels "160","180","200",...: 1
##  $ Etch.rate: num [1:20] 575 565 600 725 542 593 651 700
```

- Tratamento: Potência
- Níveis do tratamento: são os diferentes níveis da potência

```
levels(dados$Potencia)
```

```
## [1] "160" "180" "200" "220"
```

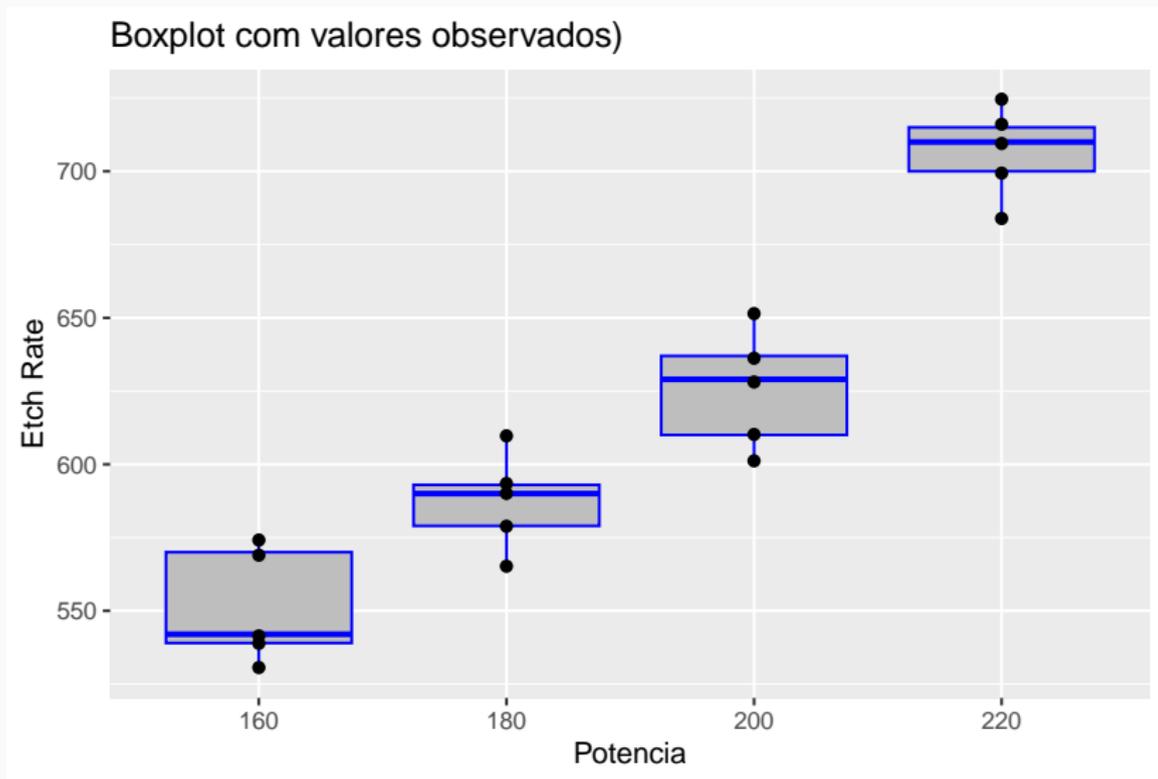
```
d1 <- tapply(dados$Etch.rate,dados$Potencia,summary)
d2 <- tapply(dados$Etch.rate,dados$Potencia,length)
d3 <- tapply(dados$Etch.rate,dados$Potencia,sum)
d4 <- tapply(dados$Etch.rate,dados$Potencia,mean)
d5 <- tapply(dados$Etch.rate,dados$Potencia,var)
d6 <- tapply(dados$Etch.rate,dados$Potencia,sd)
```

Tamanho amostral, média, variância, desvio padrão por tratamento,

```
data.frame(n=d2,soma=d3,media=d4,  
           variancia=d5,desvio.pad=round(d6,1))
```

```
##      n soma media variancia desvio.pad  
## 160 5 2756 551.2      400.7      20.0  
## 180 5 2937 587.4      280.3      16.7  
## 200 5 3127 625.4      421.3      20.5  
## 220 5 3535 707.0      232.5      15.2
```

Boxplot com valores observados



ANOVA de acordo com as fórmulas apresentadas em Montgomery - Passo a passo

```
(media.trat <- d4)

##    160    180    200    220
## 551.2 587.4 625.4 707.0

y <- dados$Etch.rate
k<-n1 <- n2 <- n3 <- n4 <- n5 <- 5
(SQTrat <- k*sum((media.trat-mean(y))^2))

## [1] 66870.55

(SQTotal <- sum((y-mean(y))^2))

## [1] 72209.75
```

```
(SQRes <- SQTotal - SQTrat)
```

```
## [1] 5339.2
```

```
a <- 4 # a é o número de tratamentos
```

```
# Quadrado médio do resíduo
```

```
(QMTrat <- SQTrat/(a-1))
```

```
## [1] 22290.18
```

```
# Quadrado médio do resíduo (estimativa para sigma2)
```

```
(QMRes <- SQRes/(n-a))
```

```
## [1] 333.7
```

Cálculo do valor p

```
#F observado
```

```
(F_obs <- QMTrat/QMRes)
```

```
## [1] 66.79707
```

```
#Cálculo do valor P
```

```
(p.valor <- pf(F_obs,4-1,20-4,lower.tail=F))
```

```
## [1] 2.882866e-09
```

```
(p.valor <- round(pf(F_obs,4-1,20-4,lower.tail=F),4))
```

```
## [1] 0
```

Conclusão:

Neste momento é possível concluir que:

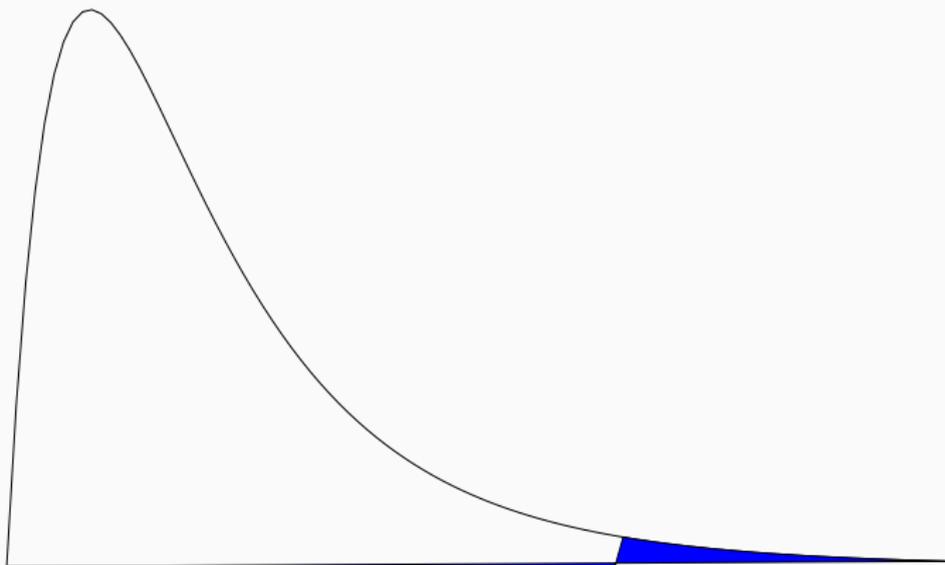
- Rejeita-se H_0 no nível de significância $\alpha = 5\%$
- Em média as taxas de gravuras diferem entre si.
- Ou ainda, pelo menos duas taxas de gravuras diferem entre si.

Resposta: Não. É necessário análise dos resíduos.

- <https://blog.minitab.com/pt/entendendo-analise-de-variacao-anova-e-o-teste-f>
- A probabilidade que queremos calcular é a probabilidade de observar uma estatística F que é pelo menos tão alta quanto o valor que nosso estudo obteve.
- Essa probabilidade nos permite determinar quão comum ou raro nosso valor F é, sob a suposição de que a hipótese nula é verdadeira.
- Se a probabilidade for baixa o suficiente, podemos concluir que nossos dados são inconsistentes com a hipótese nula. A evidência nos dados amostrais é forte o suficiente para rejeitar a hipótese nula para toda a população.
- Essa probabilidade que estamos calculando também é conhecida como valor-p!

Distribuição F

$$P(F < 3.239) = 0.05$$



Quantis da F

Análise utilizando a função aov do R

```
mod1 <- aov(dados$Etch.rate ~ dados$Potencia)

## Visualização do quadro da análise de variância
anova(mod1)
## Analysis of Variance Table
##
## Response: dados$Etch.rate
##           Df Sum Sq Mean Sq F value    Pr(>F)
## dados$Potencia  3  66871  22290.2  66.797 2.883e-09 ***
## Residuals     16   5339   333.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ou equivalentemente, utilizando a função lm do R

```
mod2 <- lm(dados$Etch.rate ~ dados$Potencia)

## Visualização do quadro da análise de variância
anova(mod2)
## Analysis of Variance Table
##
## Response: dados$Etch.rate
##           Df Sum Sq Mean Sq F value    Pr(>F)
## dados$Potencia  3  66871  22290.2  66.797 2.883e-09 ***
## Residuals     16   5339   333.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Final da análise?

Não, antes de concluir a análise deve-se fazer uma análise de resíduos.

- Segundo o modelo, deve-se verificar que:

$$e_{ij} \sim N(0, \sigma^2)$$

Com essa finalidade deve-se verificar as pressuposições básicas

- erros são normalmente distribuídos?
- erros são homocedásticos?
- erros são independentes?

$$e_{ij} = y_{ij} - \hat{y}_{ij}$$

Como obter \hat{y}_{ij} ???

Utilizando o modelo de efeito de tratamentos

$$y_{ij} = \mu + \tau_i + e_{ij}$$

Erro

$$e_{ij} = y_{ij} - \mu - \tau_i$$

y estimado: \hat{y}_{ij}

$$\hat{y}_{ij} = \hat{\mu} + \hat{\tau}_i$$

Erro estimado

$$\hat{e}_{ij} = y_{ij} - \hat{y}_{ij} = y_{ij} - (\hat{\mu} + \hat{\tau}_i)$$

Erro estimado

$$\hat{e}_{ij} = y_{ij} - \hat{y}_{ij} = y_{ij} - \hat{\mu} + \hat{\tau}_i$$

Para obtenção do erro deve-se estimar o parâmetro θ com dimensão $(a + 1) \times 1$

$$\theta = \begin{bmatrix} \mu \\ \tau_1 \\ \tau_2 \\ \vdots \\ \tau_a \end{bmatrix}$$

Estimação

O objetivo é encontrar $(\mu, \tau_1, \tau_2, \dots, \tau_a)^t$ que minimize a soma dos quadrados dos erros,

$$L(\mu, \tau_1, \tau_2, \dots, \tau_a) = L = \sum_{i=1}^a \sum_{j=1}^n (y_{ij} - \mu - \tau_i)^2$$

Ou seja, encontrar $(\mu, \tau_1, \tau_2, \dots, \tau_a)^t$ que satisfaça o sistema de $a + 1$ equações dado por:

$$\begin{cases} \frac{\partial L}{\partial \mu} = 0 \\ \frac{\partial L}{\partial \tau_i} = 0, \quad i = 1, \dots, a \end{cases}$$

ou equivalentemente,

$$\begin{cases} -2 \sum_{i=1}^a \sum_{j=1}^k (y_{ij} - \mu - \tau_i) = 0 \\ -2 \sum_{j=1}^k (y_{ij} - \mu - \tau_i) = 0, \quad i = 1, \dots, a \end{cases}$$

ou seja,

$$\begin{cases} n\mu + k\tau_1 + k\tau_2 + \dots + k\tau_a = y_{..} \\ n\mu + k\tau_1 + + = y_{1.} \\ n\mu + + k\tau_2 + = y_{2.} \\ + + + = \vdots \\ n\mu + + + + k\tau_a = y_{a.} \end{cases}$$

Modelo superparametrizado

As $a + 1$ equações são chamadas de equações normais de mínimos quadrados. Observe que a primeira linha pode ser obtida como soma das demais. Isso ocorre porque o modelo de efeito é um modelo é superparametrizado.

Com a restrição soma de efeitos de tratamentos igual a zero

$$\sum_{i=1}^a \tau_i = 0$$

tem-se:

$$\hat{\mu} = \overline{y_{..}}$$

$$\tau_i = \overline{y_{i.}} - \overline{y_{..}}, \quad i = 1, \dots, a$$

Solução não é única

A solução depende da restrição utilizada (será discutido posteriormente.)

```
mu    <- mean(dados$Etch.rate)
(med.trat <- tapply(dados$Etch.rate,dados$Potencia,mean))

##    160    180    200    220
## 551.2 587.4 625.4 707.0

tau1 <- med.trat[1] - mu
tau2 <- med.trat[2] - mu
tau3 <- med.trat[3] - mu
tau4 <- med.trat[4] - mu
theta.soma <- c(mu,tau1,tau2,tau3,tau4)
media.dos.trat <- c(mu+tau1,mu+tau2,mu+tau3,mu+tau4)
```

Solução com a restrição soma dos efeitos de tratamento igual a zero

```
data.frame(theta=theta.soma,  
           media=c(mu,media.trat))
```

```
##      theta  media  
##      617.75 617.75  
## 160 -66.55 551.20  
## 180 -30.35 587.40  
## 200  7.65 625.40  
## 220 89.25 707.00
```

Interpretação

- Observe que a estimativa para o parâmetro
 - μ é a média dos valores observados (a média geral)

```
mean(dados$Etch.rate)
```

```
## [1] 617.75
```

- O efeito do nível um do tratamento τ_1 é o quanto ele difere da média amostral.
- O efeito do nível dois do tratamento τ_2 é o quanto ele difere da média amostral.
- O efeito do nível três do tratamento τ_3 é o quanto ele difere da média amostral.
- O efeito do nível quatro do tratamento τ_4 é o quanto ele difere da média amostral.

O modelo de efeito de tratamentos para o DIC, matricialmente:

Considerando-se $a = 4$ e $k = 5$ portanto $n = 20$

$$\begin{bmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{15} \\ y_{21} \\ y_{22} \\ y_{23} \\ y_{24} \\ y_{25} \\ y_{31} \\ y_{32} \\ y_{33} \\ y_{34} \\ y_{35} \\ y_{41} \\ y_{42} \\ y_{43} \\ y_{44} \\ y_{45} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu \\ \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} + \begin{bmatrix} e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \\ e_{15} \\ e_{21} \\ e_{22} \\ e_{23} \\ e_{24} \\ e_{25} \\ e_{31} \\ e_{32} \\ e_{33} \\ e_{34} \\ e_{35} \\ e_{41} \\ e_{42} \\ e_{43} \\ e_{44} \\ e_{45} \end{bmatrix}$$

equivalentemente,

$$Y_{n \times 1} = X_{n \times (a+1)} \theta_{(a+1) \times 1} + e_{n \times 1}$$

a matriz X é chamada matriz do delineamento ou matriz de incidência.

A matriz de incidência, ou matriz do delineamento, no caso do exemplo (Etch.rate) (Por construção)

```
n <- 20
n.trat <- rep(5,4)
X.mu <- rep(1,length(n))
X1 <- c(rep(1,n.trat[1]),rep(0,n-n.trat[1]))
X2 <- c(rep(0,n.trat[1]),rep(1,n.trat[2]),
        rep(0,n.trat[3]),rep(0,n.trat[4]))
X3 <- c(rep(0,n.trat[1]),rep(0,n.trat[2]),
        rep(1,n.trat[3]),rep(0,n.trat[4]))
X4 <- c(rep(0,n.trat[1]),rep(0,n.trat[2]),
        rep(0,n.trat[3]),rep(1,n.trat[4]))
X <- cbind(X.mu,X1,X2,X3,X4)
```

```
dim(X)
```

```
## [1] 20 5
```

A matriz de incidência, ou matriz do delineamento, no caso do exemplo (Etch.rate) (Por construção)

X

```
##      X.mu X1 X2 X3 X4
## [1,]    1  1  0  0  0
## [2,]    1  1  0  0  0
## [3,]    1  1  0  0  0
## [4,]    1  1  0  0  0
## [5,]    1  1  0  0  0
## [6,]    1  0  1  0  0
## [7,]    1  0  1  0  0
## [8,]    1  0  1  0  0
## [9,]    1  0  1  0  0
## [10,]   1  0  1  0  0
## [11,]   1  0  0  1  0
## [12,]   1  0  0  1  0
## [13,]   1  0  0  1  0
## [14,]   1  0  0  1  0
## [15,]   1  0  0  1  0
## [16,]   1  0  0  0  1
## [17,]   1  0  0  0  1
## [18,]   1  0  0  0  1
## [19,]   1  0  0  0  1
## [20,]   1  0  0  0  1
```

$$Y = X\theta + e$$

$$e = Y - X\theta$$

$$\hat{Y} = X\hat{\theta}$$

$$\hat{e} = Y - X\hat{\theta}$$

$$\hat{e} = Y - \hat{Y}$$

O sistema de equações normais

$$\hat{Y} = X\hat{\theta}$$

$$X\hat{\theta} = \hat{Y}$$

$$X'X\hat{\theta} = X'\hat{Y}$$

- Se $X'X$ for inversível tem-se:

$$\hat{\theta} = (X'X)^{-1}X'\hat{Y}$$

- Ocorre que o sistema de equações normais não possui solução considerando-se X igual a matriz do delineamento, pois, $X'X$ não é inversível.

```
round(det(t(X)%*%X),1)
```

```
## [1] 0
```

Para resolução do sistema serão consideradas algumas restrições:

- Restrição soma dos efeitos de tratamento igual a zero (já considerado anteriormente).
- Restrição primeiro nível do tratamento igual a zero (restrição default do R).
- Restrição último nível do tratamento igual a zero (restrição default do SAS)
- Restrição de qualquer nível intermediário entre o primeiro e o último igual a zero.
- Para cada uma destas restrições tem-se uma matriz para o modelo, e assim, será possível encontrar uma estimativa para o erro, $\hat{\epsilon}$, e dessa maneira proceder à análise de resíduos.

Y estimado e o erro estimado no caso da restrição soma dos efeitos de tratamento igual a zero.

```
data.frame(X=X,yhat=X%*%theta.soma,y=dados$Etc.h.rate,erro=dados$Etc.h.rate-X%*%theta.soma)
```

##	X.X.mu	X.X1	X.X2	X.X3	X.X4	yhat	y	erro
## 1	1	1	0	0	0	551.2	575	23.8
## 2	1	1	0	0	0	551.2	565	13.8
## 3	1	1	0	0	0	551.2	600	48.8
## 4	1	1	0	0	0	551.2	725	173.8
## 5	1	1	0	0	0	551.2	542	-9.2
## 6	1	0	1	0	0	587.4	593	5.6
## 7	1	0	1	0	0	587.4	651	63.6
## 8	1	0	1	0	0	587.4	700	112.6
## 9	1	0	1	0	0	587.4	530	-57.4
## 10	1	0	1	0	0	587.4	590	2.6
## 11	1	0	0	1	0	625.4	610	-15.4
## 12	1	0	0	1	0	625.4	715	89.6
## 13	1	0	0	1	0	625.4	539	-86.4
## 14	1	0	0	1	0	625.4	579	-46.4
## 15	1	0	0	1	0	625.4	637	11.6
## 16	1	0	0	0	1	707.0	685	-22.0
## 17	1	0	0	0	1	707.0	570	-137.0
## 18	1	0	0	0	1	707.0	610	-97.0
## 19	1	0	0	0	1	707.0	629	-78.0
## 20	1	0	0	0	1	707.0	710	3.0

Ajuste do modelo com as funções do R

contr.sum contraste soma zero (Montgomery)

Especificamente, neste caso, o valor da constante (o intercepto) é igual a média dos dados observados

```
(mod_szero <- lm(Etch.rate ~ Potencia,  
                 data = dados,  
                 contrasts = list("Potencia" = contr.sum)))
```

```
##
```

```
## Call:
```

```
## lm(formula = Etch.rate ~ Potencia, data = dados, contrasts
```

```
##
```

```
## Coefficients:
```

```
## (Intercept)      Potencia1      Potencia2      Potencia3  
##      617.75         -66.55         -30.35           7.65
```

Comparando solução Montgomery (passo a passo) com a solução obtida no R

```
theta.soma
```

```
##           160      180      200      220  
## 617.75 -66.55 -30.35   7.65  89.25
```

```
dummy.coef(mod_szero)
```

```
## Full coefficients are
```

```
##
```

```
## (Intercept):      617.75
```

```
## Potencia:           160      180      200      220
```

```
##                -66.55 -30.35   7.65  89.25
```

```
mod <- lm(Etch.rate ~ Potencia,  
          data=dados,  
          contrasts = list("Potencia" = contr.sum))  
names(mod)
```

```
## [1] "coefficients" "residuals" "effects" "ra  
## [5] "fitted.values" "assign" "qr" "df  
## [9] "contrasts" "xlevels" "call" "te  
## [13] "model"
```

Y estimado e o erro estimado

```
data.frame(Y.obs=dados$Etch.rate,y.hat=X%*%theta.soma, e=dados$Etch.rate-X%*%theta.soma,  
           mod_szero$fit,mod_szero$res)
```

##	Y.obs	y.hat	e	mod_szero.fit	mod_szero.res
## 1	575	551.2	23.8	551.2	23.8
## 2	565	551.2	13.8	587.4	-22.4
## 3	600	551.2	48.8	625.4	-25.4
## 4	725	551.2	173.8	707.0	18.0
## 5	542	551.2	-9.2	551.2	-9.2
## 6	593	587.4	5.6	587.4	5.6
## 7	651	587.4	63.6	625.4	25.6
## 8	700	587.4	112.6	707.0	-7.0
## 9	530	587.4	-57.4	551.2	-21.2
## 10	590	587.4	2.6	587.4	2.6
## 11	610	625.4	-15.4	625.4	-15.4
## 12	715	625.4	89.6	707.0	8.0
## 13	539	625.4	-86.4	551.2	-12.2
## 14	579	625.4	-46.4	587.4	-8.4
## 15	637	625.4	11.6	625.4	11.6
## 16	685	707.0	-22.0	707.0	-22.0
## 17	570	707.0	-137.0	551.2	18.8
## 18	610	707.0	-97.0	587.4	22.6
## 19	629	707.0	-78.0	625.4	3.6
## 20	710	707.0	3.0	707.0	3.0

Tratamento tem que ser definido como fator

```
(nivel <- levels(dados$Potencia))
```

```
## [1] "160" "180" "200" "220"
```

Restrição: Primeiro nível igual a zero

```
dados$Potencia <- relevel(dados$Potencia,ref="160")
model.matrix(~dados$Potencia,ref="160")
```

```
##      (Intercept) dados$Potencia180 dados$Potencia200 dados$Potencia220
## 1             1             0             0             0
## 2             1             1             0             0
## 3             1             0             1             0
## 4             1             0             0             1
## 5             1             0             0             0
## 6             1             1             0             0
## 7             1             0             1             0
## 8             1             0             0             1
## 9             1             0             0             0
## 10            1             1             0             0
## 11            1             0             1             0
## 12            1             0             0             1
## 13            1             0             0             0
## 14            1             1             0             0
## 15            1             0             1             0
## 16            1             0             0             1
## 17            1             0             0             0
## 18            1             1             0             0
## 19            1             0             1             0
## 20            1             0             0             1
## attr("assign")
## [1] 0 1 1 1
## attr("contrasts")
## attr("contrasts")$`dados$Potencia`
## [1] "contr.treatment"
```

Restrição: Segundo nível igual a zero

```
dados$Potencia <- relevel(dados$Potencia,ref="180")
levels(dados$Potencia)
```

```
## [1] "180" "160" "200" "220"
```

```
model.matrix(~dados$Potencia,ref="180")
```

```
##      (Intercept) dados$Potencia160 dados$Potencia200 dados$Potencia220
## 1             1             1             0             0
## 2             1             0             0             0
## 3             1             0             1             0
## 4             1             0             0             1
## 5             1             1             0             0
## 6             1             0             0             0
## 7             1             0             1             0
## 8             1             0             0             1
## 9             1             1             0             0
## 10            1             0             0             0
## 11            1             0             1             0
## 12            1             0             0             1
## 13            1             1             0             0
## 14            1             0             0             0
## 15            1             0             1             0
## 16            1             0             0             1
## 17            1             1             0             0
## 18            1             0             0             0
## 19            1             0             1             0
## 20            1             0             0             1
## attr(,"assign")
## [1] 0 1 1 1
## attr(,"contrasts")
```

Restrição: Terceiro nível igual a zero

```
dados$Potencia <- relevel(dados$Potencia,ref="200")
levels(dados$Potencia)
```

```
## [1] "200" "180" "160" "220"
```

```
model.matrix(~dados$Potencia,ref="200")
```

```
##      (Intercept) dados$Potencia180 dados$Potencia160 dados$Potencia220
## 1             1             0             1             0
## 2             1             1             0             0
## 3             1             0             0             0
## 4             1             0             0             1
## 5             1             0             1             0
## 6             1             1             0             0
## 7             1             0             0             0
## 8             1             0             0             1
## 9             1             0             1             0
## 10            1             1             0             0
## 11            1             0             0             0
## 12            1             0             0             1
## 13            1             0             1             0
## 14            1             1             0             0
## 15            1             0             0             0
## 16            1             0             0             1
## 17            1             0             1             0
## 18            1             1             0             0
## 19            1             0             0             0
## 20            1             0             0             1
## attr(,"assign")
## [1] 0 1 1 1
## attr(,"contrasts")
```

Restrição: Quarto nível igual a zero

```
dados$Potencia <- relevel(dados$Potencia,ref="220")
levels(dados$Potencia)
```

```
## [1] "220" "200" "180" "160"
(X.4<-model.matrix(-dados$Potencia,ref="220"))
```

```
##      (Intercept) dados$Potencia200 dados$Potencia180 dados$Potencia160
## 1             1             0             0             1
## 2             1             0             1             0
## 3             1             1             0             0
## 4             1             0             0             0
## 5             1             0             0             1
## 6             1             0             1             0
## 7             1             1             0             0
## 8             1             0             0             0
## 9             1             0             0             1
## 10            1             0             1             0
## 11            1             1             0             0
## 12            1             0             0             0
## 13            1             0             0             1
## 14            1             0             1             0
## 15            1             1             0             0
## 16            1             0             0             0
## 17            1             0             0             1
## 18            1             0             1             0
## 19            1             1             0             0
## 20            1             0             0             0
## attr(,"assign")
## [1] 0 1 1 1
## attr(,"contrasts")
```

A matriz $X'X$ é não-singular

- Considerando-se a notação para a matriz do modelo: X

```
X <- X.4
```

```
mat <- t(X)%*%X
```

```
det(mat)
```

```
## [1] 625
```

```
head(X,4)
```

```
##      (Intercept) dados$Potencia200 dados$Potencia180 dados$Potencia160
```

```
## 1           1           0           0
```

```
## 2           1           0           1
```

```
## 3           1           1           0
```

```
## 4           1           0           0
```

Estimativa dos parâmetros com a restrição primeiro nível igual a zero

```
(est.par <- solve(mat)%*%t(X)%*%dados$Etch.rate)

##                [,1]
## (Intercept)      707.0
## dados$Potencia200 -81.6
## dados$Potencia180 -119.6
## dados$Potencia160 -155.8

# Média dos níveis de Potência
media <- tapply(dados$Etch.rate,dados$Potencia,mean)
```

Observe:

```
data.frame(est.par=est.par,  
           media=media,  
           d.med = media-media[1])
```

```
##                est.par media  d.med  
## (Intercept)         707.0 707.0    0.0  
## dados$Potencia200   -81.6 625.4  -81.6  
## dados$Potencia180 -119.6 587.4 -119.6  
## dados$Potencia160 -155.8 551.2 -155.8
```

```
mod1 <- aov(Etch.rate ~ Potencia, dados )  
coef(mod1)
```

```
## (Intercept) Potencia200 Potencia180 Potencia160  
##          707.0          -81.6          -119.6          -155.8
```

```
dummy.coef(mod1)
```

```
## Full coefficients are
```

```
##
```

```
## (Intercept):          707
```

```
## Potencia:           220    200    180    160
```

```
##                   0.0  -81.6 -119.6 -155.8
```

Exercício

- Considerando-se o modelo de médias de tratamentos
 - Construa a matriz do delineamento
 - Verifique se essa matriz é inversível
 - Estime o vetor de parâmetros e interprete.

O ajuste do modelo de médias no R

```
modelo2 <- lm(Etch.rate ~ Potencia-1,dados)
```

```
modelo1 <- lm(Etch.rate ~ Potencia,dados)
modelo2 <- lm(Etch.rate ~ Potencia-1,dados)
library(car)
compareCoefs(modelo1,modelo2)
```

Comparando

```
## Loading required package: carData

## Calls:
## 1: lm(formula = Etch.rate ~ Potencia, data = dados)
## 2: lm(formula = Etch.rate ~ Potencia - 1, data = dados)
##
##           Model 1 Model 2
## (Intercept) 707.00
## SE           8.17
##
## Potencia200 -81.60 625.40
## SE           11.55  8.17
##
## Potencia180 -119.60 587.40
## SE           11.55  8.17
##
## Potencia160 -155.80 551.20
## SE           11.55  8.17
##
## Potencia220          707.00
## SE                 8.17
##
```

As matrizes utilizando-se as diferentes parametrizações do modelo

Cuidado na interpretação!!!

```
levels(dados$Potencia)
```

```
## [1] "220" "200" "180" "160"
```

```
levels(dados$Potencia)[1] # Primeiro nível
```

```
## [1] "220"
```

```
levels(dados$Potencia)[4] # Segundo nível
```

```
## [1] "160"
```

```
dados$Potencia <- relevel(dados$Potencia,ref="160")
```

```
levels(dados$Potencia)
```

```
## [1] "160" "220" "200" "180"
```

contr.treatment anula o primeiro nível

```
mod_zero1 <- lm(Etch.rate ~ Potencia, data = dados,  
contrasts = list("Potencia" = contr.treatment))  
coef(mod_zero1)  
  
## (Intercept) Potencia220 Potencia200 Potencia180  
##          551.2          155.8           74.2           36.2
```

```
mod_zerot <- lm(Etch.rate ~ Potencia, data = dados,  
contrasts = list("Potencia" = contr.SAS))
```

Comparando os coeficientes dos modelos

```
coef(mod_zero1)
```

```
## (Intercept) Potencia220 Potencia200 Potencia180  
##          551.2          155.8           74.2           36.2
```

```
coef(mod_zerot)
```

```
## (Intercept) Potencia160 Potencia220 Potencia200  
##          587.4          -36.2          119.6           38.0
```

```
coef(mod_szero)
```

```
## (Intercept) Potencia1 Potencia2 Potencia3  
##          617.75      -66.55      -30.35           7.65
```

```
#coef(modelo2)
```

Estimativa de todos os parâmetros envolvidos no modelo

```
dummy.coef(mod_zero1)
```

```
## Full coefficients are
##
## (Intercept):    551.2
## Potencia:       160   220   200   180
##                0.0 155.8  74.2  36.2
```

```
dummy.coef(mod_zerot)
```

```
## Full coefficients are
##
## (Intercept):    587.4
## Potencia:       160   220   200   180
##                -36.2 119.6  38.0   0.0
```

```
dummy.coef(mod_szero)
```

```
## Full coefficients are
##
## (Intercept):    617.75
## Potencia:       160   180   200   220
##                -66.55 -30.35  7.65  89.25
```

```
dummy.coef(modelo2)
```

```
## Full coefficients are
##
## Potencia:       220   200   180   160
##                707.0 625.4 587.4 551.2
```

```
media
```

Estimativa para σ^2

- Prova-se que:

$$E\left(\frac{SQRes}{n-a}\right) = \sigma^2$$

em que

$$SQRes = (y - X.Mod \theta^*)$$

é a soma de quadrados do resíduo e a é o número de tratamentos.

- Assim,

$$\frac{SQRes}{n-a}$$

é um estimador não viciado para σ^2

Soma de quadrados com a restrição soma

```
a <- 4
X <- model.matrix(mod_zero1) # X é a matriz de incidência
theta.soma <- coef(mod_zero1)
(SQResiduo <- sum((y - X%%theta.soma)^2))

## [1] 5339.2

(hat.sigma2 <- SQResiduo/(n-a)) #QMRésíduo

## [1] 333.7

sigma(mod_zero1)

## [1] 18.26746

sigma(mod_zero1)^2

## [1] 333.7
```

Verificação dos pressupostos básicos do modelo

$$y_{ij} = \mu + \tau_i + e_{ij}, \quad e_{ij} \sim N(0, \sigma^2)$$

- 1) os erros são normalmente distribuídos,
 - 2) existe homogeneidade de variâncias,
 - 3) os erros são independentes.
- Para esta finalidade serão utilizados gráficos e testes estatísticos apropriados.

- Testes de normalidade
 - Técnica gráfica
 - Teste de Kolmogorov-Smirnov
 - Teste de Anderson-Darling
 - Teste de Shapiro-Wilk

Teste de Shapiro-Wilk

- https://en.wikipedia.org/wiki/Shapiro%E2%80%93Wilk_test
- O teste de Shapiro-Wilk é um teste de normalidade na estatística frequentista.
- Foi publicado em 1965 por Samuel Sanford Shapiro and Martin Wilk.
- Shapiro, S. S.; Wilk, M. B. (1965). “An analysis of variance test for normality (complete samples)”. *Biometrika*. 52 (3–4): 591–611.

- O teste Shapiro-Wilk é baseado na estatística W dada por:

$$W = \frac{b^2}{\sum_{i=1}^n (x_{(i)} - \bar{x})^2}$$

em que $x_{(i)}$ são os valores da amostra ordenados ($x_{(1)}$ é o mínimo).
A constante b é determinada da seguinte forma

b?

$$b = \begin{cases} \sum_{i=1}^{n/2} a_{n-i+1} \times (x_{(n-i+1)} - x_{(i)}) & \text{se } n \text{ é par} \\ \sum_{i=1}^{(n+1)/2} a_{n-i+1} \times (x_{(n-i+1)} - x_{(i)}) & \text{se } n \text{ é ímpar} \end{cases}$$

em que a_{n-i+1} são constantes geradas pelas médias, variâncias e covariâncias das estatísticas de ordem de uma amostra de tamanho n de uma distribuição Normal. Seus valores são tabelados.

Passos para realização do teste de Shapiro-Wilk:

- Passo 1. Formulação da Hipóteses:

$$\begin{cases} H_0 : & \text{A amostra vem de uma população Normal} \\ H_1 : & \text{A amostra não vem de uma população Normal} \end{cases}$$

- Definir a estatística do teste

$$W = \frac{b^2}{n \sum_{i=1} (x_{(i)} - \bar{x})^2}$$

em que

$$b = \begin{cases} \sum_{i=1}^{n/2} a_{n-i+1} \times (x_{(n-i+1)} - x_{(i)}) & \text{se } n \text{ é par} \\ \sum_{i=1}^{(n+1)/2} a_{n-i+1} \times (x_{(n-i+1)} - x_{(i)}) & \text{se } n \text{ é ímpar} \end{cases}$$

Os valores de W são tabelados.

3. Estabelecer o Nível de significância do teste (α), normalmente 0,05, e definir a região crítica do teste.

4. Calcular o valor observado da estatística do teste:

- Ordenar as n observações da amostra: $x_{(1)}, x_{(2)}, x_{(3)}, \dots, x_{(n)}$;
- Calcular $\sum_{i=1}^n (x_i - \bar{x})^2$;
- Calcular b ;
- Calcular W .

5. Tomar a decisão: Rejeitar H_0 ao nível de significância α se $W_{\text{calculado}} < W_{\alpha}$ (os valores críticos da estatística W de Shapiro-Wilk são tabelados).

Shapiro-Wilk é um teste mais poderoso

- Isto é, a probabilidade de rejeitar H_0 quando H_0 é falsa é mais alta quando comparada aos outros testes
 - Simulação Monte Carlo demonstrou que o teste de Shapiro-Wilk é mais poderoso para um dado nível de significância.
 - Seguido pelo teste Anderson-Darling quando é feita a comparação entre Shapiro-Wilk e os testes Kolmogorov-Smirnov, Lilliefors e Anderson-Darling.

<https://pt.slideshare.net/rosariocacao/testes-parametricos-e-nao-parametricos-3396639>

<http://leg.ufpr.br/~lucambio/Nonparam/Nparam.html>

<http://www.liaaq.ccb.ufsc.br/files/2013/10/Aula-4.pdf>

<https://www.ime.usp.br/~sandoval/mae5755/ANOVA-1fator.pdf>

<https://www.ime.usp.br/~sandoval/mae5755/naoparametrica.pdf>

Teste de Bartlett

- É um teste para homogeneidade de variâncias (Snedecor & Cochran, 1983)
- É usado para testar se k amostras tem variâncias iguais
- Variâncias iguais entre grupos é denominada homogeneidade de variâncias.
- Teste de Bartlett é sensível a falta de normalidade
- Um teste alternativo é o teste de Levene que é um teste menos sensível.

Passos do teste

1- Hipóteses a serem testadas

Para o modelo heterocedástico, vamos inicialmente testar as hipóteses

$$\begin{cases} H_0 : \text{Os erros são homocedásticos} \\ H_1 : \text{Os erros não são homocedásticos} \end{cases}$$

ou equivalentemente,

$$\begin{cases} H_0 : \sigma_1^2 = \sigma_2^2 = \dots = \sigma_a^2 = \sigma^2 \\ H_1 : \sigma_i^2 \neq \sigma_j^2, \text{ para } i \neq j, \quad i, j = 1, \dots, a \end{cases}$$

2. Definir a estatística do teste

A estatística do teste proposta por Bartlett é dada por

$$B_0 = \frac{q}{c}$$

em que

$$q = (N - k) \ln s_p^2 - \sum_{i=1}^k [(n_i - 1) \ln s_i^2]$$

e

$$c = 1 + \frac{1}{3(k-1)} \left(\sum_{i=1}^k \frac{1}{n_i - 1} - \frac{1}{N - k} \right)$$

$$s_p^2 = \frac{1}{N - k} \sum_{i=1}^k (n_i - 1) s_i^2$$

$$s_i^2 = \sum_{j=1}^{n_i} \frac{(y_{ij} - \bar{y}_{i.})^2}{n_i - 1}$$

- $T \approx \chi_{1-\alpha, k-1}^2$
3. Definir a região crítica do teste considerando-se o nível de significância α
 - Região crítica: As variâncias são consideradas desiguais se, $T > \chi_{1-\alpha, k-1}^2$ em que $\chi_{1-\alpha, k-1}^2$ é o valor crítico da distribuição qui-quadrado com $k - 1$ graus de liberdade e nível de significância α
 4. Calcular o valor observado da estatística.
 5. Concluir.

<http://www.sthda.com/english/wiki/compare-multiple-sample-variances-in-r>

Verificação dos pressupostos básicos

```
mod1 <- modelo1
anova(mod1)

## Analysis of Variance Table
##
## Response: Etch.rate
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Potencia   3  66871 22290.2  66.797 2.883e-09 ***
## Residuals 16   5339   333.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```



```
res <- rstudent(mod1)
```

Análise gráfica (Normalidade)

qqnorm, qqline

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

```
qqnorm(res,pch=19,col="blue")
```

```
qqline(res,col="blue")
```

```
hist(res,border="blue",col="lightgray")
```

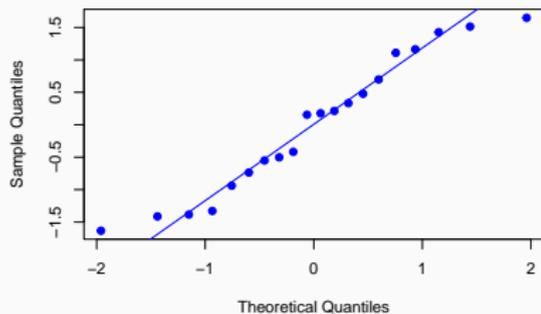
```
boxplot(res)
```

```
plot(mod1$fit,res,col="blue",pch=19)
```

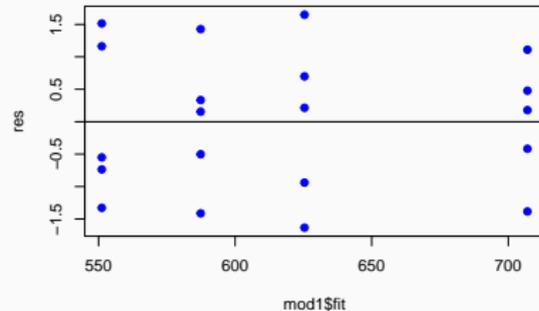
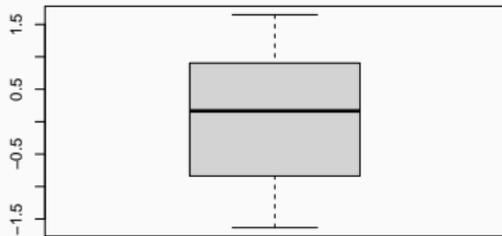
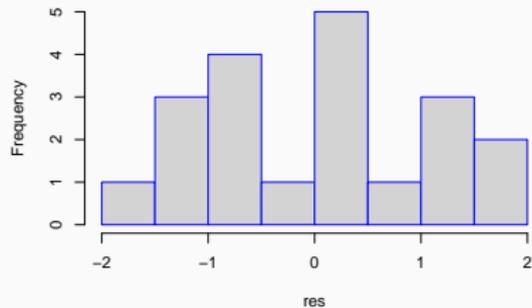
```
abline(h=0)
```

Gráfico de quantis: qqnorm, qqline

Normal Q-Q Plot



Histogram of res



- Aparentemente os erros são normalmente distribuídos.
- Não apresenta **outlier**

Teste de normalidade (shapiro.test)

```
shapiro.test(res)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data:  res  
## W = 0.94514, p-value = 0.2993
```

Não existem evidências para rejeitar a normalidade dos erros, ao nível de significância, $\alpha = 0,05$ ($p\text{-value} = 0,2993 > 0,05$)

Gráfica

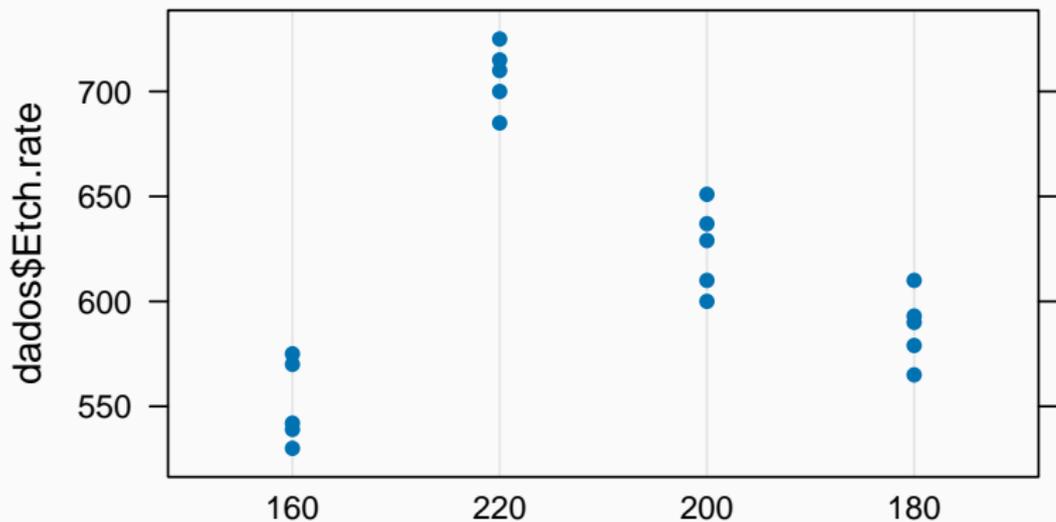
```
require(lattice)
dotplot(dados$Etch.rate ~ dados$Potencia)
plot(mod1$fit,res)
abline(h=0)
```

Teste estatístico - Bartlett test

```
bartlett.test(dados$Etch.rate ~ dados$Potencia)
```

Análise gráfica

```
## Loading required package: lattice
```



Teste estatístico - Bartlett test

```
##
```

```
## Bartlett test of homogeneity of variances
```

```
##
```

```
## data: dados$Etch.rate by dados$Potencia
```

```
## Bartlett's K-squared = 0.43349, df = 3, p-value = 0.9332
```

- $p\text{-value} = 0,9332 > 0,05$
- Não existem evidências para rejeitar a homogeneidade de variâncias.

Análise gráfica

```
acf(res)
```

Teste estatístico

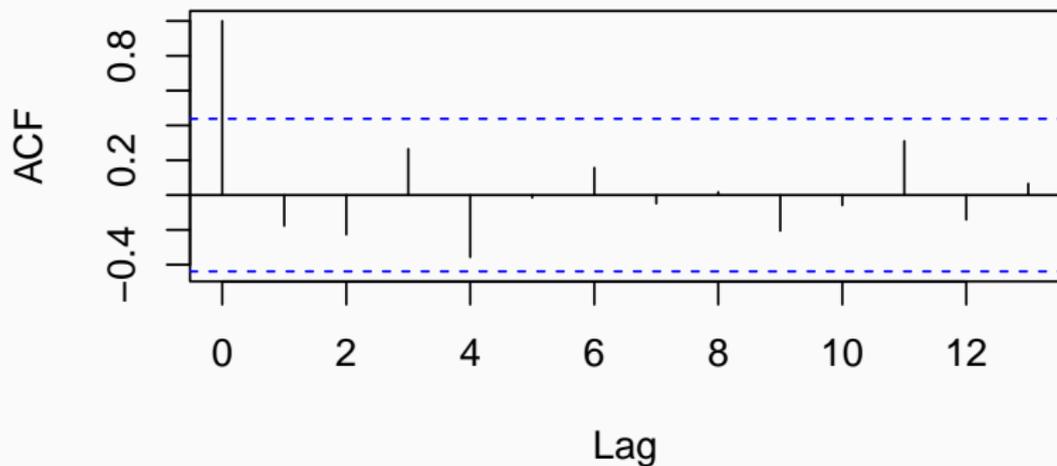
```
library(car)
```

```
durbinWatsonTest(mod1)
```

Teste para independência dos erros

Análise gráfica

Series res



```
## lag Autocorrelation D-W Statistic p-value
## 1 -0.1851588 2.262541 0.474
## Alternative hypothesis: rho != 0
```

- $p\text{-valor} = 0,094 > 0,05$
- Não existem evidências para rejeitar a independência dos erros considerando-se $\alpha = 0,05$.

Observe!

- Testar independência é algo complicado.
- Em muitas situações você precisa de sistema de coordenadas para as observações para ver se é detectado um padrão de dependência.
- Para um experimento planejado acreditamos que a independência seja garantida pela casualização, controle local, bordadura, teste cego, etc.

- Os pressupostos básicos foram atendidas, sendo assim, pode-se concluir sobre o teste F.
- Rejeita-se H_0 no nível de significância $\alpha = 5\%$
- Em média as taxas de gravuras diferem entre si.
- Ou ainda, pelo menos duas taxas de gravuras diferem entre si.

ANOVA

```
anova(mod1)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: Etch.rate
```

```
##           Df Sum Sq Mean Sq F value    Pr(>F)
```

```
## Potencia   3  66871 22290.2  66.797 2.883e-09 ***
```

```
## Residuals 16   5339   333.7
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

- <https://gorgens.github.io/compR-experimental/delineamento-inteiramente-casualizado.html>
- <https://ridiculas.wordpress.com/author/walmes/page/3/>