

Capítulo 6

Experimentos com um Fator de Interesse

Gustavo Mello Reis

José Ivo Ribeiro Júnior

Universidade Federal de Viçosa

Departamento de Informática

Setor de Estatística

Viçosa 2007

1. Introdução

O objetivo neste capítulo é de aplicar os testes de hipóteses paramétricos ou não paramétricos a um ou mais níveis qualitativos ou quantitativos de um fator (causa) controlável de interesse, denominado de variável X.

No caso dos paramétricos, têm-se os testes t, F, de qui-quadrado (χ^2), análise de variância (anova) e análise de regressão para os modelos lineares normais, segundo os delineamentos inteiramente (DIC) e em blocos casualizados (DBC).

2. Um Nível de X

Os dados serão lidos a partir de um arquivo externo, dados1.csv, que esta situado na pasta C:/Rdados.

```
dados1<- read.csv2("dados1.csv",dec=".")
```

```
dados1
```

	X	Y	YY
1	1	93.45	0
2	1	94.46	0
3	1	94.93	0
4	1	96.17	0
5	1	96.74	1
6	1	97.07	0
7	1	97.68	0
8	1	97.93	0
9	1	99.10	0
10	1	99.30	1
11	1	100.73	0
12	1	103.29	0
13	1	103.60	0
14	1	103.83	1
15	1	105.20	0

```
attach(dados1) # para utilizar as colunas separadamente
```

2.1. Testes de Aderência

Para $x = 1$, os n valores da variável resposta Y apresentam variações devidas a causas aleatórias, dado que não existe nenhuma outra informação de X . Portanto, Y é considerada uma variável aleatória e, conseqüentemente, possui uma distribuição de probabilidades.

Para verificar se Y segue distribuição normal podem ser utilizados um dos três métodos, além de outros: gráfico dos quantis normais e os testes de Lilliefors e de Kolmogorov-Smirnov.

2.1.1. Gráfico dos Quantis Normais

Na ajuda dessa função (?qqnorm), têm-se:

```
qqnorm(y, ylim = c(limite inferior, limite superior), main = "título do gráfico", xlab = "nome do eixo x", ylab = "nome do eixo y", plot.it = TRUE ou FALSE, datax = FALSE ou TRUE, ...)
```

```
qqline(y, datax = FALSE ou TRUE, ...)
```

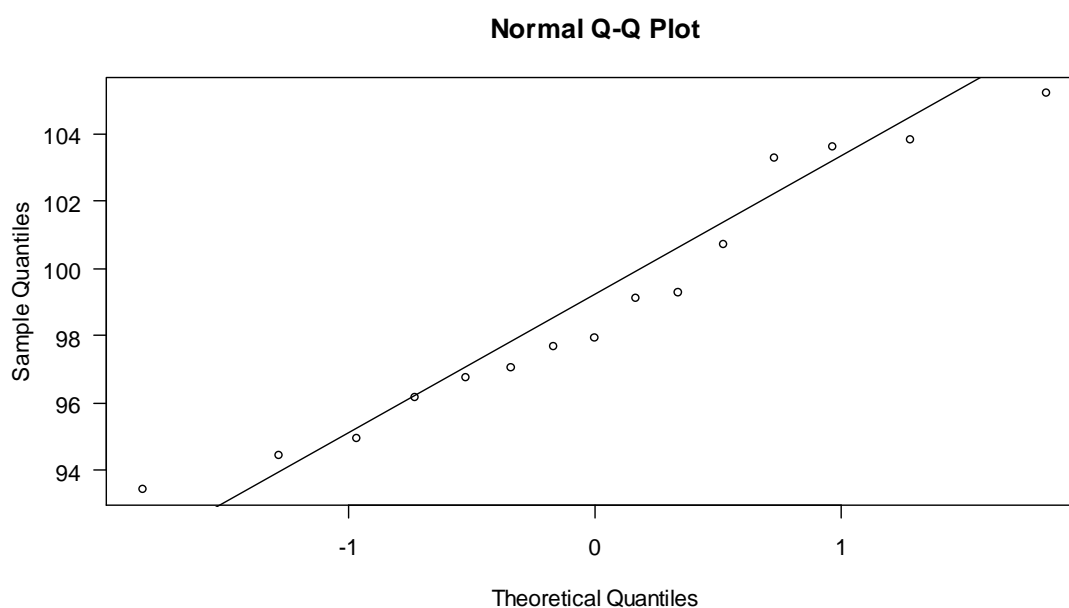
Se os valores dos argumentos ylim, main, xlab e ylab não forem fornecidos, o gráfico será construído com os seus valores padrões (default).

A primeira função cria a dispersão dos pontos [x = quantis teóricos (valores de z), y = quantis amostrais (valores de Y)] e, a segunda, a reta ajustada aos mesmos. Quando se usa datax = TRUE, os valores do eixo x serão os de y e os do eixo y serão os de x .

De forma simplificada, o gráfico (Figura 1) será construído da seguinte forma:

```
par(las=1) # Exibir os valores dos eixos y e x na horizontal
qqnorm(Y) # Construir o gráfico com os pontos
qqline(Y) # Inserir a linha ajustada
```

Figura 1. Gráfico dos quantis normais



De forma visual, se todos os pontos plotados estiverem próximos à reta, pode-se concluir que a variável Y tem distribuição normal, como apresentado no exemplo.

2.1.2. Teste de Lilliefors

Para esse teste será utilizado o pacote **nortest** que, após a instalação, deverá ser ativado pelo comando `library(nortest)`.

Utilizando a ajuda do R para visualizar as funções contidas no pacote **nortest**, tem-se: `help(package=nortest)` # Irá abrir uma nova janela e as funções estarão no tópico Index

O pacote **nortest** possui cinco funções: `ad.test` (teste de Anderson-Darling), `cvm.test` (teste de Cramer-von Mises), `lillie.test` (teste de Lilliefors), `pearson.test` (teste de Pearson qui-quadrado), `sf.test` (teste de Shapiro-Francia).

O teste de lilliefors será feito da seguinte forma:

```
library(nortest) # Abrir o pacote que contém a função do teste
lillie.test(Y)   # Fazer o teste
```

```
Lilliefors      (Kolmogorov-Smirnov)
normality test

data: Y
D = 0.1488, p-value = 0.4965
```

Desse modo, para $\alpha = 0,05$, não se rejeita a hipótese de normalidade dos dados de Y, dado que p-valor $> \alpha$.

2.1.3. Teste de Kolmogorov-Smirnov

Pela ajuda do programa R (`?ks.test`), tem-se:

```
ks.test(x, y, ..., alternative = c("two.sided" ou "less" ou "greater"))
```

O argumento x recebe os valores a serem testados e, y o nome da distribuição ajustada. No entanto, deve-se adicionar a letra p na frente do nome da distribuição, indicando que é a probabilidade acumulada da função.

Os três pontos indicam que outros argumentos podem ser utilizados. No caso da distribuição normal, os argumentos são mean e sd, que constituem a média e o desvio padrão, respectivamente.

De forma simplificada, o teste será feito da seguinte forma:

```
ks.test(Y, "pnorm",sd=sd(Y),mean=mean(Y))
```

```
One-sample Kolmogorov-Smirnov test

data:  Y
D = 0.1488, p-value = 0.847
alternative hypothesis: two.sided
```

Do mesmo modo, para $\alpha = 0,05$, conclui-se que Y é normal, para μ_Y e σ_Y estimados com base nos dados. No entanto, esse argumentos conferem a possibilidade de testar diferentes distribuições normais para diferentes combinações de μ_Y e σ_Y de interesses.

O gráfico (Figura 2) é construído através de:

```
plot(ecdf(Y), verticals = T) # Construir o gráfico
```

A função `ecdf()` é aplicada aos dados de Y para que eles sejam organizados de forma crescente. O argumento `verticals=T` indica que as linhas verticais, que ligam um ponto ao outro, também devem ser postas no gráfico. A curva pontilhada no gráfico (Figura 2) é construída através de:

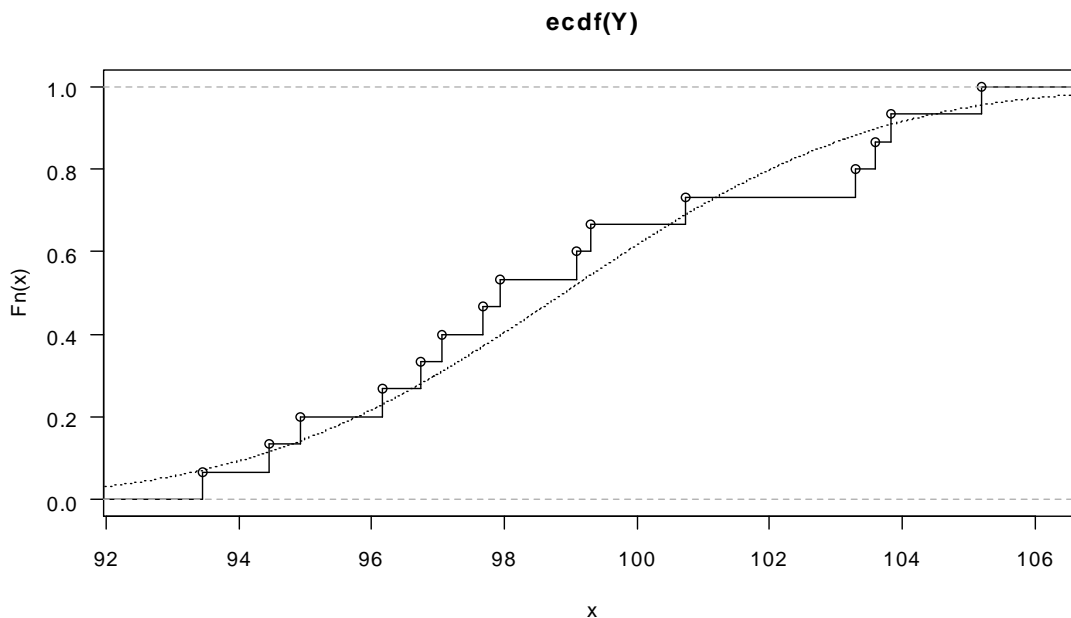
```
x<-seq(min(Y)-3, max(Y)+3, 0.01) # Criar as coordenadas teóricas para o eixo x
```

```
lines(x, pnorm(x, mean=mean(Y), sd=sd(Y)), lty=3) # Criar a curva teórica
```

O comando `seq(min(Y)-3, max(Y)+3, 0.01)` cria um vetor de números, separados por 0,01 unidades, entre o menor valor de Y subtraído de três unidades e o maior valor de Y adicionado de três unidades.

O comando `lines(x, pnorm(x, mean=mean(Y), sd=sd(Y)), lty=3)` é responsável por gerar a curva no gráfico. Este comando, como pode ser observado, possui três argumentos: o primeiro recebe o vetor x, que possui as coordenadas do eixo x para a curva; o segundo recebe as coordenadas do eixo y, criadas pelo comando `pnorm(x, mean=mean(Y), sd=sd(Y))`, que gera um vetor de probabilidades de acordo com a distribuição normal e os parâmetros de média e desvio padrão de Y; o terceiro (`lty`), indica o tipo de linha que deve ser traçada (3 = pontilhada).

Figura 2. Gráfico do teste de Kolmogorov-Smirnov



2.2. Teste t

Como foi verificado que os dados de Y seguem distribuição normal ($P > 0,05$), então procede-se à aplicação do teste t. No R, tem-se:

`t.test(x, y = NULL, alternative = "two.side" ou "less" ou "greater", mu = 0, paired = FALSE ou TRUE, var.equal = FALSE ou TRUE, conf.level = 1-0.05).`

Como exemplo, serão testadas as hipóteses $H_0 (\mu_Y = 100)$ e $H_a (\mu_Y \neq 100)$:

```
t.test(Y, mu=100) # Teste bilateral
```

```
One Sample t-test

data:  Y
t = -1.1519, df = 14, p-value = 0.2687
alternative hypothesis: true mean is not equal to 100
95 percent confidence interval:
 96.84797 100.94937
sample estimates:
mean of x
98.89867
```

Para $\alpha = 0,05$, não se rejeita H_0 , dado que $p\text{-valor} > \alpha$. Do mesmo modo, pode-se observar que $\mu_Y = 100$ pertence ao intervalo para a média μ_Y com $100(1-0,05)\%$ de confiança.

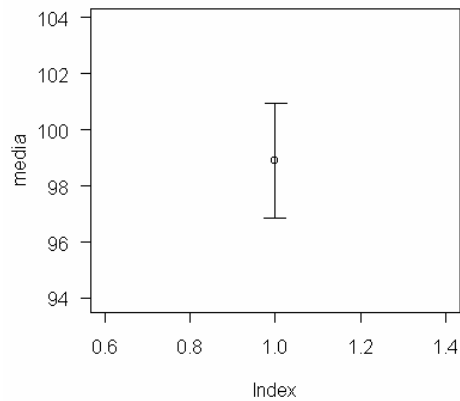
Para construir as barras de erros da média amostral de Y , será utilizado o intervalo de confiança fornecido pela função `t.test`. Primeiro, deve-se construir o gráfico com a média e especificar os limites do eixo y , que deve conter intervalos um pouco maiores que o intervalo de confiança. Para este exemplo tem-se:

```
media<-mean(Y)
plot(media, ylim=c(media-5, media+5))
```

Para adicionar as barras de erros, deve-se informar as coordenadas das barras. Para o eixo y , as coordenadas são os intervalos de confiança e, para o eixo x , são 1 e 1. Para o teste bilateral (Figura 3), tem-se:

```
arrows(1, 96.84797, 1, 100.94937, length=0.1,angle=90, code=3)
```

Figura 3. Estimativa do intervalo para μ_Y com $100(1-0,05)\%$ de confiança



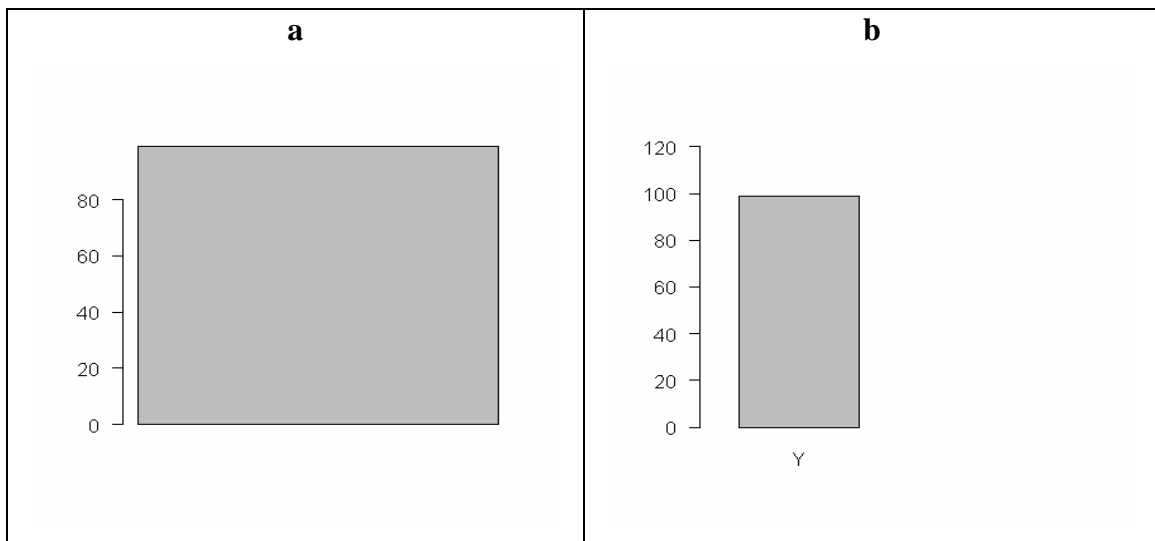
Para a construção de um outro gráfico, o de barras, basta utilizar o comando `barplot`, da seguinte forma:

```
media<-mean(Y)
```

```
barplot(media) # sem configuração (Figura 4a)
```

```
barplot(media,ylim=c(0,120),xlim=c(0,3),names="Y") # mais organizado (Figura 4b)
```

Figura 4. Estimativa da média

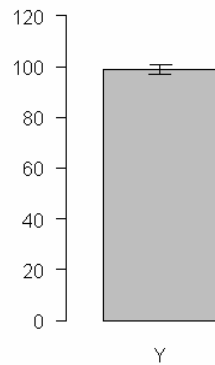


Para a representação do intervalo de confiança pela adição das barras (Figura 5), têm-se:

```
cx<-barplot(media, ylim=c(0,120), xlim=c(0,3), names="Y")
```

```
arrows(cx, 96.84797, cx, 100.94937, length=0.1, angle=90, code=3)
```

Figura 5. Estimativa da média e do intervalo para μ_Y com 100(1-0,05)% de confiança



3. Dois Níveis de X

3.1. Independentes

Como exemplo, considere as variáveis Y, YY, YYY e YYYY, cujos dados foram coletados em duas amostras diferentes (x_1 e x_2), de tamanhos iguais a cinco. Neste caso o arquivo de dados utilizado será “teste2i.csv”. No R tem-se:

```
dados2i<-read.csv2(“teste2i.csv”,dec=“.”)
```

```
dados2i
```

rept	X	Y	YY	YYY	YYYY
1	1	37.4	47.9	2	10
2	1	38.9	48.0	6	80
3	1	38.2	47.7	10	55
4	1	38.5	47.3	15	95
5	1	37.6	47.5	14	45
1	2	36.0	50.5	4	35
2	2	35.4	47.7	8	90
3	2	35.3	50.0	18	85
4	2	35.8	49.3	1	65
5	2	34.9	48.0	5	7

```
attach(dados2i)
```

3.1.1. Variáveis Y e YY Normais

Para $X = 1$, Y_1 é variável aleatória e para $X = 2$, Y_2 é variável aleatória. Portanto possuem uma distribuição de probabilidades.

Para verificar se Y_1 e Y_2 seguem a distribuição normal, deve-se aplicar um dos testes de normalidade a cada uma delas separadamente, da seguinte forma:

```
# Gráfico dos quantis normais
```

```
qqnorm(Y[X= =1])
```

```
qqline(Y[X= =1])
```

```
qqnorm(Y[X= =2])
```

```
qqline(Y[X= =2])
```

```
# Teste de Lilliefors
```

```
library(nortest) # Ativar o pacote que possui a função do teste de Lilliefors
```

```
lillie.test(Y[X= =1])
```

```
lillie.test(Y[X= =2])
```

```
# Teste de Kolmogorov-Smirnov
```

```
ks.test(Y[X= =1], "pnorm", mean=mean(Y[X= =1]), sd=sd(Y[X= =1]))
```

```
ks.test(Y[X= =2], "pnorm", mean=mean(Y[X= =2]), sd=sd(Y[X= =2]))
```

O mesmo deve ser feito para a variável YY.

3.1.1.1. Teste F

Para a aplicação desse teste, a pressuposição de normalidade deve ser verificada dentro de cada nível de X, ou seja, as variáveis respostas Y's têm distribuição normal nos níveis 1 e 2, separadamente.

No exemplo, como apenas as variáveis Y e YY apresentaram normalidade ($P > 0,05$) nos níveis 1 e 2 de X, então pode-se verificar a relação existente entre as variâncias das populações Y_1 e Y_2 e entre YY_1 e YY_2 , pelo teste F. Através do tópico Usage do help (?var.test), tem-se por default que ratio=1, ou, seja, que $\sigma^2_{Y1} / \sigma^2_{Y2} = 1$ e que $\sigma^2_{YY1} / \sigma^2_{YY2} = 1$, para a hipótese H_0 . Então, no R, para variável Y, têm-se:

`var.test(Y~X) # Comparar as variâncias das populações Y_1 e Y_2`

```
F test to compare two variances

data:  Y by X
F = 2.0695, num df = 4, denom df = 4, p-value = 0.4985
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2154732 19.8767544
sample estimates:
ratio of variances
      2.069519
```

`var.test(YY~X) # Comparar as variâncias das populações YY_1 e YY_2`

```
F test to compare two variances

data:  YY by X
F = 0.0548, num df = 4, denom df = 4, p-value = 0.01566
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.005710795 0.526803646
sample estimates:
ratio of variances
      0.0548495
```

Para a variável Y e $\alpha = 0,05$, conclui-se que as variâncias são homogêneas ($P > 0,05$). Por outro lado, a variável YY apresentou variâncias heterogêneas ($P < 0,05$).

Caso haja interesse em testar $\sigma^2_{Y1} > \sigma^2_{Y2}$ ou $\sigma^2_{Y1} < \sigma^2_{Y2}$, deve-se utilizar o argumento `alternative = "greater"` ou `"less"`, respectivamente.

3.1.1.2. Teste t

3.1.1.2.1. Variâncias Homogêneas

Em função do teste F, serão comparadas as médias da variável Y, sendo $H_0 (\mu_{Y1} = \mu_{Y2})$ e $H_a (\mu_{Y1} \neq \mu_{Y2})$:

`t.test(Y~X, var.equal=T) # Variâncias homogêneas`

```

Two Sample t-test

data:  Y by X
t = 7.7917, df = 8, p-value = 5.277e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.858676 3.421324
sample estimates:
mean in group 1 mean in group 2
      38.12      35.48

```

De acordo com os resultados, conclui-se pela rejeição de H_0 ($P > 0,05$).

Para a construção do intervalo com 100 (1- α)% de confiança para $\mu_{Y1} - \mu_{Y2}$ (Figura 6), será utilizado o fornecido pela função `t.test`, com $\alpha = 0,05$ (default), através da função `barplot`, da seguinte forma:

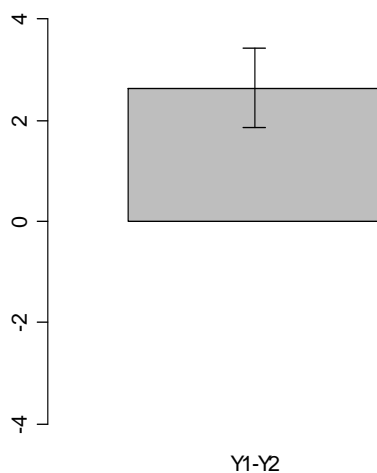
```

media12<-mean(Y[X= =1])-mean(Y[X= =2])
cx12<-barplot(media12, ylim=c(-4,4), xlim=c(0,3), names= "Y1-Y2")
arrows(cx12, 1.858676, cx12, 3.421324, length=0.1, angle=90, code=3)

```

Do mesmo modo, como $\mu_{Y1} - \mu_{Y2} = 0$ não pertence ao intervalo de confiança, então conclui-se que $\mu_{Y1} \neq \mu_{Y2}$. Na verdade $\mu_{Y1} > \mu_{Y2}$, dado que os dois limites são positivos.

Figura 6. Gráfico de barras com o intervalo com 100(1-0,05)% de confiança para $\mu_{Y1} - \mu_{Y2}$



3.1.1.2.2. Variâncias Heterogêneas

Em função do teste F, serão comparadas as médias da variável YY sendo $H_0 (\mu_{YY1} = \mu_{YY2})$ e $H_a (\mu_{YY1} \neq \mu_{YY2})$:

```
t.test(YY~X) # por default: var.equal = FALSE
```

```
Welch Two Sample t-test

data:  YY by X
t = -2.5285, df = 4.437, p-value = 0.05874
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.92047081  0.08047081
sample estimates:
mean in group 1 mean in group 2
      47.68      49.10
```

De acordo com os resultados, conclui-se pela não rejeição de H_0 ($P > 0,05$).

Da mesma forma vista para variável Y, os comandos para construir o intervalo com 100 (1- α)% de confiança para $\mu_{YY1} - \mu_{YY2}$ são:

```
media12<-mean(YY[X= =1])-mean(YY[X= =2])
cx12<-barplot(media12, ylim=c(-4,4), xlim=c(0,3), names= "YY1-YY2")
arrows(cx12, -2.92047081, cx12, 0.08047081, length=0.1, angle=90, code=3)
```

3.1.1.3. Anova DIC

Como exemplo, considere a variável resposta Y para os níveis 1 e 2 da variável X. A pressuposição imposta pela Anova, é que os erros experimentais ou resíduos associados a cada resposta Y tenham distribuição normal e variâncias homogêneas em todos os níveis de X.

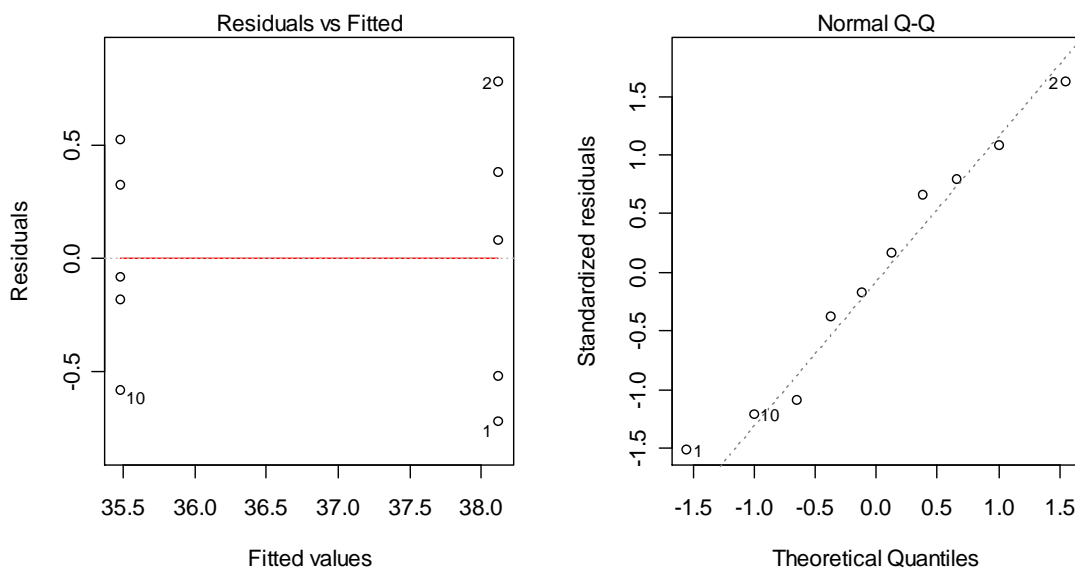
Para fazer e verificar as pressuposições da ANOVA, deve-se proceder os seguintes passos:

```
dic.y<-lm(Y~X)          # Montar o modelo
re.dic.y<-residuals(dic.y) # Armazenar os resíduos
re.dic.y                # Ver os resíduos
```

1	2	3	4	5	6	7	8	9	10
-0.72	0.78	0.08	0.38	-0.52	0.52	-0.08	-0.18	0.32	-0.58

```
# De forma visual
par(mfrow=c(1,2))
plot(dic.y,which=c(1,2)) # Ver os gráficos 1 e 2 (Figura 7)
```

Figura 7. Análises de resíduos



```
# Pelo teste de Lilliefors
```

```
library(nortest)
```

```
lillie.test(re.dic.y)
```

```
Lilliefors (Kolmogorov-Smirnov) normality test

data: re.dic.y
D = 0.1484, p-value = 0.7699
```

```
# Pelo teste de Kolmogorov-Smirnov
```

```
ks.test(re.dic.y, "pnorm", mean=0, sd=sd(re.dic.y))
```

```
One-sample Kolmogorov-Smirnov test

data: re.dic.y
D = 0.1484, p-value = 0.9578
alternative hypothesis: two.sided
```

Para testar se os resíduos têm variâncias homogêneas será utilizado o teste de Bartlett, da seguinte forma:

```
bartlett.test(re.dic.y,X)
```

```
Bartlett test of homogeneity of variances

data: re.dic.y and X
Bartlett's K-squared = 0.4602, df = 1, p-value = 0.4975
```

Assim, para $\alpha = 0,05$, conclui-se que os resíduos são normais e têm variâncias homogêneas.

Logo, a anova é apresentada como segue:

```
anova(dic.y) # Ver o quadro da anova
```

```
Analysis of Variance Table

Response: Y
      Df Sum Sq Mean Sq F value    Pr(>F)
X         1  17.424   17.424  60.711 5.277e-05 ***
Residuals  8   2.296    0.287
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A ANOVA também poderia ter sido obtida por meio de:

```
dic.y<-aov(Y~X)
```

3.1.2. Variáveis YYY e YYYY não Normais

3.1.2.1. Teste de χ^2

Com base nas variáveis YYY (números de defeitos) e YYYY (tamanho da amostra), deseja testar se a proporção média de itens defeituosos nos níveis 1 e 2 de X são iguais.

As hipóteses a serem testadas são: $H_0 (p_{X1} = p_{X2})$ e $H_a (p_{X1} \neq p_{X2})$. Para isto deve-se calcular a média de YYY e YYYY dentro cada nível de X:

```
tapply(YYY, X, mean)
```

```
  1      2
9.4  7.2
```

```
tapply(YYYY, X, mean)
```

```
  1      2
57.0 56.4
```

O teste será realizado da seguinte forma:

```
prop.test(c(9.4, 7.2), c(57, 56.4)) # por default (padrão): alternative = "two.sided"
```

```
2-sample test for equality of proportions with continuity
correction

data:  c(9.4, 7.2) out of c(57, 56.4)
X-squared = 0.0873, df = 1, p-value = 0.7676
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.1102547  0.1847601
sample estimates:
  prop 1    prop 2 
0.1649123 0.1276596
```

Para $\alpha = 0,05$, não rejeita-se H_0 , dado que p-valor $> \alpha$. Logo, as proporções de peças defeituosas nos níveis 1 e 2 de X, são estatisticamente iguais.

Caso haja interesse em testar $H_a (p_{X1} > p_{X2})$ ou $H_a (p_{X1} < p_{X2})$, deve-se utilizar o argumento `alternative = "greater"` ou `"less"`, respectivamente.

3.1.2.2. Teste de Wilcoxon-Mann-Whitney

Como no exemplo, as variáveis `YYY` (contagem) e `YYY / YYYYY` (proporção) não apresentaram distribuição aproximadamente normal nos níveis 1 e 2 de X ($P < 0,05$), então uma opção apropriada é aplicar o teste de Mann-Whitney, muitas vezes chamado de teste de Wilcoxon da soma dos postos, como é o caso no R.

```
wilcox.test(YYY~X) # Por default (mu = 0), o que significa  $H_0: \mu_{YYY1} - \mu_{YYY2} = 0$ 
```

```
Wilcoxon rank sum test
data:  YYY by X
W = 16, p-value = 0.5476
alternative hypothesis: true mu is not equal to 0
```

```
y34<-YYY/YYYYY # Criar um vetor com a proporção (YYY / YYYYY)
```

```
wilcox.test(y34~X) #  $H_0: \mu_{YYY1/YYYYY1} - \mu_{YYY2/YYYYY2} = 0$ 
```

```
Wilcoxon rank sum test
data:  y34 by X
W = 14, p-value = 0.8413
alternative hypothesis: true mu is not equal to 0
```

No exemplo, não se rejeita H_0 ($P > 0,05$), para YYY e $YYY/YYYY$.

Caso haja interesse em testar H_a ($\mu_{YYY1} > \mu_{YYY2}$) ou H_a ($\mu_{YYY1} < \mu_{YYY2}$), deve-se utilizar o argumento `alternative = "greater"` ou `"less"`, respectivamente.

3.2. Dependentes

Como exemplo, considere a variável resposta Y para os níveis qualitativos A e B da variável X (tratamentos) e para os níveis quantitativos 1, 2, 3, 4 e 5 da variável `bloco` ou para os níveis qualitativos `bloco1`, `bloco2`, `bloco3`, `bloco4` e `bloco5` da variável `XX`. Os dados serão lidos a partir do arquivo `C:/Rdados/teste2d.csv` pelo R da seguinte forma:

```
dados2d<-read.csv2("teste2d.csv", dec= ".")
```

```
dados2d
```

bloco	XX	X	Y	YYY	YYYY
1	bloco1	A	37.4	2	10
1	bloco1	B	36.0	4	35
2	bloco2	A	38.9	6	80
2	bloco2	B	35.4	8	90
3	bloco3	A	38.2	10	55
3	bloco3	B	35.3	18	85
4	bloco4	A	38.5	15	95
4	bloco4	B	35.8	1	65
5	bloco5	A	37.6	14	45
5	bloco5	B	34.9	5	7

```
attach(dados2d)
```

Os objetos `dados2d` e `dados2i` apresentam os mesmos valores para as variáveis Y , YYY e $YYYY$. A diferença é que o primeiro apresenta uma classificação das variáveis Y 's em função da coluna X (tratamentos) em ordem crescente, e o segundo em função da coluna `bloco`.

Em termos práticos, as variáveis `bloco` e `XX` são idênticas, porém o R adota diferentes procedimentos para cada uma delas. Portanto, deve-se considerá-las sempre de forma qualitativa. Desse modo, pode-se digitar os seus valores representados por "labels" ou "alfanuméricos". Neste caso, o R entende automaticamente como níveis qualitativos. Ou digitá-los como números e, no R, transformá-los para qualitativos, através da função `factor`.

O mesmo procedimento deve ser adotado aos tratamentos, ou seja, de caracterizá-los corretamente como qualitativos ou quantitativos.

XX # A mensagem “levels” indica que é um fator qualitativo

```
[1] bloco1 bloco1 bloco2 bloco2 bloco3 bloco3 bloco4 bloco4 bloco5  
bloco5  
Levels: bloco1 bloco2 bloco3 bloco4 bloco5
```

`blocoq<-factor(bloco) # Transformar em fator qualitativo`

`blocoq`

```
[1] 1 1 2 2 3 3 4 4 5 5  
Levels: 1 2 3 4 5
```

3.2.1. Variável Y Normal

3.2.1.1. Teste t

Antes da realização do teste t, deve-se verificar se as diferenças entre cada par de valores das duas amostras seguem distribuição normal. Para isso, pode-se utilizar um dos três métodos, como seguem, para a variável dif:

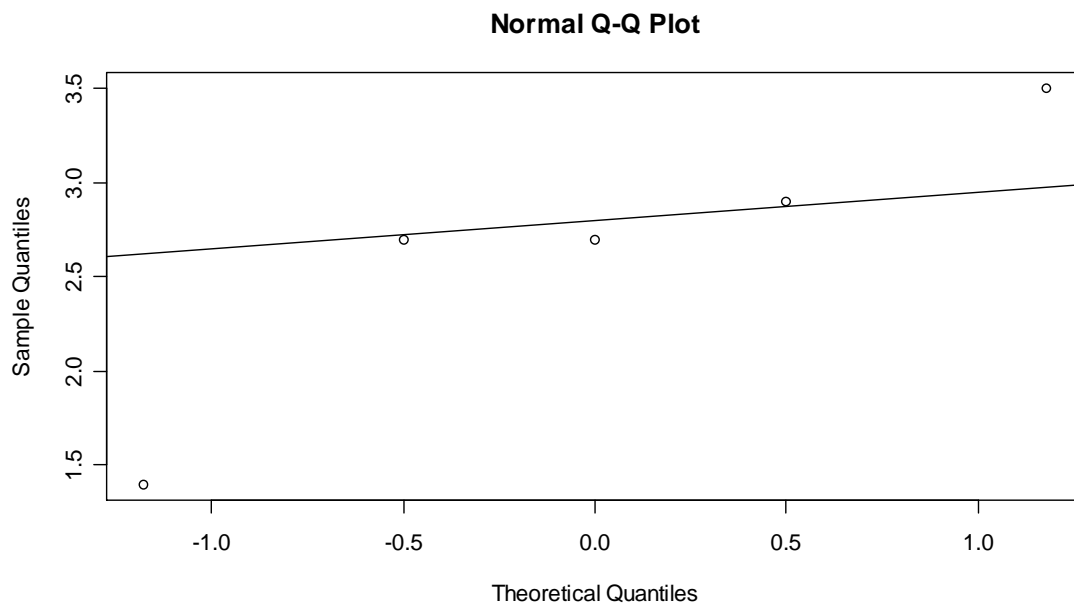
`# Gráfico do quantis normais (Figura 8)`

```
dif<-Y[X=="A"]-Y[X=="B"]
```

```
qqnorm(dif)
```

```
qqline(dif)
```

Figura 8. Gráfico dos quantis normais



```
# Teste de Lilliefors
```

```
library(nortest)
```

```
lillie.test(dif)
```

```
Lilliefors (Kolmogorov-Smirnov) normality test  
  
data: dif  
D = 0.3312, p-value = 0.07677
```

```
# Teste de Kolmogorov-Smirnov
```

```
ks.test(dif, "pnorm", mean= mean(dif), sd=sd (dif))
```

```
One-sample Kolmogorov-Smirnov test  
  
data: dif  
D = 0.3312, p-value = 0.6431  
alternative hypothesis: two.sided  
  
Warning message:  
não é possível calcular os níveis descritivos corretos com empates in:  
ks.test(dif, "pnorm", mean = mean(dif), sd = sd(dif))
```

Após satisfazer a pressuposição de normalidade ($P > 0,05$), pode-se fazer o teste t por meio de duas formas:

```
t.test(dif)
```

```
t.test(Y~X, paired= T)
```

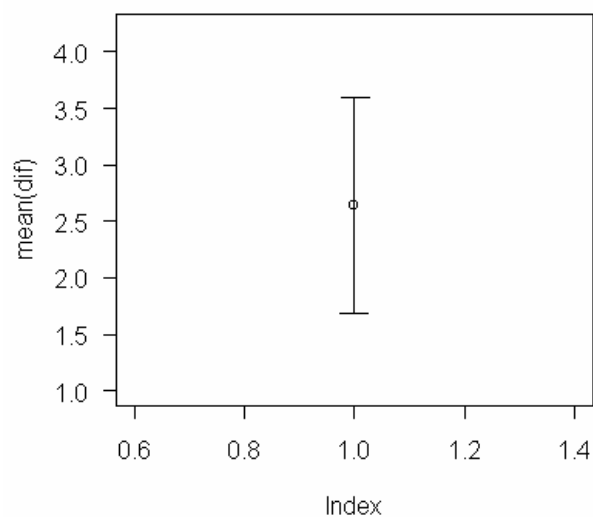
```
Paired t-test

data:  Y by X
t = 7.6984, df = 4, p-value = 0.001532
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.687878 3.592122
sample estimates:
mean of the differences
                2.64
```

Para fazer a barra de erro da diferença média (Figura 9) serão utilizados os intervalos fornecidos pelo teste t.

```
plot(mean(dif),ylim=c(1, 4.2))
arrows(1 , 1.687878, 1, 3.592122, length=0.1,angle=90, code=3)
```

Figura 9. Intervalo de confiança para a média das diferenças



3.2.1.2. Anova DBC

A análise de variância só poderá ser considerada após a verificação da normalidade e da homogeneidade de variâncias dos resíduos, como seguem:

```
dbc.y<-lm(Y~XX+X) # ou dbc.y<-aov(Y~XX+X)
```

```
re.dbc.y<-residuals(dbc.y)
```

```
re.dbc.y
```

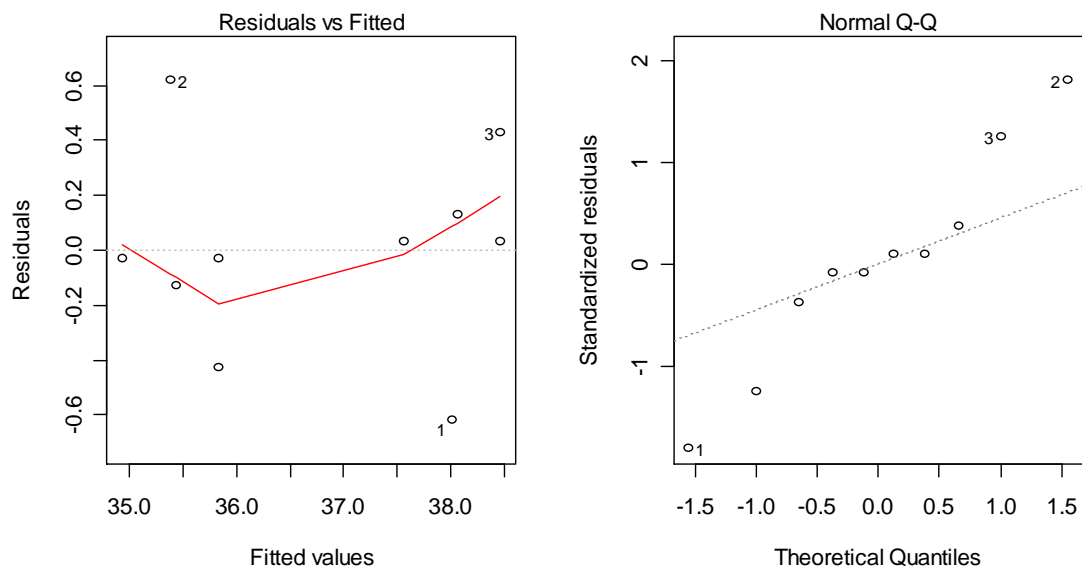
1	2	3	4	5	6	7	8	9	10
-0.62	0.62	0.43	-0.43	0.13	-0.13	0.03	-0.03	0.03	-0.03

```
# Verificar a normalidade de forma visual
```

```
par(mfrow=c(1,2))
```

```
plot(dbc.y,which=c(1,2)) # Figura 10
```

Figura 10. Análise de resíduos



```
# Teste de Lilliefors
```

```
library(nortest)
```

```
lillie.test(re.dbc.y)
```

```
Lilliefors (Kolmogorov-Smirnov) normality test  
  
data: re.dbc.y  
D = 0.1669, p-value = 0.5972
```

```
# Teste de Kolmogorov-Smirnov
```

```
ks.test(re.dbc.y, "pnorm", mean= 0, sd=sd(re.dbc.y))
```

```
One-sample Kolmogorov-Smirnov test

data:  re.dbc.y
D = 0.1669, p-value = 0.9017
alternative hypothesis: two.sided
```

```
# Teste de Bartlett
```

```
bartlett.test(re.dbc.y,X)
```

```
Bartlett test of homogeneity of variances

data:  re.dbc.y and X
Bartlett's K-squared = 0, df = 1, p-value = 1
```

Assim, como as pressuposições foram satisfeitas ($P > 0,05$), tem-se a anova:

```
anova(dbc.y)
```

```
Analysis of Variance Table

Response: Y

      Df Sum Sq Mean Sq F value    Pr(>F)
XX      4  1.120   0.280  0.9524 0.518289
X       1 17.424  17.424 59.2653 0.001532 **
Residuals 4  1.176   0.294
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3.2.2. Variáveis YYY e YYYY não Normais

3.2.2.1. Teste de Wilcoxon

Para as variáveis não normais YYY (contagem) e YYY / YYYY (proporção), têm-se:

```
wilcox.test(YYY~X,paired=T)
```

```
Wilcoxon signed rank test with continuity correction

data:  YYY by X
V = 8, p-value = 1
alternative hypothesis: true mu is not equal to 0

Warning message:
cannot compute exact p-value with ties in: wilcox.test.default(x =
c(2, 6, 10, 15, 14), y = c(4, 8, 18,
```

```
y34<-YYY/YYYY # Criar um vetor com a proporção (YYY / YYYY)
```

```
wilcox.test(y34~X, paired=T)
```

```
Wilcoxon signed rank test

data:  y34 by X
V = 12, p-value = 0.3125
alternative hypothesis: true mu is not
equal to 0
```

O aviso que aparece no teste de Wilcoxon para a variável YYY, chama a atenção para os dados utilizados na realização desse teste. Um p-valor exato será calculado quando a variável possuir tamanho menor que cinquenta (com valores finitos) e quando não houver repetições (ties) na diferença entre os valores da variável dentro de cada nível de X. Caso isto não ocorra, uma aproximação normal será usada.

Observe as diferenças para as variáveis YYY e y34 e veja que para YYY possui uma repetição (-2) enquanto que para y34 não possui.

```
YYY[X= "A"] - YYY[X= "B"]
```

```
[1] -2 -2 -8 14 7
```

```
y34[X= "A"] - y34[X= "B"]
```

```
[1] 0.08571429 -0.01388889 -0.02994652 0.14251012 0.21777778
```

4. Mais de Dois Níveis Qualitativos de X

4.1. DIC

Como exemplo, considere o arquivo C:/Rdados/teste3dic.csv de um experimento com cinco níveis qualitativos de X (tratamentos) e três repetições, onde foram avaliadas as variáveis Y (normal), YY (Poisson) e YY/YYY (binomial). Para ler o arquivo, tem-se:

```
dados3dic<-read.csv2("teste3dic.csv",dec=".")
```

```
dados3dic
```

rept	X	Y	YY	YYY
1	A	1.5	2	15
2	A	1.8	5	49
3	A	1.65	8	75
1	B	1.4	14	42
2	B	1.55	7	26
3	B	1.6	6	17
1	C	1.65	1	5
2	C	1.7	13	95
3	C	1.73	18	55
1	D	1.55	10	50
2	D	1.4	7	80
3	D	1.45	4	65
1	E	1.6	9	35
2	E	1.45	11	25
3	E	1.53	4	60

```
attach(dados3dic)
```

4.1.1. Variável Y Normal

4.1.1.1. Anova DIC

A normalidade será testada através do erro experimental (e_{ij}), pois y_{ij} não é variável aleatória.

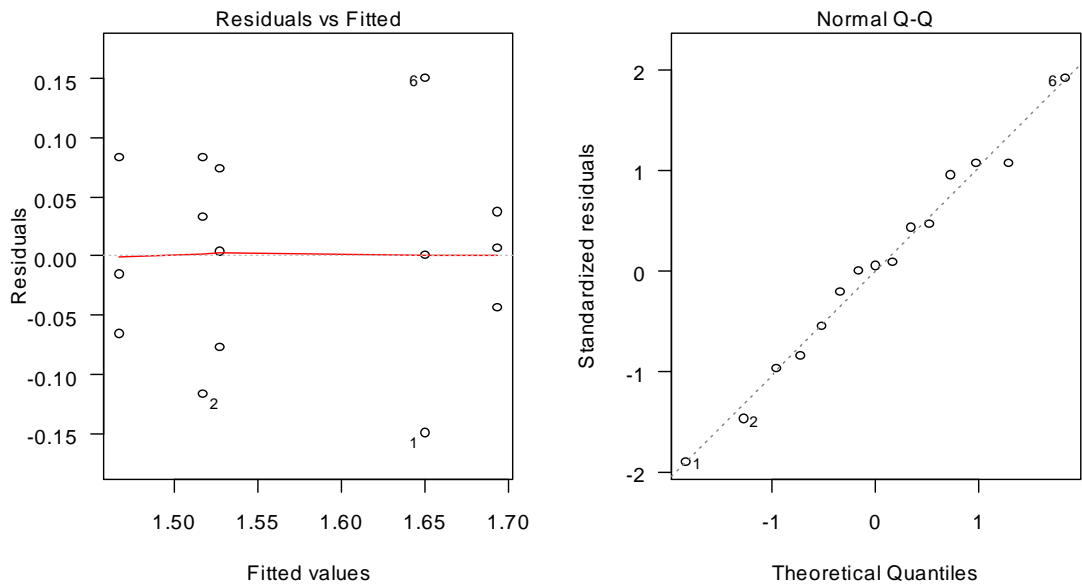
```
dic.y<-lm(Y~X)
re.dic.y<-residuals(dic.y)
bartlett.test(re.dic.y, X)
```

```
Bartlett test of homogeneity of variances

data: re.dic.y and X
Bartlett's K-squared = 2.7731, df = 4, p-value = 0.5965
```

```
par(mfrow=c(1,2))
plot(dic.y, which=c(1,2))
```

Figura 11. Análises de resíduos



```
anova(dic.y)
```

```
Analysis of Variance Table

Response: Y
      Df  Sum Sq Mean Sq F value Pr(>F)
X         4 0.111027  0.027757   2.9889 0.07297 .
Residuals 10 0.092867  0.009287
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4.1.1.2. Teste de Tukey

Para fazer o teste de Tukey o modelo deve ser montado utilizando a função aov, da seguinte forma:

```
dic.y<-aov(Y~X)
```

```
teste.dic.y<-TukeyHSD(dic.y, conf.level=(1 - 0.05)) # Por default  $\alpha = 0,05$ 
```

```
teste.dic.y      # Para ver o resultado do teste de Tukey
```

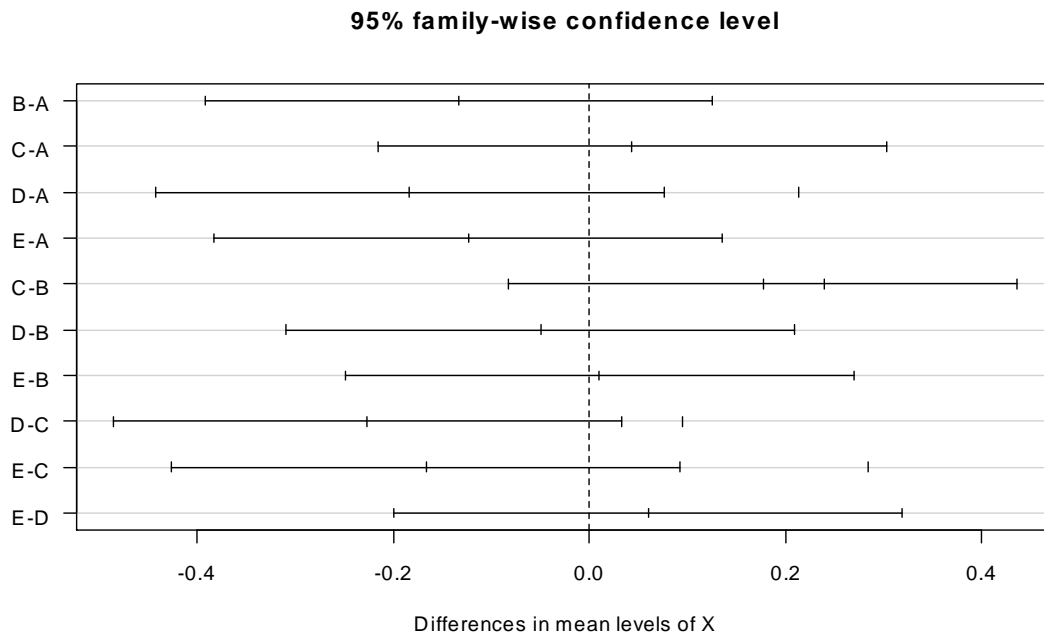
```
Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = Y ~ X)

$X
      diff      lwr      upr      p adj
B-A -0.13333333 -0.39228757 0.12562091 0.4779094
C-A  0.04333333 -0.21562091 0.30228757 0.9793603
D-A -0.18333333 -0.44228757 0.07562091 0.2124291
E-A -0.12333333 -0.38228757 0.13562091 0.5468424
C-B  0.17666667 -0.08228757 0.43562091 0.2391180
D-B -0.05000000 -0.30895424 0.20895424 0.9655836
E-B  0.01000000 -0.24895424 0.26895424 0.9999298
D-C -0.22666667 -0.48562091 0.03228757 0.0941497
E-C -0.16666667 -0.42562091 0.09228757 0.2841712
E-D  0.06000000 -0.19895424 0.31895424 0.9357538
```

```
plot(teste.dic.y) # Gerar o gráfico do teste de Tukey (Figura 12)
```

Figura 12. Intervalo de confiança para os contrastes estabelecidos



Os intervalos que sobrepõem a diferença igual a zero, indicam que as duas médias são semelhantes ($P > \alpha$).

4.1.2. Variáveis YY e YYY não Normais

4.1.2.1. Teste de Kruskal-Wallis

O teste de Kruskal-Wallis será aplicado às variáveis YY (contagem) e YY / YYY (proporção):

```
kruskal.test(YY~X)
```

```
Kruskal-Wallis rank sum test

data:  YY by X
Kruskal-Wallis chi-squared = 1.7981, df = 4, p-value = 0.7728
```

```
kruskal.test(YY/YYY~X)
```

```
Kruskal-Wallis rank sum test

data:  YY/YYY by X
Kruskal-Wallis chi-squared = 7.1878, df = 4, p-value = 0.1263
```

4.2. DBC

Como exemplo, considere o arquivo C:/Rdados/teste3dbc.csv, que em relação ao arquivo de dados utilizado no DIC, acrescentou-se a variável XX (bloco) e foi ordenado por esta. O arquivo será lido da seguinte forma:

```
dados3dbc<-read.csv2("teste3dbc.csv",dec=".")
dados3dbc
rept    XX    X    Y    YY    YYY
1      bloco1  A    1.5  2    15
1      bloco1  B    1.4  14   42
1      bloco1  C    1.65 1    5
1      bloco1  D    1.55 10   50
1      bloco1  E    1.6   9   35
2      bloco2  A    1.8   5   49
2      bloco2  B    1.55  7   26
2      bloco2  C    1.7  13   95
2      bloco2  D    1.4   7   80
2      bloco2  E    1.45 11   25
3      bloco3  A    1.65  8   75
3      bloco3  B    1.6   6   17
3      bloco3  C    1.73 18   55
3      bloco3  D    1.45  4   65
3      bloco3  E    1.53  4   60
attach(dados3dbc)
```

4.2.1. Variável Y Normal

4.2.1.1. Anova DBC

Como y_{ij} não é variável aleatória, a normalidade será testada através do erro experimental (e_{ij}), da seguinte forma:

```
dbc.y<-lm(Y~XX+X) # ou dbc.y<-aov(Y~XX+X)
```

```
re.dbc.y<-residuals(dbc.y)
```

```
bartlett.test(re.dbc.y, X)
```

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

```
data: re.dbc.y and X
```

```
Bartlett's K-squared = 5.277, df = 4, p-value = 0.2600
```

```
par(mfrow=c(1,2))
```

```
plot(dbc.y,wich=c(1,2))
```

Com a normalidade e a homogeneidade de variância testadas, tem-se a anova da seguinte forma:

```
anova(dbc.y)
```

4.2.1.2. Variáveis YY e YYY não Normais

4.2.1.2.1. Teste de Friedman

O teste de Friedman será aplicado como segue, às variáveis YY e YY / YYY:

```
friedman.test(YY~X|XX)
```

```
Friedman rank sum test
```

```
data: YY and X and XX
```

```
Friedman chi-squared = 1.5862, df = 4, p-value = 0.8113
```

```
friedman.test(YY/YYY~X|XX)
```

```
Friedman rank sum test
```

```
data: YY/YYY and X and XX
```

```
Friedman chi-squared = 7.9322, df = 4, p-value = 0.0941
```

4.3. Teste de χ^2

Como exemplo, será utilizado o arquivo de dados “prop.csv”, cujo a primeira coluna contém o número de defeitos (nd) e a segunda o tamanho da amostra (ta).

Os dados serão lidos da seguinte forma:

```
dados.prop<-read.csv2("prop.csv", dec= ".")
```

```
dados.prop
```

```
nd      ta
      83   86
      90   93
     129  136
      70   82
```

```
attach(dados.prop)
```

As hipóteses a serem testadas são:

Ho: As proporções de itens defeituosos são iguais em todas as amostras;

Ha: Pelo menos uma amostra possui proporção de itens defeituosos diferente das demais.

O teste será feito da seguinte forma:

```
prop.test(nd, ta)
```

```
4-sample test for equality of proportions without continuity
correction
```

```
data: nd out of ta
```

```
X-squared = 12.6004, df = 3, p-value = 0.005585
```

```
alternative hypothesis: two.sided
```

```
sample estimates:
```

```
prop 1    prop 2    prop 3    prop 4
0.9651163 0.9677419 0.9485294 0.8536585
```

5. Mais de Dois Níveis Quantitativos de X

5.1. Regressão de 1º Grau

Como exemplo, será utilizado o arquivo C:/Rdados/regressao1.csv, que será acessado por:

```
dados.reg1<-read.csv2("regressao1.csv", dec=".")
```

```
dados.reg1
```

```
 X      Y
 5    24.1
5.4   24.5
```

5.7	24.4
5.9	24.7
6.3	24.9
6.8	25.2
7.2	25.5
7.3	25.8
7.6	25.7
7.8	26

```
attach(dados.reg1)
```

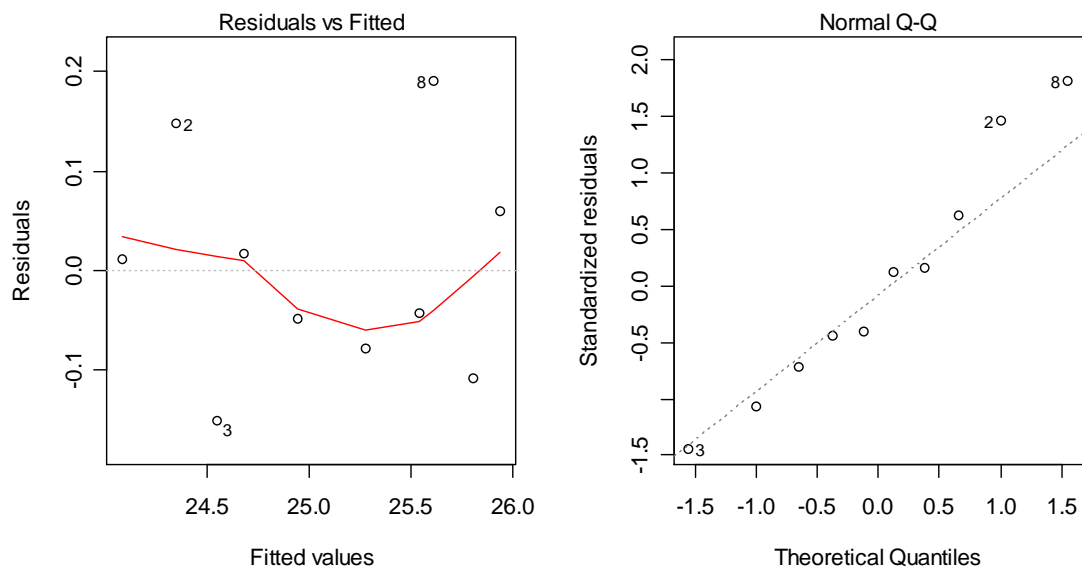
Para estudar a variável Y em função da variável X com dez níveis quantitativos, tem-se:

```
reg.y<-lm(Y~X)      # Montar o modelo
```

Porém, antes de interpretar a equação de regressão ajustada e a sua significância, é preciso verificar a validade das pressuposições de normalidade e de homogeneidade das variâncias dos erros experimentais por meio de:

```
par(mfrow=c(1,2))
plot(reg.y, which=c(1,2))
```

Figura 13. Análises de resíduos



De acordo com os gráficos (Figura 13), percebe-se que as pressuposições de normalidade e homogeneidade de variâncias foram satisfeitas

Para verificar a significância do modelo é necessário verificar a tabela da análise de variância (Teste F) ou utilizar a função summary (Teste t), que além dessa informação, apresenta as estimativas dos coeficientes de regressão.

anova(reg.y) # Teste F

```
Analysis of Variance Table

Response: Y

      Df Sum Sq Mean Sq F value    Pr(>F)
X         1  3.7691   3.7691  282.18 1.597e-07 ***
Residuals  8  0.1069   0.0134
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

summary(reg.y) # Teste t

```
Call:
lm(formula = Y ~ X)

Residuals:
    Min       1Q   Median       3Q      Max
-0.15100 -0.07072 -0.01550  0.04947  0.19100

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  20.78186    0.25847   80.40 6.38e-13 ***
X              0.66125    0.03936   16.80 1.60e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1156 on 8 degrees of freedom
Multiple R-Squared:  0.9724,    Adjusted R-squared:  0.969
F-statistic: 282.2 on 1 and 8 DF,  p-value: 1.597e-07
```

De acordo com os resultados obtidos, têm-se:

$\hat{Y} = 20,78186 + 0,66125**X$, em que ** significa: significativo pelo teste t ($P < 0,01$).

Uma outra maneira de ter acesso aos coeficientes de regressão, é, por meio de:

```
coef(reg.y)
```

(Intercept)	X
20.7818561	0.6612529

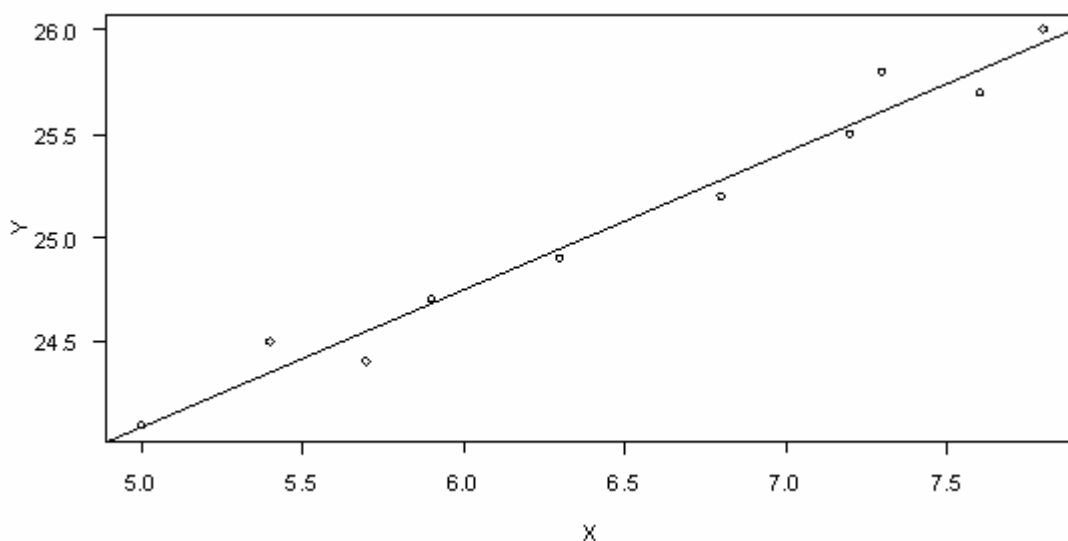
O gráfico de regressão linear de Y em função de X é construído por meio de:

```
plot(Y~X)           # Gerar o diagrama de dispersão dos pontos  
abline(reg.y)       # Traçar a reta ajustada aos pontos (Figura 14)
```

Outra forma de gerar o gráfico de regressão de Y em função de X seria:

```
re.reg.y<-residuals(reg.y) # Resíduos do modelo  
fit.reg.y<-Y-re.reg.y      # Valor de Y ajustado pelo modelo  
plot(fit.reg.y~X, type = "l") # Y ajustado em função de X, ligados por linha  
points(Y~X)                # Y em função de X, representados por pontos (Figura 14)
```

Figura 14. Estimativas de Y em função de X



Para estimar Y em função de um valor de X especificado dentro do intervalo estudado, será criada uma função da seguinte forma:

```
est.y<-function(reg, x) {coef(reg)[1]+coef(reg)[2]*x} # Criar a função  
est.y(reg.y, 6.25) # Valor de Y para X=6,25
```

(Intercept)
24.91469

5.2. Regressão de 2º grau

Como exemplo, será utilizado o arquivo C:/Rdados/regressao2.csv, que será lido por:

```
dados.reg2<-read.csv2("regressao2.csv", dec= ".")
```

```
dados.reg2
```

```
  X    Y
```

```
  5    10
```

```
 5.4  10.9
```

```
 5.7  11.4
```

```
 5.9  11.5
```

```
 6.3   12
```

```
 6.8  11.6
```

```
 7.2  11.1
```

```
 7.3  10.5
```

```
 7.6  10.1
```

```
 7.8   9.6
```

```
attach(dados.reg2)
```

Para a variável Y em função da variável X com níveis quantitativos, tem-se:

```
X2<-X^2          # Criar um vetor com o valor de X2
```

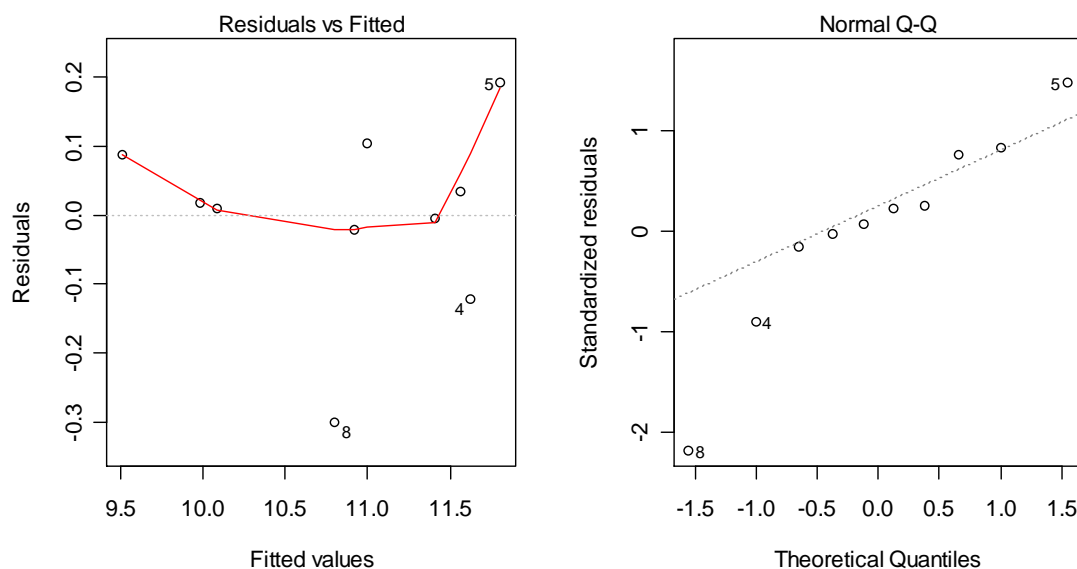
```
reg2.y<-lm(Y~X+X2) # Modelo de regressão
```

A verificação das pressuposições de normalidade e de homogeneidade de variâncias dos erros experimentais (Figura 15), é feita por meio de:

```
par(mfrow=c(1,2))
```

```
plot(reg2.y,which=c(1,2))
```

Figura 15. Gráficos dos resíduos para Y



Para verificar a significância do modelo, é necessário verificar a tabela da análise de variância:

`anova(reg2.y)`

```
Analysis of Variance Table

Response: Y

      Df Sum Sq Mean Sq F value    Pr(>F)
X         1  0.5463   0.5463  23.393  0.001885 **
X2        1  4.9313   4.9313 211.166 1.744e-06 ***
Residuals  7  0.1635   0.0234
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Para ver os coeficientes da regressão e construir o gráfico de Y em função de X (Figura 16), têm-se:

summary(reg2.y)

```
Call:
lm(formula = Y ~ X + X2)

Residuals:
    Min       1Q   Median       3Q      Max
-0.30087 -0.01629  0.01407  0.07451  0.19325

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -30.04554    2.94812  -10.19 1.89e-05 ***
X             13.24450    0.93021   14.24 2.00e-06 ***
X2            -1.04782    0.07211  -14.53 1.74e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

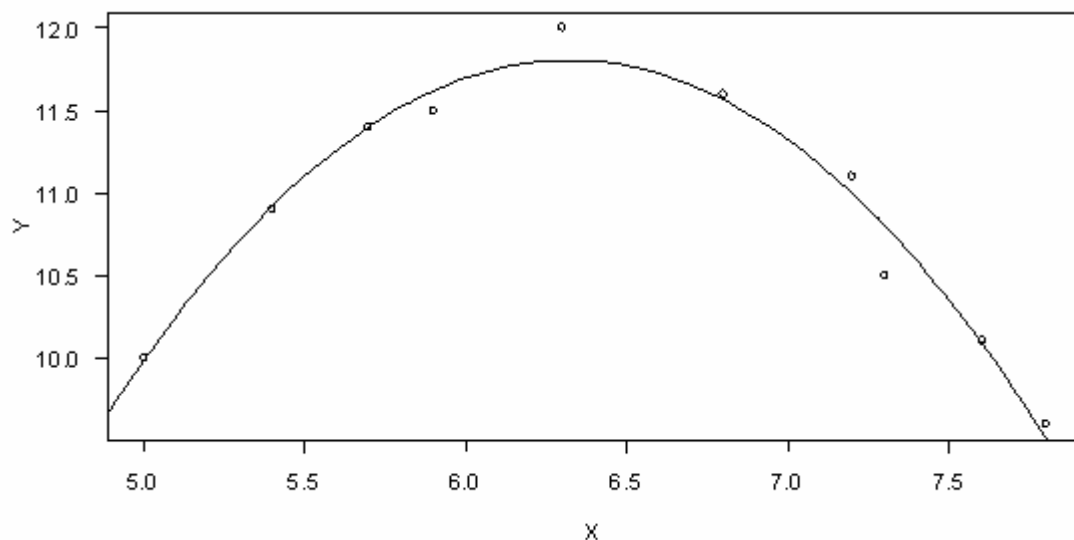
Residual standard error: 0.1528 on 7 degrees of freedom
Multiple R-Squared:  0.971,    Adjusted R-squared:  0.9627
F-statistic: 117.3 on 2 and 7 DF,  p-value: 4.143e-06
```

De acordo com os resultados do summary, tem-se: $\hat{Y} = -30,04554 + 13,24450X - 1,04782X^2$

plot(Y~X) # Gráfico de dispersão dos pontos

curve(-30.045545 + 13.244500*x + (-1.047822*x^2),add=T) # Curva ajustada

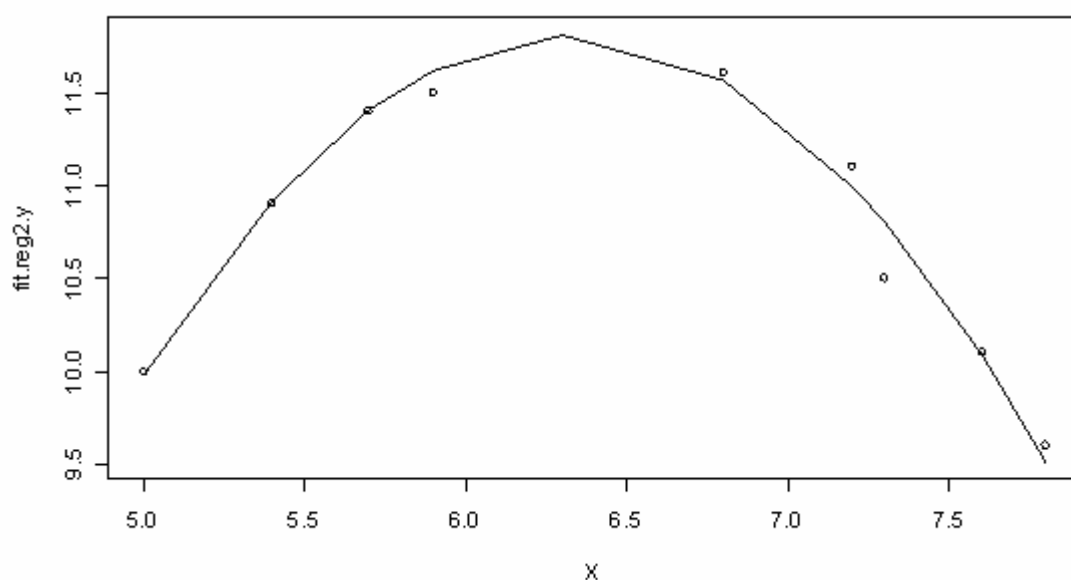
Figura 16. Estimativas de Y em função de X



Outra forma de gerar o gráfico de regressão de Y em função de X seria:

```
re.reg2.y<-residuals(reg2.y) # Resíduos do modelo
fit.reg2.y<-Y-re.reg2.y      # Valor de Y ajustado pelo modelo
plot(fit.reg2.y~X, type = "l") # Y ajustado em função de X, ligados por linha
points(Y~X)                  # Y em função de X, representados por pontos (Figura 17)
```

Figura 17. Estimativas de Y em função de X



Como pode-se perceber a segunda forma de gerar o gráfico, apesar de ser mais rápida, gera uma linha de regressão menos precisa, devido à pequena quantidade de amostras aqui utilizadas. Quando se está trabalhando com um número maior de amostras, o segundo método irá se aproximar ainda mais do primeiro.

Para estimar Y em função de um valor de X especificado dentro do intervalo, será criada a seguinte função:

```
est2.y<-function(reg, x) {coef(reg)[1]+coef(reg)[2]*x+coef(reg)[3]*x^2}
est2.y(reg2.y, 6.25) # Estimativa de Y para X = 6,25
```

```
(Intercept)
11.80203
```