Aula Prática 01 - Estatística Experimental

DELINEAMENTO INTEIRAMENTE CASUALIZADO

```
> library(dae)
> n = 20
> Parcelas = list(Seedling = n)
> Trat = fac.gen(list(Trat = c("A", "B", "C", "D")), each = 5)
> Trat
```

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

```
> DIC = fac.layout(unrandomized = Parcelas, randomized = Trat,
+     seed = 105)
> DIC
```

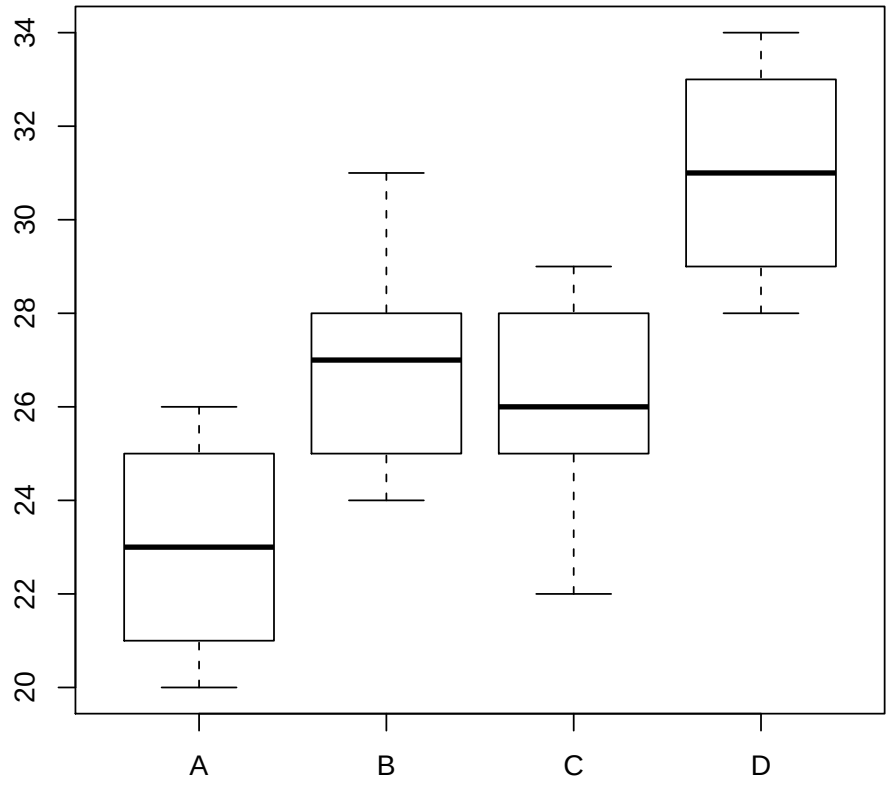
```
   Units Permutation Seedling Trat
1      1           2        1    B
2      2          20        2    A
3      3           4        3    C
4      4           7        4    A
5      5           9        5    D
6      6          14        6    D
7      7          17        7    A
8      8          15        8    B
9      9           8        9    A
10    10           1       10    C
11    11          12       11    D
12    12          18       12    C
13    13          10       13    D
14    14          16       14    B
15    15           3       15    B
16    16          13       16    C
17    17          11       17    B
```

```
18    18          19       18    C
19    19           6       19    D
20    20           5       20    A
```

```
> trat = c("A", "A", "A", "A", "A", "B", "B", "B", "B", "B", "C",
+     "C", "C", "C", "C", "D", "D", "D", "D", "D")
> prod = c(25, 26, 20, 23, 21, 31, 25, 28, 27, 24, 22, 26, 28,
+     25, 29, 33, 29, 31, 34, 28)
> dados = data.frame(trat, prod)
> dados
```

```
   trat prod
1     A   25
2     A   26
3     A   20
4     A   23
5     A   21
6     B   31
7     B   25
8     B   28
9     B   27
10    B   24
11    C   22
12    C   26
13    C   28
14    C   25
15    C   29
16    D   33
17    D   29
18    D   31
19    D   34
20    D   28
```

```
> boxplot(split(dados$prod, dados$trat))
```




```
> n = tapply(dados$prod, dados$trat, length)
> n

A B C D
5 5 5 5

> media = tapply(dados$prod, dados$trat, mean)
> media

 A  B  C  D
23 27 26 31

> variancia = tapply(dados$prod, dados$trat, var)
> variancia

  A   B   C   D
6.5 7.5 7.5 6.5

> desv.padr = tapply(dados$prod, dados$trat, sd)
> desv.padr
```

```
       A        B        C        D
2.549510 2.738613 2.738613 2.549510
```

```
> coef.var = (desv.padr)/(media) * 100
> coef.var
```

```
        A         B         C         D
11.084825 10.143010 10.533126  8.224225
```

```
> amplitude = tapply(dados$prod, dados$trat, range)
> amplitude
```

```
$A
[1] 20 26

$B
[1] 24 31

$C
[1] 22 29

$D
[1] 28 34
```

```
> soma = tapply(dados$prod, dados$trat, sum)
> soma
```

```
  A   B   C   D
115 135 130 155
```

```
> resumo = rbind(n, media, variancia, desv.padr, coef.var, soma)
> rownames(resumo) = c("Tamanho amostra", "Média", "Variância",
+     "Desvio-padrão", "Coeficiente de variação", "Soma")
> round(resumo, 2)
```

```
                            A      B      C      D
Tamanho amostra          5.00   5.00   5.00   5.00
Média                   23.00  27.00  26.00  31.00
Variância                6.50   7.50   7.50   6.50
Desvio-padrão            2.55   2.74   2.74   2.55
Coeficiente de variação 11.08  10.14  10.53   8.22
Soma                   115.00 135.00 130.00 155.00
```

```
> tapply(dados$prod, dados$trat, quantile)
```

```
$A
  0%  25%  50%  75% 100%
  20   21   23   25   26

$B
```

```
  0%  25%  50%  75% 100%
  24   25   27   28   31

$C
  0%  25%  50%  75% 100%
  22   25   26   28   29

$D
  0%  25%  50%  75% 100%
  28   29   31   33   34
```

```
> amplitude
```

```
$A
[1] 20 26

$B
[1] 24 31

$C
[1] 22 29

$D
[1] 28 34
```

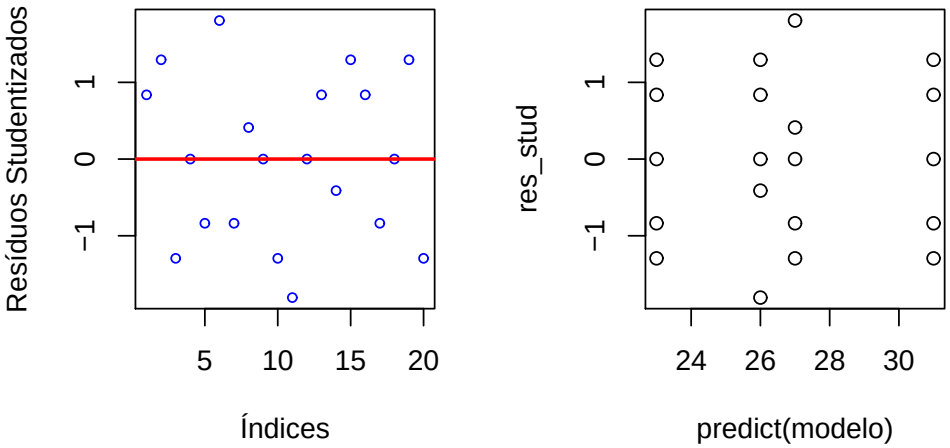
```
> modelo = aov(prod ~ trat, data = dados)
> res_ord <- residuals(modelo)
> res_pad <- rstandard(modelo)
> res_stud <- rstudent(modelo)
> var.res = tapply(res_stud, trat, var)
> var.res
```

```
       A        B        C        D
1.187936 1.451142 1.451142 1.187936
```

```
> Fmaximo = max(var.res)/min(var.res)
> Fmaximo
```

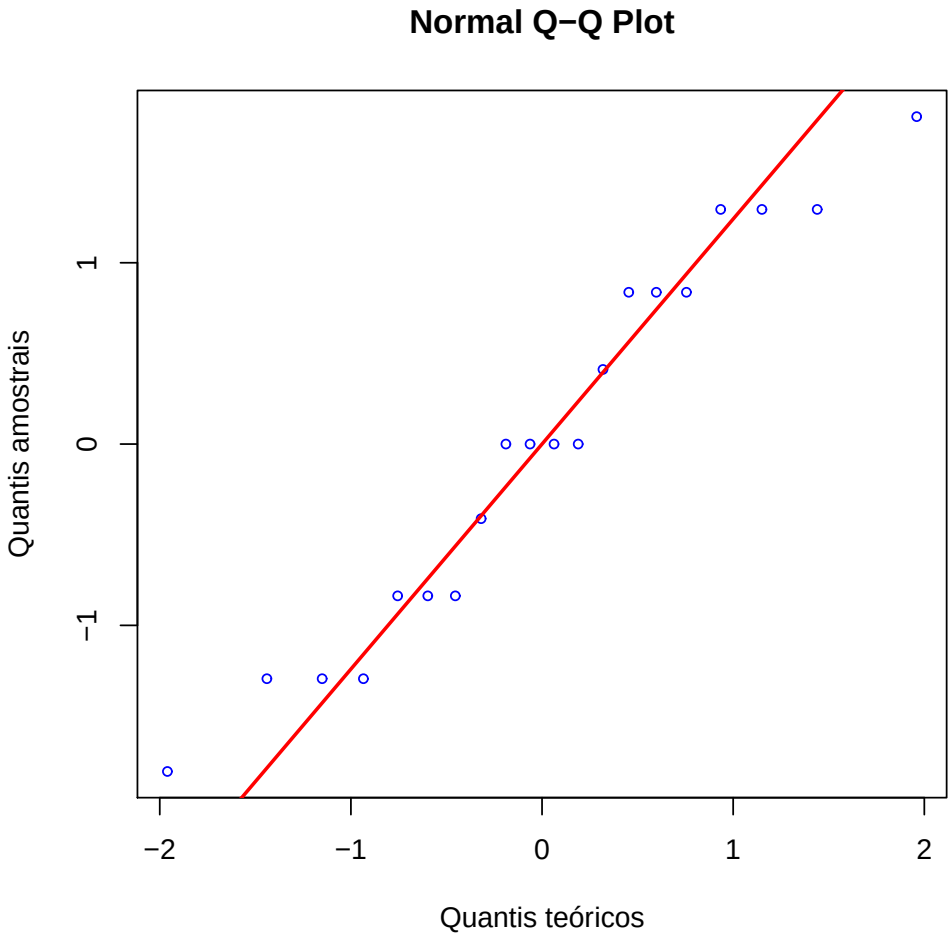
```
[1] 1.221566
```

```
> par(mfrow = c(1, 2), pty = "s")
> plot(res_stud, col = "blue", cex = 0.7, xlab = "Índices", ylab = "Resíduos Studentizados")
> abline(h = 0, col = "red", lwd = 2)
> plot(predict(modelo), res_stud)
```

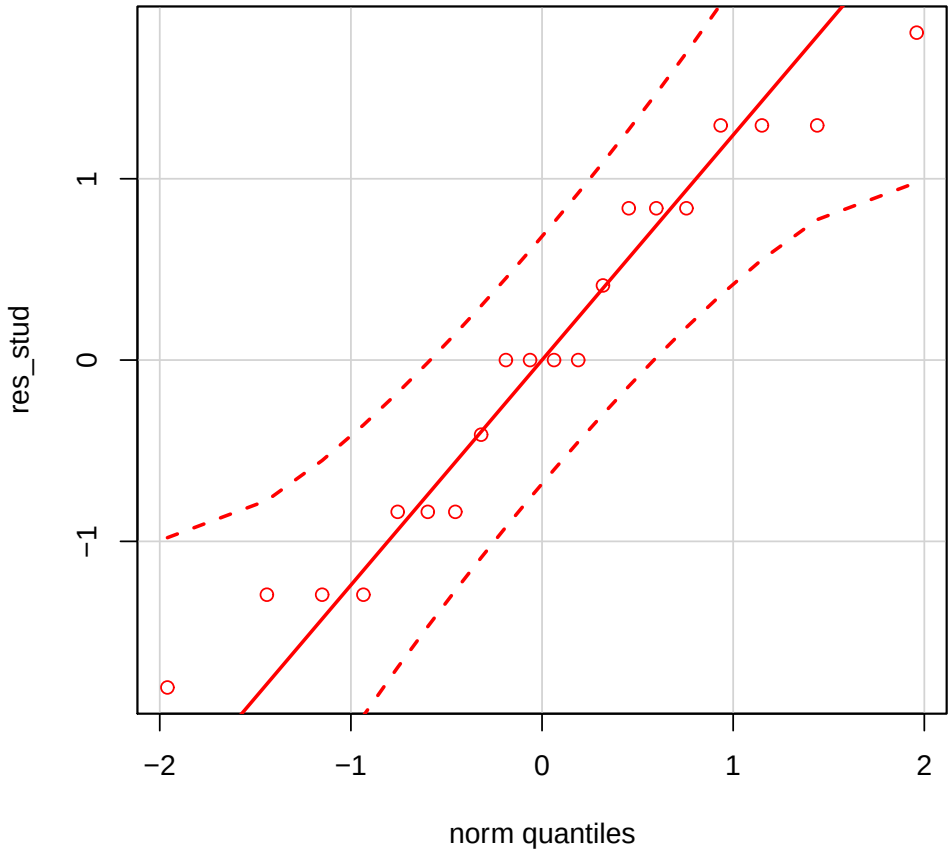



```
> shapiro.test(res_ord)

        Shapiro-Wilk normality test

data:  res_ord
W = 0.9396, p-value = 0.2359
```

```
> qqnorm(res_stud, col = "blue", cex = 0.7, xlab = "Quantis teóricos",
+     ylab = "Quantis amostrais")
> qqline(res_stud, col = "red", lwd = 2)
```




```
> require(car)
> qqPlot(res_stud)
```




```
> anova(modelo)

Analysis of Variance Table

Response: prod
          Df Sum Sq Mean Sq F value   Pr(>F)
trat       3 163.75  54.583  7.7976 0.001976 **
Residuals 16 112.00   7.000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


> require(laercio)
> LDuncan(modelo, "trat", conf.level = 0.95)


 DUNCAN TEST TO COMPARE MEANS

 Confidence Level:  0.95
 Dependent Variable:  prod
 Variation Coefficient:  9.89066 %
```

```
 Independent Variable:  trat
  Factors Means
  D       31    a
  B       27     b
  C       26     bc
  A       23      c
```

```
> LDuncan(modelo, "trat", conf.level = 0.99)
```

```
 DUNCAN TEST TO COMPARE MEANS

 Confidence Level:  0.99
 Dependent Variable:  prod
 Variation Coefficient:  9.89066 %


 Independent Variable:  trat
  Factors Means
  D       31    a
  B       27    ab
  C       26    ab
  A       23     b
```

```
> LTukey(modelo, "trat", conf.level = 0.95)
```

```
 TUKEY TEST TO COMPARE MEANS

 Confidence level:  0.95
 Dependent variable:  prod
 Variation Coefficient:  9.89066 %

Independent variable:  trat
  Factors Means
  D       31    a
  B       27    ab
  C       26     b
  A       23     b
```

```
> LTukey(modelo, "trat", conf.level = 0.99)
```

```
 TUKEY TEST TO COMPARE MEANS

 Confidence level:  0.99
 Dependent variable:  prod
 Variation Coefficient:  9.89066 %

Independent variable:  trat
  Factors Means
  D       31    a
  B       27    ab
  C       26    ab
  A       23     b
```

```
> tcm.tu = TukeyHSD(modelo)
> tcm.tu


  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = prod ~ trat, data = dados)

$trat
    diff        lwr       upr     p adj
B-A    4 -0.7874018  8.787402 0.1192178
C-A    3 -1.7874018  7.787402 0.3123298
D-A    8  3.2125982 12.787402 0.0010547
C-B   -1 -5.7874018  3.787402 0.9313122
D-B    4 -0.7874018  8.787402 0.1192178
D-C    5  0.2125982  9.787402 0.0391175


> plot(tcm.tu, col = "blue")
```