

Introdução ao R com Aplicações

Lupércio França Bessegato
Augusto Carvalho Souza
Dep. de Estatística/UFJF

Visualização e Descrição de Dados

Roteiro Geral

1. Fundamentos da linguagem R
2. Visualização e descrição de dados
3. Inferência estatística básica
4. Modelos de regressão
5. Análise de dados multivariados
6. Séries temporais
7. Referências

Introdução ao R com Aplicações - 2017

2

Resumos Numéricos

- Principais medidas resumo para exploração de conjunto de dados:
 - √ Medidas de posição:
 - Média
 - Mediana
 - √ Medidas de dispersão:
 - Desvio-padrão (variância)
 - Distância interquartílica

Introdução ao R com Aplicações - 2017

4

Funções Mais Usadas

Símbolo	Função
<code>sum(x)</code>	Soma dos elementos de x
<code>prod(x)</code>	Produtório dos elementos de x
<code>max(x)</code>	Elemento máximo de x
<code>min(x)</code>	Elemento mínimo de x
<code>range(x)</code>	Elementos máximo e mínimo de x
<code>length(x)</code>	Quantidade de elementos do vetor x
<code>mean(x)</code>	Média dos elementos de x
<code>median(x)</code>	Mediana dos elementos de x
<code>var(x)</code>	Variância dos elementos de x
<code>sd(x)</code>	Desvio padrão dos elementos de x
<code>quantile(x, p)</code>	Quantil dos elementos de x , correspondente a p
<code>cor(x, y)</code>	Correlação entre os elementos de x e y

Introdução ao R com Aplicações - 2017
5

Exemplo

- Conjunto de dados de turma de alunos com as variáveis:
 - √ Sexo
 - √ Peso
 - √ altura

```
# Carregando e conhecendo o banco
Dados <- read.csv2(file = "turma.csv")
head(dados)
dim(dados)           # tamanho do conjunto de dados
attach(dados)
head(Altura)
head(Peso)
head(Sexo)
is.factor(Sexo)     # verifica se categórica está como fator
```

Introdução ao R com Aplicações - 2017
6

Função length(x)

- Calcula quantidade de elementos de vetor
- Verifica a quantidade de variáveis:

```
length(Peso)         # Calcula o tamanho da amostra
# usada no conjunto de dados
length(dados)       # Informa quantidade de variáveis
```


Introdução ao R com Aplicações - 2017
7

Valores Extremos


- Funções
 - √ `min(x)`: determina o menor valor da variável
 - √ `max(x)`: determina o maior valor da variável
 - √ `range(x)`: determina o menor e o maior valor da variável

```
min(Peso)           # Menor peso observado
max(Peso)           # Maior peso observado
range(Peso)         # Menor e maior peso observado (vetor)
```

Introdução ao R com Aplicações - 2017
8



Outras Funções



- Soma e produto:
 - √ `sum(x)`: soma todos os elementos de `x`
 - √ `prod(x)`: multiplica todos os elementos de `x`.

```


sum(Peso)           # soma todos os pesos observados
prod(Peso)          # multiplica todos os pesos observados
sum(Peso)/length(Peso) # cálculo do peso médio
            
```

- Média
 - √ `mean(x)`: médias dos elementos de `x`


```

mean(Peso)          # média dos pesos observados
mean(Altura)        # média das alturas observada
            
```

Introdução ao R com Aplicações - 2017
9



Função Aplicada a Grupos da Variável



- Determinação da média de alguns valores da variável.
 - √ Aplicando diretamente o comando `mean`

```


mean(Peso[Sexo=="F"]) # média dos pesos das alunas
mean(Peso[Sexo=="M"]) # média dos pesos das alunas
            
```

- √ Comando `tapply` e `aggregate`
 - Aplica função a cada grupo de valores dado por uma combinação única dos níveis de certos fatores.


```

# média da variável Peso por Sexo
tapply(Peso, Sexo, FUN = mean)
# média de todas as variáveis por Sexo
aggregate(dados[, -1], list(Sexo), mean)
            
```

Introdução ao R com Aplicações - 2017



Mediana




- `median(x)`: calcula mediana da variável observada.


```

median(Peso)        # mediana dos pesos de todos os alunos
median(Altura)      # mediana das alturas de todos os alunos
median(Peso[Sexo=="M"]) # mediana dos pesos dos alunos
            
```

Introdução ao R com Aplicações - 2017
12



Dispersão



- √ `var(x)`: variância dos elementos de `x`
- √ `sd(x)`: desvio padrão dos elementos de `x`.

```

var(Peso)           # variância do peso de todos os alunos
sd(Peso)            # desvio padrão do peso de todos os alunos
var(Peso[Sexo=="F"]) # variância do peso das alunas
sd(Altura[Sexo=="M"]) # desvio padrão da altura dos alunos
            
```

- √ Criação de função para coeficiente de variação:
 - # Coeficiente de variação - criação de função

```

cv <- function(x) sd(x)/mean(x)
cv(Peso)
            
```

Introdução ao R com Aplicações - 2017
13

DA
G3

Quantis

• `quantile(x, p)`: determina quantil, onde x é a variável observada e p é uma probabilidade.

```

quantile(Peso, 0.7)           # Percentil 70 dos pesos
quantile(Peso, c(0.25, 0.75)) # 1º e 3º quartis dos pesos
quantile(Peso[Sexo=="F"], 0.7) # Percentil 70 das alunas
quantile(Peso, 0.5)           # mediana de todos os pesos
    
```

Introdução ao R com Aplicações - 2017
14

DA
G3

Correlação

• Relação linear entre duas variáveis quantitativas

√ `corr(x, y)`: coeficiente de correlação linear entre as variáveis x e y .

```

cor(Peso, Altura)           # correlação linear entre peso e altura
cor.test(Peso, Altura)      # teste de significância da correlação
    
```

• Opções do comando:

√ `cor(x,y,method='pearson')`: default

√ `cor(x,y,method='spearman')`

√ `cor.test(x,y,method='pearson')`: default

√ `cor.test(x,y,method='spearman')`

Introdução ao R com Aplicações - 2017
15

DA
G3

Gráfico de dispersão

√ `plot(x,y)`: gráfico da relação das variáveis quantitativas x e y .

```

plot(Peso, Altura) # gráfico de dispersão entre Peso e Altura
    
```

Introdução ao R com Aplicações - 2017
16

DA
G3

Resumo de dos Dados

• Variáveis quantitativas:

√ Resumo de 5 números e média

– `summary(x)`: fornece o mínimo, 1º quartil, Mediana, 3º quartil, máximo e média dos elementos de x .



• Variáveis categóricas:

√ Tabela de frequências

```

summary(Peso)           # resumos da variável Peso
summary(Altura)         # resumos da variável Altura
    
```

Introdução ao R com Aplicações - 2017
17

Tabelas

- Resumo da frequência dos níveis de variável categórico (ou variável discreta).
- `table(x)`:

```
table(Sexo)           # tabela de contingência de Sexo
prop.table(table(Sexo)) # tabela de frequência relativa
```

18

Introdução ao R com Aplicações - 2017








Tabela de Frequência – Variável Contínua

- Não há comando específico no R. É necessário construí-la:
 √ Exemplo com o conjunto de dados `faithful`.

```
duracao <- faithful$eruptions
range(duracao)
# sequencia para intervalo dos dados (aproximado)
breaks <- seq(1.5, 5.5, by=0.5)
# aloca elementos em sub-intervalos de tamanho 0.5
duracao.cut <- cut(duracao, breaks, right=FALSE)
# calcula a frequência de erupções em cada sub-intervalo
duracao.freq <- table(duracao.cut)
# tabela com os resultados
cbind(duracao.freq)
```

19

Introdução ao R com Aplicações - 2017

Histograma



- Visualizando a variável `duracao`:

```
hist(duracao)
hist(duracao, label = T) # histograma com frequências

duracao.hist <- hist(duracao) # cria objeto com o histograma
str(duracao.hist)           # estrutura do objeto histograma
# limites dos sub-intervalos do histograma
duracao.hist$breaks
# frequência de valores em cada sub-intervalo
duracao.hist$counts
```

20

Introdução ao R com Aplicações - 2017

Geração de um Gráfico Aleatório

- Geração de 50 pontos ao acaso entre 0 e 2:

```
x <- runif(50, 0, 2)
y <- runif(50, 0, 2)
```

- Gráfico dos 50 pontos com título, subtítulo, rótulos eixos `x` e `y`:

```
plot(x, y, main = "Título Principal", sub = "Subtítulo",
      xlab = "nome_eixo_x", ylab = "nome_eixo_y")
```

21

Introdução ao R com Aplicações - 2017

DA
G3

Gráfico Gerado

DA
G3

√ O gráfico de cada um será diferente
– Se rodar de novo o resultado também será outro

Introdução ao R com Aplicações - 2017 22

DA
G3

Adição de Dados

DA
G3

- Adicionando texto e linhas ao gráfico

```
text(0.6, 0.6, "texto no pto (0.6,0.6)")
# linhas pelo ponto (0.6, 0.6)
abline(h = 0.6, v = 0.6)
```

- `abline(a, b)` plota a reta $y=a+bx$

Introdução ao R com Aplicações - 2017 23

DA
G3

Coordenadas das Margens

DA
G3

- Coordenadas das margens através função `mtext`

```
# coordenadas da margem
mtext(-1:4,side=1,at=0.7,line=-1:4)# coordenadas da margem
# loop para as coordenadas das margens
for(lado in 1:4) mtext(-1:4, side = lado, at = 0.7, line = -1:4)
# lado das margens
mtext(paste("lado", 1:4), side = 1:4, line = -1, font = 2)
```

Introdução ao R com Aplicações - 2017 24

DA
G3

Gráfico c/ Coordenadas Margens

DA
G3

- Layout de um gráfico padrão

Introdução ao R com Aplicações - 2017 25

DA
G3
Construindo um Gráfico por Partes

- Permite controle fino de cada elemento do gráfico
 - √ Desenha-se primeiro o gráfico sem os elementos

```
plot(x,y,type="n",xlab="",ylab="",axes=F) # plota-se nada!
```

- √ Os elementos serão adicionados subsequentemente

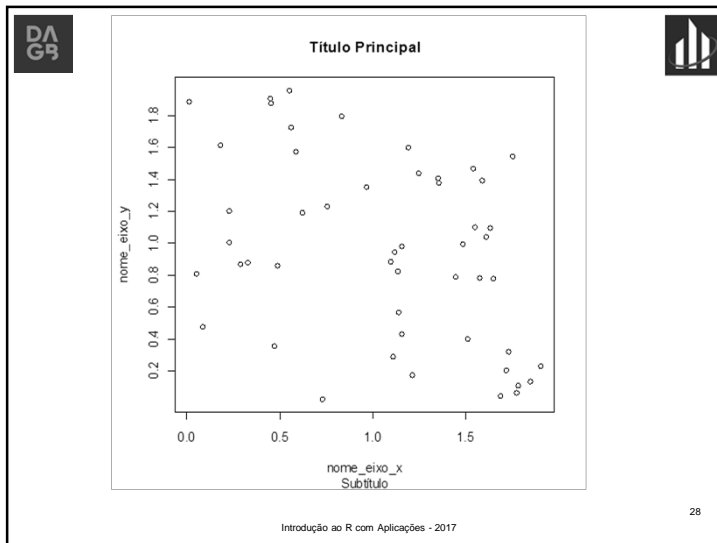
Introdução ao R com Aplicações - 2017
26

DA
G3
Montagem do Gráfico

- O gráfico pode ser montado executando cada comando por vez
 - √ Verifique o que acontecerá

```
points(x, y) # plota os pontos do gráfico
axis(1) # plota o eixo x
axis(2, at = seq(0.2, 1.8, 0.2)) # plota o eixo y
box() # caixa do gráfico
# Título, sub-título, nomes dos eixos
title(main = "Título Principal", sub = "Subtítulo",
      xlab = "nome_eixo_x", ylab = "nome_eixo_y")
```

Introdução ao R com Aplicações - 2017
27



DA
G3
Argumentos para Gráficos

- Argumentos mais usados

Table 2.1. A selective list of arguments to par().

Argument	Description
axes	should axes be drawn?
bg	background color
cex	size of a point or symbol
col	color
las	orientation of axis labels
lty, lwd	line type and line width
main, sub	title and subtitle
mar	size of margins
mfc, mfrow	array defining layout for several graphs on a plot
pch	plotting symbol
type	types (see text)
xlab, ylab	axis labels
xlim, ylim	axis ranges
xlog, ylog	log logarithmic scales

Introdução ao R com Aplicações - 2017
29

Histogramas

Construção de Histograma

- Geração de uma amostra aleatória com distribuição de frequências com simetria:

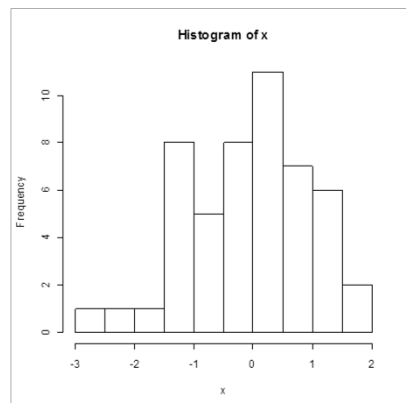
```
x <- rnorm(50)
```

- Construção do histograma (default)

```
hist(x)
```

Introdução ao R com Aplicações - 2017

31



✓ Repetir com outras amostras e verifique

Introdução ao R com Aplicações - 2017

32

Exemplo

- Levantamento de revistas econômicas.
 - ✓ Amostra aleatória com 180 observações
 - ✓ 10 variáveis (quantitativas e categóricas)
 - ✓ Período: 2000
 - ✓ Dados: *Journals{AER}* ou *Journals.csv*

Introdução ao R com Aplicações - 2017

34

• Importação dos dados – pacote AER:

```

> # carregamento direto do pacote
> data("Journals", package = "AER")
> help(Journals, package = "AER")
> revistas <- Journals
> str(revistas)
'data.frame':  180 obs. of  10 variables:
 $ title      : chr  "Asian-Pacific Economic Literature" "South African Journal of
 Economic History" "Computational Economics" "MOCT-MOST Economic Policy in
 Transitional Economics" ...
 $ publisher  : Factor w/ 52 levels "ANU Press","Academic Press",...: 11 45 28 28
 18 18 13 18 28 11 ...
 $ society    : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ price      : int  123 20 443 276 295 344 90 242 226 262 ...
 $ pages      : int  440 309 567 520 791 609 602 665 243 386 ...
 $ charpp     : int  3822 1782 2924 3234 3024 2967 3185 2688 3010 2501 ...
 $ citations  : int  21 22 22 22 24 24 24 27 28 30 ...
 $ foundingyear: int  1986 1986 1987 1991 1972 1994 1995 1968 1987 1949 ...
 $ subs       : int  14 59 17 2 96 15 14 202 46 46 ...
 $ field      : Factor w/ 24 levels "General","Economic History",...: 1 2 3 4 5
> dim(revistas)
[1] 180 10
    
```

Introdução ao R com Aplicações - 2017
36

• Importação pelo arquivo Journals.csv:

```

> # carregamento do arquivo csv
> revistas <- read.csv("Journals.csv")
> str(revistas)
'data.frame':  180 obs. of  11 variables:
 $ X          : Factor w/ 180 levels "AE","AEJ","AEL",...: 10 171 14 140 130 136 32
 166 45 144 ...
 $ title      : Factor w/ 180 levels "Agricultural Economics",...: 8 174 18 144 129
 137 47 168 43 143 ...
 $ publisher  : Factor w/ 52 levels "Academic Press",...: 11 45 28 28 18 18 13 18
 28 11 ...
 $ society    : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ price      : int  123 20 443 276 295 344 90 242 226 262 ...
 $ pages      : int  440 309 567 520 791 609 602 665 243 386 ...
 $ charpp     : int  3822 1782 2924 3234 3024 2967 3185 2688 3010 2501 ...
 $ citations  : int  21 22 22 22 24 24 24 27 28 30 ...
 $ foundingyear: int  1986 1986 1987 1991 1972 1994 1995 1968 1987 1949 ...
 $ subs       : int  14 59 17 2 96 15 14 202 46 46 ...
 $ field      : Factor w/ 24 levels "Agricultural Economics",...: 10 8 22 2 14 1
> dim(revistas)
[1] 180 11
    
```

Introdução ao R com Aplicações - 2017
37

• Conhecendo o conjunto de dados

```

> # carregamento direto do pacote
> data("Journals", package = "AER")
> help(Journals, package = "AER")
> revistas <- Journals
> names(revistas)
[1] "title"      "publisher"   "society"     "price"       "pages"
[6] "charpp"     "citations"  "foundingyear" "subs"        "field"
> head(revistas)
              title
APEL          Asian-Pacific Economic Literature
SAJoEH        South African Journal of Economic History
publisher society price pages charpp citations foundingyear
APEL          Blackwell      no    123  440 3822    21    1986
SAJoEH So Afr ec history assn no    20  309 1782    22    1986
subs          field
APEL          14      General
SAJoEH        59      Economic History
    
```

Introdução ao R com Aplicações - 2017
38

• Preparação do banco

$\sqrt{\text{Preço unitário por citação}}$

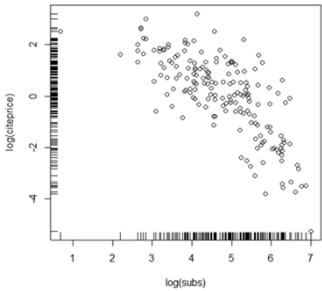
```

> revistas$citeprice <- revistas$price/revistas$citations
> # Anexando o conjunto de dados para trabalho
> attach(revistas)
    
```

Introdução ao R com Aplicações - 2017
39

DA G3 • *Plot* com distribuições marginais:

```
> plot(log(subs), log(citeprice))
> rug(log(subs))
> rug(log(citeprice), side = 2)
```



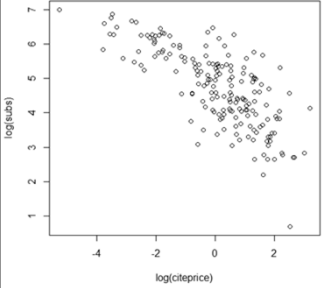
√ Comando rug: marcas para cada dado

Introdução ao R com Aplicações - 2017 40

DA G3 • *Plot* – comando alternativo

√ Sem o arquivo estar anexado

```
> detach(revistas)
> plot(log(subs) ~ log(citeprice), data = revistas)
```

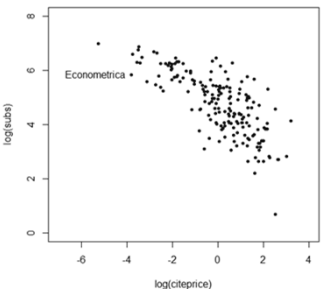


√ Formato Y ~ X

Introdução ao R com Aplicações - 2017 41

DA G3 • *Plot* – parâmetros gráficos:

```
> plot(log(subs) ~ log(citeprice), data = revistas, pch = 20, col = "blue",
+ ylim = c(0, 8), xlim = c(-7, 4), main = "Assinaturas Bibliotecas")
> text(-3.798, 5.846, "Econometrica", pos = 2)
```



√ Comando text

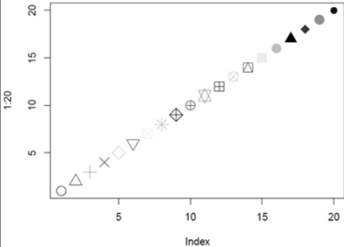
√ Parâmetros:

- Pch, col, xlim, ylim, main

Introdução ao R com Aplicações - 2017 42

DA G3 • **Exportação de Gráficos**

```
> pdf("meuArquivo.pdf", height = 5, width = 6)
> plot(1:20, pch = 1:20, col = 1:20, cex = 2)
> dev.off()
windows
2
```



√ Gráfico exportado para o pdf

√ Formatos de caracteres

√ Cores:

Introdução ao R com Aplicações - 2017 43

DA
G3
Anotação Matemática

- Inserção de símbolos matemáticos:

```
> curve(dnorm, from = -5, to = 5, col = "slategray", lwd = 3,
+ main = "Densidade da distribuição normal")
> text(-5, 0.3, expression(f(x) == frac(1, sigma * sqrt(2*pi)) *
+ e^{-frac((x - mu)^2, 2*sigma^2)}), adj = 0)
```

Densidade da distribuição normal

Introdução ao R com Aplicações - 2017 44

DA
G3
Sugestão

- Experimentar o comando:
 $\sqrt{\text{demo("plotmath")}}$

Introdução ao R com Aplicações - 2017 45

DA
G3
Exemplo – Simulação

- População com densidade em U:
 $\sqrt{f(x) = 12(x - 0.5)^2}$

$\sqrt{\text{Geração de 10.000 amostras dessa população}}$

Introdução ao R com Aplicações - 2017 46

DA
G3
Geração de números aleatória da função:

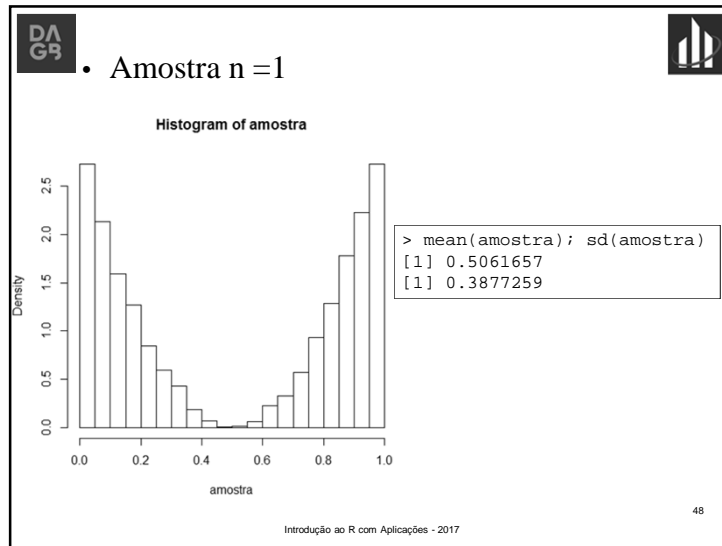
$\sqrt{\text{Funções}}$

```
> # carregamento direto do pacote
> # densidade da V.A.
> fx <- function(x) 12*(x - 0.5)^2
> # função distribuição acumulada da V.A.
> Fx <- function(x) 4*(x-0.5)^3+0.5
> # inversa da função distribuição acumulada
> invFx <- function(U){arg = U - 0.5; sign(arg)*(abs(arg)/4)^(1/3) + 1/2}
```

$\sqrt{\text{Geração de amostra unitária}}$

```
> # Amostra unitária (n = 1)
>
> tamanho <- 1
> U <- runif(run*tamanho)
> amostra <- invFx(U)
> hist(amostra, freq = F, xlim = c(0,1))
```

Introdução ao R com Aplicações - 2017 47



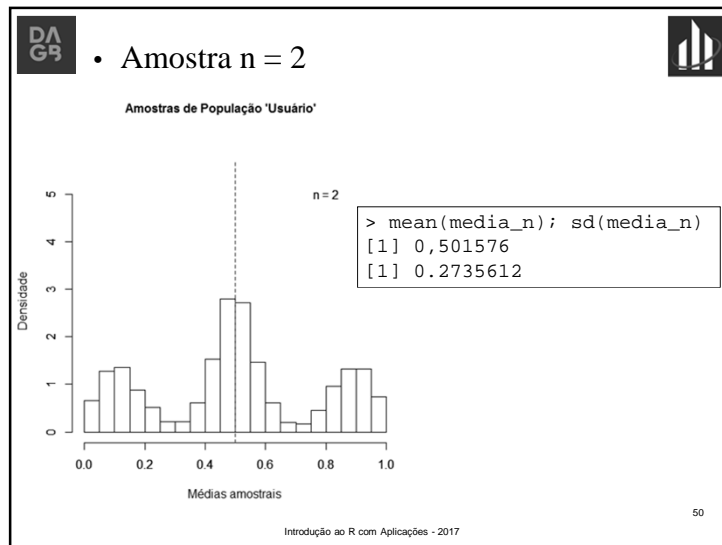
DA G3

√ Geração de amostra de tamanho n = 2

```
> # Amostra de tamanho n = 2
> tamanho <- 2
> U <- runif(run*tamanho)
> amostra.2 <- matrix(invF(x(U), nrow = run)
> media.2 <- rowMeans(amostra.2)
> hist(media.2, freq = F)
> abline(v = 0.5, lty = 2)
> text(0.8, 2, "n = 2")
> media <- mean(media.2)
> desvio <- sd(media.2)
> list(n = tamanho, media = media, dpadrao = desvio)
$n
[1] 2
$media
[1] 0.5015234
$dpadrao
[1] 0.2722762
```

Introdução ao R com Aplicações - 2017

49



DA G3

√ Comparação de médias amostrais

```
> par(mfrow = c(2, 2))
> esperado <- 0.5
> abs_x <- 0.8
> ord_y <- 5
> maximo <- 5.5
> tamanho
[1] 20
> tabela <- matrix(numeric(0), nrow = 4, ncol = 3)
> i = 0
> for(tamanho in c(2, 4, 10, 20)){
+ i = i + 1
+ U <- runif(run*tamanho)
+ amostra_n <- matrix(invF(x(U), nrow = run)
+ media_n <- rowMeans(amostra_n)
+ hist(media_n, freq = F, xlim = c(0, 1), ylim = c(0, maximo))
+ abline(v = esperado, lty = 2)
+ text(abs_x, ord_y, paste("n = ", tamanho))
+
+ media <- mean(media_n)
+ desvio <- sd(media_n)
+ tabela[i, ] <- c(round(tamanho), media, desvio)
+ }
> par(mfrow=c(1,1))
> colnames(tabela) <- c("n", "media", "padrao")
> tabela
```

Introdução ao R com Aplicações - 2017

51

• Amostras de tamanhos 2, 4, 10 e 20

	n	media	padrao
[1,]	2	0.4944818	0.27672467
[2,]	4	0.5012002	0.19319284
[3,]	10	0.4999705	0.12153137
[4,]	20	0.4996709	0.08681695

Introdução ao R com Aplicações - 2017

Família Apply

- Comando `apply`:
 - √ Usado para aplicar uma função a uma matriz
 - √ Saída: vetor com os resultados da função

```
> dados <- matrix(c(1:10, 21:30), nrow = 5, ncol = 4)
> dados
     [,1] [,2] [,3] [,4]
[1,]  1   6  21  26
[2,]  2   7  22  27
[3,]  3   8  23  28
[4,]  4   9  24  29
[5,]  5  10  25  30
> apply(dados, 1, mean)
[1] 13.5 14.5 15.5 16.5 17.5
```

- √ 1: aplica a linhas
- √ 2: aplica a colunas

Introdução ao R com Aplicações - 2017

- Comando `lapply`:
 - √ O mesmo que `apply`, mas aplica função a cada elemento de uma lista
 - √ Saída: lista com os resultados da função



```
> dados <- list(x = 1:5, y = 6:10, z = 11:15)
> dados
 $x
[1] 1 2 3 4 5
 $y
[1] 6 7 8 9 10
 $z
[1] 11 12 13 14 15
> sapply(dados, FUN = median)
 $x
[1] 3
 $y
[1] 8
 $z
[1] 13
```

Introdução ao R com Aplicações - 2017

- Comando `sapply`:
 - √ O mesmo que `lapply` (entrada é uma lista)
 - √ Saída: vetor com os resultados da função

```
> dados <- list(x = 1:5, y = 6:10, z = 11:15)
> dados
 $x
[1] 1 2 3 4 5
 $y
[1] 6 7 8 9 10
 $z
[1] 11 12 13 14 15
> sapply(dados, FUN = median)
 x y z
 3 8 13
```

Introdução ao R com Aplicações - 2017



- Comando `tapply`:
 - ✓ Aplica a função em cada parcela de vetor dividido com base a critério especificado (em geral, níveis de fator)
 - ✓ Saída: vetor com os resultados da função

```

> library(datasets)
> tapply(mtcars$wt, mtcars$ cyl, mean)
 4      6      8
2.285727 3.117143 3.999214
    
```

Introdução ao R com Aplicações - 2017

56



- Comando `mapply`:
 - ✓ É versão multivariada de `sapply`.
 - ✓ Aplica a função ao primeiro elemento de cada argumento, seguido pelo segundo elemento e assim por diante.

```

> x <- 1:5
> b <- 6:10
> mapply(sum, x, b)
[1] 7 9 11 13 15
    
```

Introdução ao R com Aplicações - 2017

57

- Comando `sweep`:
 - ✓ Similar ao comando `apply`.
 - ✓ Ações diferentes nos elementos MARGIN.

```

> dados <- matrix(c(1:10, 21:30), nrow = 5, ncol = 4)
> medias <- apply(dados, 2, mean)
> desvios <- apply(dados, 2, sd)
> sweep(dados, 2, medias, "-")
  [,1] [,2] [,3] [,4]
[1,] -2 -2 -2 -2
[2,] -1 -1 -1 -1
[3,] 0 0 0 0
[4,] 1 1 1 1
[5,] 2 2 2 2
    
```



✓ Comandos `sweep` aninhados

```

> sweep(sweep(dados, 2, medias, "-"), 2, desvios, "/")
  [,1] [,2] [,3] [,4]
[1,] -1.2649111 -1.2649111 -1.2649111 -1.2649111
[2,] -0.6324555 -0.6324555 -0.6324555 -0.6324555
[3,] 0.0000000 0.0000000 0.0000000 0.0000000
[4,] 0.6324555 0.6324555 0.6324555 0.6324555
[5,] 1.2649111 1.2649111 1.2649111 1.2649111
    
```

Introdução ao R com Aplicações - 2017

58



- Comando `by`:
 - ✓ Divide os dados por fatores e faz cálculos em cada subconjunto.
 - ✓ Retorna um objeto da classe "by"

```

> data(iris)
> by(iris[, 1:4], iris$Species, colMeans)
iris$Species: setosa
Sepal.Length Sepal.Width Petal.Length Petal.Width
 5.006      3.428      1.462      0.246
-----
iris$Species: versicolor
Sepal.Length Sepal.Width Petal.Length Petal.Width
 5.936      2.770      4.260      1.326
-----
iris$Species: virginica
Sepal.Length Sepal.Width Petal.Length Petal.Width
 6.588      2.974      5.552      2.026
    
```

Introdução ao R com Aplicações - 2017

59



• Comando `replicate`:

√ Avalia repetidamente uma função.

```
> replicate(4, rnorm(10))
      [,1]      [,2]      [,3]      [,4]
[1,] -0.6132738 -0.1607784 -0.670232639 -1.0644673
[2,] -2.6708596  0.3286909  1.912281711  2.6088157
[3,] -0.2547377  0.4903698  0.920117832 -0.5598459
[4,]  0.9225393 -0.3555194 -2.690485385 -2.5016242
[5,]  1.0573482  1.0674285  0.012816851  1.0838098
[6,] -0.4733374 -0.8163549 -0.454644535  0.1109681
[7,]  0.3551233 -0.5401991 -0.009457087 -0.3825013
[8,] -0.8488420 -0.6780893 -1.021707465  0.1051927
[9,] -0.5881127 -0.3190091 -0.765430543  2.8160557
[10,] -0.1602037  0.2605569  0.443206189 -0.6364794
```

Introdução ao R com Aplicações - 2017

60






Exemplo

- Experimento sobre influência de dieta no crescimento de pintos
 - √ Amostra aleatória com 578 observações
 - √ 4 variáveis (quantitativas e categóricas)
 - √ Fonte: <https://davetang.org/muse/2013/05/22/using-aggregate-and-apply-in-r/>
 - √ Dados: `ChickWeight{datasets}`

Introdução ao R com Aplicações - 2017

62






√ Variáveis:

- Time: número de dias entre o nascimento e a medição
- Chick: fator de identificação do pinto, com 50 níveis (18< ...)
- Diet: tipo de dieta recebida pelo pinto (níveis de 1 a 4)

Introdução ao R com Aplicações - 2017

63

• Importação dos dados:

```
> # carregamento dos dados
> dados <- ChickWeight
> help(ChickWeight)
> str(dados)
Classes 'nfnGroupedData', 'nfGroupedData', 'groupedData' and 'data.frame':   578
obs. of 4 variables:
 $ weight: num  42 51 59 64 76 93 106 125 149 171 ...
 $ Time  : num  0 2 4 6 8 10 12 14 16 18 ...
 $ Chick : Ord.factor w/ 50 levels "18"<"16"<"15"<...: 15 15 15 15 15 15 15 15 15 15 ...
...
 $ Diet  : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...

> str(dados)
 weight Time Chick Diet
1      42    0    1    1
2      51    2    1    1
3      59    4    1    1
4      64    6    1    1
5      76    8    1    1
6      93   10    1    1
```

Introdução ao R com Aplicações - 2017

64

DA G3 • Explorando as variáveis:

```

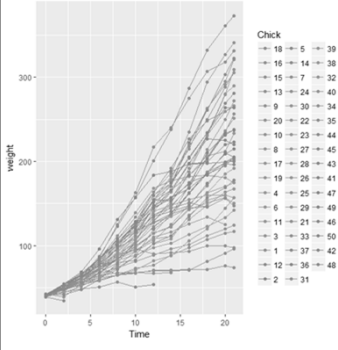
> # dimensão do conjunto de dados
> dim(dados)
[1] 578 4
> # quantidade de pintos
> unique(dados$Chick)
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
[26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
50 Levels: 18 < 16 < 15 < 13 < 9 < 20 < 10 < 8 < 17 < 19 < 4 < 6 < 11 < ... < 48
> length(unique(dados$Chick))
[1] 50
> #quantidade de dietas
> unique(dados$Diet)
[1] 1 2 3 4
Levels: 1 2 3 4
> # quantidade de instantes de tempo
> unique(dados$Time)
[1] 0 2 4 6 8 10 12 14 16 18 20 21
    
```

Introdução ao R com Aplicações - 2017 65

DA G3 • Visualização gráfica dos crescimentos:

```

> library(ggplot2)
> ggplot(data = dados, aes(x = Time, y = weight, group = Chick, colour=Chick)) +
+   geom_line() +
+   geom_point()
    
```



√ Crescimento individual do peso dos pintos
 √ Qual o efeito do tipo da dieta no crescimento?

Introdução ao R com Aplicações - 2017 66

DA G3 • Estatísticas descritivas:
 √ Uso do comando `aggregate`.

```

> # peso médio por dieta
> aggregate(dados$weight, list(diet = dados$Diet), mean)
  diet      x
1    1 102.6455
2    2 122.6167
3    3 142.9500
4    4 135.2627

> # média por instante de medição
> aggregate(dados$weight, list(time = dados$Time), mean)
  time      x
1    0  41.06000
2    2  49.22000
3    4  59.95918
4    6  74.30612
5    8  91.24490
6   10 107.83673
7   12 129.24490
8   14 143.81250
9   16 168.08511
10  18 190.19149
11  20 209.71739
12  21 218.68889
    
```

Introdução ao R com Aplicações - 2017 67

DA G3 √ Agregação por várias variáveis:

```

> # médias agregadas por time e diet
> head(aggregate(dados$weight,
+   list(time = dados$Time, diet = dados$Diet),
+   mean
+ )
+ )
  time diet      x
1    0  1  41.40000
2    2  1  47.25000
3    4  1  56.47368
4    6  1  66.78947

> tail(aggregate(dados$weight,
+   list(time = dados$Time, diet = dados$Diet),
+   mean
+ )
+ )
  time diet      x
45  16  4 182.0000
46  18  4 202.9000
47  20  4 233.8889
48  21  4 238.5556
    
```

Introdução ao R com Aplicações - 2017 68

DA G3 • Visualização dos crescimentos por dieta:

```
> ggplot(dados) + geom_line(aes(x = Time, y = weight, colour = Chick)) +
+   facet_wrap(~Diet) +
+   guides(col = guide_legend(ncol=3))
```

√ Qual o efeito do tipo da dieta no crescimento?

- Tendência
- Dispersão

Introdução ao R com Aplicações - 2017 69

Análise Exploratória de Dados

DA G3 **Exemplo**


- Current Population Survey, Maio/85.
 - √ Amostra aleatória com 534 observações
 - √ 11 variáveis (quantitativas e categóricas)
 - √ Dados: *CPS1985{AER}* ou *CPS1985.csv*

Introdução ao R com Aplicações - 2017 71

DA G3 √ Variáveis:

- wage: salário, em US\$ por hora
- education: anos de escolaridade
- experience: anos de experiência profissional potencial (age - education - 6).
- age: idade, em anos
- ethnicity: etnia. (cauc, hispanic, other)
- region: mora no Sul? (south, other)
- gender: sexo. (male, female).
- occupation: ocupação. (worker, technical, services, office, sales, management).
- sector: setor de ocupação. (manufacturing, construction, other).
- union: trabalho sindicalizado? (no, yes).
- married: É casado? (no, yes).

Introdução ao R com Aplicações - 2017 72


DA G3 • Importação dos dados – pacote AER: 

```
> # carregamento direto do pacote
> data("CPS1985", package = "AER")
> cps <- CPS1985
> head(cps)
  wage education experience age ethnicity region gender occupation
1    5.10         8         21 35 hispanic other female  worker
1100 4.95         9         42 57 cauc    other female  worker
2     6.67        12         1 19 cauc    other  male   worker

  sector union married
1  manufacturing no   yes
1100 manufacturing no  yes
2   manufacturing no   no

> str(CPS1985)
'data.frame':   534 obs. of  11 variables:
 $ wage      : num  5.1 4.95 6.67 4 7.5 ...
 $ education : num  8 9 12 12 12 13 10 12 16 12 ...
 $ experience: num  21 42 1 4 17 9 27 9 11 9 ...
 $ age       : num  35 57 19 22 35 28 43 27 33 27 ...
 $ ethnicity : Factor w/ 3 levels "cauc","hispanic",...: 2 1 1 1 1 1 1 1 1 1 ...
 $ region    : Factor w/ 2 levels "south","other": 2 2 2 2 2 1 2 2 2 ...
 $ gender    : Factor w/ 2 levels "male","female": 2 2 1 1 1 1 1 1 1 ...
 $ occupation: Factor w/ 6 levels "worker","technical",...: 1 1 1 1 1 1 1 1 1 1 ...
 $ sector    : Factor w/ 3 levels "manufacturing",...: 1 1 1 3 3 3 3 3 1 3 ...
 $ union     : Factor w/ 2 levels "no","yes": 1 1 1 1 1 2 1 1 1 1 ...
 $ married   : Factor w/ 2 levels "no","yes": 2 2 1 1 2 1 1 1 2 1 ...
```

Introdução ao R com Aplicações - 2017


DA G3 • Importação pelo arquivo CPS1985.csv: 

```
> # carregamento do arquivo csv
cps <- read.csv("CPS1985.csv")
> head(cps)
  wage education experience age ethnicity region gender occupation
1    5.10         8         21 35 hispanic other female  worker
1100 4.95         9         42 57 cauc    other female  worker
2     6.67        12         1 19 cauc    other  male   worker

  sector union married
1  manufacturing no   yes
1100 manufacturing no  yes
2   manufacturing no   no

> str(CPS1985)
'data.frame':   534 obs. of  11 variables:
 $ wage      : num  5.1 4.95 6.67 4 7.5 ...
 $ education : num  8 9 12 12 12 13 10 12 16 12 ...
 $ experience: num  21 42 1 4 17 9 27 9 11 9 ...
 $ age       : num  35 57 19 22 35 28 43 27 33 27 ...
 $ ethnicity : Factor w/ 3 levels "cauc","hispanic",...: 2 1 1 1 1 1 1 1 1 1 ...
 $ region    : Factor w/ 2 levels "south","other": 2 2 2 2 2 1 2 2 2 ...
 $ gender    : Factor w/ 2 levels "male","female": 2 2 1 1 1 1 1 1 1 ...
 $ occupation: Factor w/ 6 levels "worker","technical",...: 1 1 1 1 1 1 1 1 1 1 ...
 $ sector    : Factor w/ 3 levels "manufacturing",...: 1 1 1 3 3 3 3 3 1 3 ...
 $ union     : Factor w/ 2 levels "no","yes": 1 1 1 1 1 2 1 1 1 1 ...
 $ married   : Factor w/ 2 levels "no","yes": 2 2 1 1 2 1 1 1 2 1 ...
```

Introdução ao R com Aplicações - 2017

DA G3 • Abreviação níveis do fator occupation 

√ Dados carregados do pacote

```
> levels(cps$occupation)
[1] "worker" "technical" "services" "office" "sales"
[6] "management"
> levels(cps$occupation)[c(2, 6)] <- c("mgmt", "techn")
> levels(cps$occupation)
[1] "worker" "techn" "services" "office" "sales" "mgmt"
```


√ Dados carregados do arquivo csv

```
> levels(cps$occupation)
[1] "management" "office" "sales" "services" "technical"
[6] "worker"
> levels(cps$occupation)[c(1, 5)] <- c("mgmt", "techn")
> levels(cps$occupation)
[1] "mgmt" "office" "sales" "services" "techn" "worker"
```

√ Anexando conjunto de dados para trabalho:

```
> # anexando o arquivo
> attach(cps)
```

Introdução ao R com Aplicações - 2017

DA G3 • Distribuição de wage na amostra: 

√ Resumo dos 5 números e média

```
> summary(wage)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 1.000  5.250   7.780   9.024 11.250  44.500

> mean(wage)
[1] 9.024064
> median(wage)
[1] 7.78
```

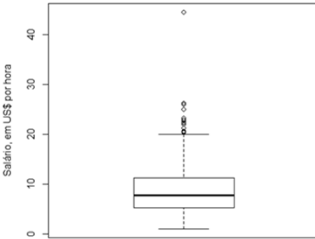
√ Outros comandos para quantis

```
> fivenum(wage)
[1] 1.00 5.25 7.78 11.25 44.50
> quantile(wage)
 0%  25%  50%  75% 100%
1.00 5.25 7.78 11.25 44.50
> quantile(wage, probs = c(0.05, 0.95))
 5%  95%
3.50 19.98
> max(wage); min(wage)
[1] 44.5
[1] 1
```

Introdução ao R com Aplicações - 2017

DA G3 • **Box-plot da variável wage:**

```
> boxplot(wage, ylab = "Salário, em US$ por hora")
```

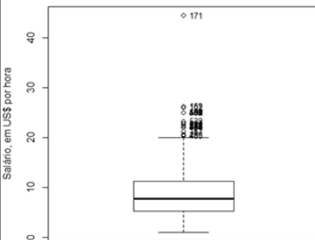


✓ Gráfico dos 5 números
✓ Há outliers

Introdução ao R com Aplicações - 2017 77

DA G3 • **Identificação de outliers em wage:**

```
> # identificação outliers
> wage.bxp <- boxplot(wage, ylab = "Salário, em US$ por hora", plot = F)
> outliers.row <- which(wage %in% wage.bxp$out)
> for(i in 1:length(wage.bxp$group)){
+   # adiciona texto no boxplot
+   text(wage.bxp$group[i], wage.bxp$out[i], which(wage==wage.bxp$out[i]),
+ pos = 4, cex = 0.8)
+ }
```



✓ Há empates

Introdução ao R com Aplicações - 2017 78

DA G3 • **Detalhamento dos outliers**

```
> outliers.row
[1] 18 20 107 157 162 169 171 178 181 185 211 410 432 434 436 450 480 485 486
[20] 495 497 503 522 532
> length(outliers.row)
[1] 24
> cps[outliers.row,]
  wage education experience age ethnicity region gender occupation
17  22.20         12         26  44      cauc  other   male   worker
19  20.55         12         33  51      cauc  other   male   worker
106 26.00         14         21  41      cauc  other   male   worker
156 24.98         16         18  40      cauc  other   male   mgmt
161 21.25         13         32  51      cauc  other   male   mgmt
sector union married
17  manufacturing yes    yes
19  other         no     yes
106 other         yes    yes
156 other         no     yes
161 other         no     no
```

Introdução ao R com Aplicações - 2017 79

DA G3 • **Medidas de dispersão de wage:**

✓ Desvio-padrão, variância e distância interquartílica

```
> sd(wage)
[1] 5.139097
> var(wage)
[1] 26.41032
> IQR(wage)
[1] 6
```

Introdução ao R com Aplicações - 2017 80

DA G3 • Histograma da variável wage:

```
> wage.hst <- hist(wage, freq = FALSE, main = "", ylab = "Densidade",
+ xlab = "Salário, em US$ por hora", ylim = c(0, 0.105))
> text(wage.hst$mid, wage.hst$density + 0.005, wage.hst$counts, cex = 0.75)
```

√ Fortemente assimétrica
√ Apenas um outlier?

Introdução ao R com Aplicações - 2017 81

DA G3 • Assimetria e curtose de wage:

```
> library(moments)
> skewness(wage)
[1] 1.692514
> kurtosis(wage)
[1] 7.933936
```

Introdução ao R com Aplicações - 2017 82

DA G3 • Histograma com suavização:

```
> # suavização por núcleo estimador
> lines(density(wage), col = 4)
```

√ Indício de bimodalidade?
– poucos dados para afirmar

Introdução ao R com Aplicações - 2017 83

DA G3 • Histograma de $\log(\text{wage})$:

```
> hist(log(wage), freq = FALSE, main = "", ylab = "Densidade",
+ xlab = "Salário, em US$ por hora")
> lines(density(log(wage)), col = 4)
```

√ Distribuição de $\log(\text{wage})$ é menos assimétrica

Introdução ao R com Aplicações - 2017 84

DA G3 • **Resumo da variável occupation:**

√ Tabelas de frequências absolutas e relativas

```
> summary(occupation)
worker techn services office sales mgmt
156    105    83    97    38    55
```

```
> tab <- table(occupation)
> prop.table(tab)
occupation
worker      techn  services   office    sales    mgmt
0.29213483 0.19662921 0.15543071 0.18164794 0.07116105 0.10299625
```

Introdução ao R com Aplicações - 2017 85

DA G3 • **Barplot da variável occupation:**

```
> nomes <- c("Indústria", "Técnico", "Serviço", "Escritório", "Vendas",
+ "Gestão")
> occup.bp <- barplot(tab, names.arg = nomes, cex.names = 0.85, las = 3,
+ ylim = c(0, 165))
> text(occup.bp, tab, labels = tab, cex = 0.85, pos = 3, offset = 0.5)
```

√ Ficaria melhor se as barras estivessem ordenadas por frequência?

Introdução ao R com Aplicações - 2017 86

DA G3 • **Barplot com as barras ordenadas:**

```
> ordem <- order(tab, decreasing = T)
> occup.ord <- barplot(tab[ordem], names.arg = nomes[ordem], cex.names = 0.85,
+ las = 3, ylim = c(0, 165))
> text(occup.ord, tab[ordem], labels = tab[ordem], cex = 0.85, pos = 3,
+ offset = 0.5)
```

√ Pode ser conveniente quando houver muitos níveis

Introdução ao R com Aplicações - 2017 87

DA G3 • **Pie chart da variável occupation:**

```
> pie(tab)
```

√ Eventualmente, podem ser úteis.

Introdução ao R com Aplicações - 2017 88

DA G3

- Análise bivariada – categóricas
(Variáveis: `gender` e `occupation`)
√ Tabela de dupla entrada

```
> xtabs(~ gender + occupation, data = cps)
      occupation
gender worker techn services office sales mgmt
male    126   53   34   21   21   34
female   30   52   49   76   17   21
```

Introdução ao R com Aplicações - 2017 89

DA G3

- *Barplot* com duas variáveis categóricas:

```
> plot(gender ~ occupation, xlab = "Ocupação", ylab = "Sexo")
```

- √ Proporção de homens e mulheres variam consideravelmente com a ocupação
- √ Há mais pessoas trabalhando em “workers” que em “sales”

Introdução ao R com Aplicações - 2017 90

DA G3

- Análise bivariada – quantitativas
(Variáveis: `log(wage)` e `education`)
√ Correlação de Pearson

```
> cor(wage, education)
[1] 0.3819221
> cor(log(wage), education)
[1] 0.3803983
```

- √ Correlação de Spearman – Não Paramétrico

```
> cor(wage, education, method = "spearman")
[1] 0.3813425
> cor(log(wage), education, method = "spearman")
[1] 0.3813425
```

Introdução ao R com Aplicações - 2017 91

DA G3

- *Plot* para duas variáveis quantitativas:

```
> plot(log(wage) ~ education, xlab = "Escolaridade, em anos")
```

- √ Difícil perceber tendência – Correlação linear baixa

Introdução ao R com Aplicações - 2017 92

DA G3 • *Plot com suavizador:*

```
> plot(log(wage) ~ education, xlab = "Escolaridade, em anos")
> wage.lo <- loess(log(wage) ~ education)
> lines(sort(of.lo$x), sort(of.lo$fit))
```

√ Correlação baixa entre as variáveis

Introdução ao R com Aplicações - 2017 93

DA G3 • *Bivariada – quantitativa vs. categórica:*
(Variáveis: $\log(\text{wage})$ e gender)

√ Estatísticas descritivas por estrato

```
> # média por estrato
> tapply(log(wage), gender, mean)
male female
2.165286 1.934037
> # média e desvio padrão por estrato
> aggregate(log(wage) ~ gender, FUN = function(x) c(M=mean(x), SD=sd(x)))
gender log(wage).M log(wage).SD
1 male 2.165286 0.534453
2 female 1.934037 0.492118
```

√ Divisão da variável wage por gender:

```
> cor(wage, education, method = "spearman")
[1] 0.3813425
> cor(log(wage), education, method = "spearman")
[1] 0.3813425
```

Introdução ao R com Aplicações - 2017 94

DA G3 • *Boxlot wage por gender:*

```
> plot(log(wage) ~ gender, xlab = "Sexo", xaxt = "n")
> axis(1, at = 1:2, labels = c("Masculino", "Feminino"))
```

√ Similares as formas gerais de ambas as distribuições
√ Homens levam vantagem, principalmente na 'faixa média'

Introdução ao R com Aplicações - 2017 95

DA G3 • *qq-plot de wage por gender:*

```
> plot(log(wage) ~ gender, xlab = "Sexo", xaxt = "n")
> axis(1, at = 1:2, labels = c("Masculino", "Feminino"))
```

√ Na maioria dos quantis, salário dos homens é tipicamente mais alto

Introdução ao R com Aplicações - 2017 96

DA G3 • Gráfico de probabilidade normal:

```

> qqnorm(mwage, ylab = "Log(wage)", xlab = "Escores normais",
+ main="Gráfico de probabilidade \nNormal")
> qqline(mwage)
> norm.fem <- qqnorm(fwage, plot.it = F)
> points(norm.fem$x, norm.fem$y, pch = 21, col = "blue", bg = "blue")
> qqline(fwage, lty = 2, col = "blue")
    
```

Gráfico de probabilidade Normal

√ Distribuições similares com valores menores de quantis para Mulheres

Introdução ao R com Aplicações - 2017

Gráficos de Probabilidade

DA G3 • Como saber se uma distribuição de probabilidades é um modelo razoável para os dados?

- √ Pode-se fazer uma verificação de suposições:
 - Forma da distribuição, frequência esperada das observações
- Verificação gráfica:
 - √ Histogramas
 - Dão uma ideia da forma da distribuição,
 - Em geral não são indicadores confiáveis (a menos que o tamanho amostral seja grande)

Estadística Aplicada à Engenharia

DA G3 • Gráfico de probabilidades:

- √ Procedimento geral é simples
- √ Mais confiável que histograma para tamanhos amostrais pequenos ou moderados
- √ Usa eixos especiais, projetados para a distribuição hipotética

Estadística Aplicada à Engenharia

DA G3 • Procedimento:

- √ Ordenação das observações amostrais:
 - $x_{(1)}, x_{(2)}, \dots, x_{(n)}$
- √ Plotam-se os pontos $(x_{(j)}, (j - 0,5)/n)$
(observação, frequência acumulada observação)
- √ Usa-se uma escala de probabilidade
- √ Distribuição descreve adequadamente os dados:
 - pontos cairão, aproximadamente, ao longo de uma linha reta
- √ Modelo hipotético não é apropriado
 - os pontos desviam-se significativamente de uma linha reta

101

Estatística Aplicada à Engenharia

DA G3 • É subjetivo determinar se os pontos seguem ou não uma linha reta!

102

Introdução ao R com Aplicações - 2017

DA G3 **Gráfico de Probabilidades Normal**

- Pode ser útil na identificação de distribuições que sejam simétricas mas que tenham caudas mais pesadas (ou mais leves) que a normal

103

Introdução ao R com Aplicações - 2017

DA G3 • Distribuição de cauda leve

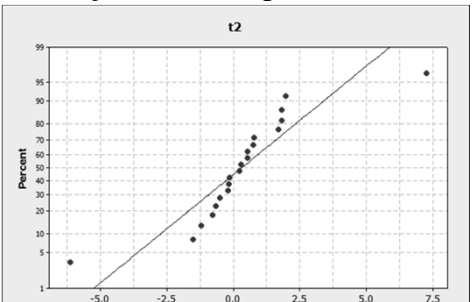


- √ Pontos à esquerda tendem a ficar abaixo da linha e à direita tendem a ficar acima
 - As menores e maiores observações não serão tão extremas como se esperaria de uma normal

104

Introdução ao R com Aplicações - 2017

DA G3 • Distribuição de cauda pesada

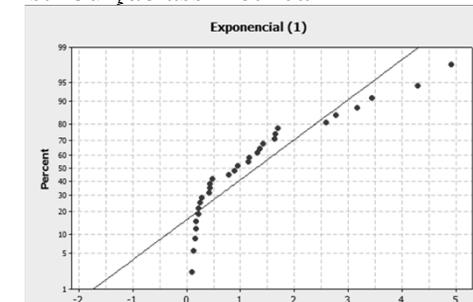


✓ Pontos à esquerda tendem a ficar acima da linha e à direita tendem a ficar abaixo

✓ Gráfico em forma de S

Introdução ao R com Aplicações - 2017 105

DA G3 • Distribuição assimétrica



✓ Pontos de ambas as extremidades tendem a estar abaixo da linha

✓ Gráfico tem forma curvada

Introdução ao R com Aplicações - 2017 106

Referências

DA G3

Bibliografia Recomendada

- ALBERT, J.; RIZZO, M. *R by Example*. Springer, 2012.
- CHAPMAN, C.; FEIT, E. M. *R for marketing research and analytics*. Springer, 2015.
- KLEIBER, C.; ZEILEIS, A. *Applied econometrics with R*. Springer, 2008.
- DALGAARD, P. *Introductory statistics with R*. Springer, 2008.

Introdução ao R com Aplicações - 2017 108