

Laboratório de Estatística

Lupércio França Bessegato
Dep. de Estatística/UFJF

Visualização e Descrição de Dados em R



Roteiro Geral



1. Fundamentos da linguagem R
2. Visualização e descrição de dados em R
3. Probabilidade em R
4. Tópicos especiais
5. Referências

Laboratório de Estatística - 2020

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

Laboratório de Estatística - 2020

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

Laboratório de Estatística - 2020 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
```

Laboratório de Estatística - 2020 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
```

Laboratório de Estatística - 2020 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)
```

Laboratório de Estatística - 2020 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
```

Laboratório de Estatística - 2020

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)
```

Laboratório de Estatística - 2020

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
```

Laboratório de Estatística - 2020

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
```

Laboratório de Estatística - 2020

13

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
```

14

Laboratório de Estatística - 2020

Resumo dos Dados

- Resumo de 5 números e média
 - ✓ `summary(x)`: fornece o mínimo, 1º quartil, Mediana, 3º quartil, máximo e média dos elementos de x.
 - ✓ Resumo apenas para variáveis quantitativas
 - Fatores: contagem de níveis

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

15

Laboratório de Estatística - 2020

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
```

16

Laboratório de Estatística - 2020

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 frequencia de erupções em cada sub-intervalo
duracao.freq <- table(duracao.cut)
# tabela com os resultados
cbind(duracao.freq)
```

17

Laboratório de Estatística - 2020

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$breaks
# frequência de valores em cada ub-intervalo
duracao.hist$counts
```

18

Laboratório de Estatística - 2020

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
```

19

Laboratório de Estatística - 2020

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
```

- Opcões 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')`

20

Laboratório de Estatística - 2020

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

Laboratório de Estatística - 2020

Gráfico Gerado

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

22

Laboratório de Estatística - 2020

Adição de Dados

- 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 $y=a+bx$

23

Laboratório de Estatística - 2020

Coordenadas das Margens

- 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)
```

24

Laboratório de Estatística - 2020

Gráfico c/ Coordenadas Margens

- Layout de um gráfico padrão

25

Laboratório de Estatística - 2020



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

Laboratório de Estatística - 2020

26



Montagem do Gráfico



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

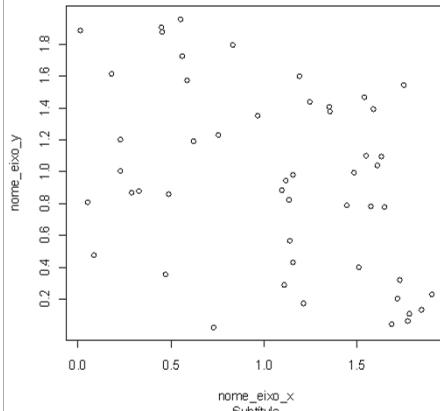
```
points(x, y) # plota os pontos do gráfico
axis(1) # plota o eixo x
axis(2, at = seq(0.2, 1.8, 0.2)) # plota o eixo y
box() # caixa do gráfico
# Título, sub-título, nomes dos eixos
title(main = "Título Principal", sub = "Subtítulo",
      xlab = "nome_eixo_x", ylab = "nome_eixo_y")
```

Laboratório de Estatística - 2020

27



Título Principal



Laboratório de Estatística - 2020

28

Histogramas



Construção de Histograma



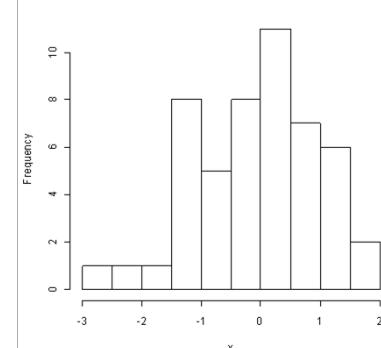
- Geração de uma amostra aleatória com distribuição de freqüências com simetria:
`x <- rnorm(50)`
- Construção do histograma (default)
`hist(x)`

Laboratório de Estatística - 2020

30



Histogram of x



✓ Repetir com outras amostras e verifique

Laboratório de Estatística - 2020

31

Visualização e Descrição de Dados – Exemplos



Exemplo



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

Laboratório de Estatística - 2020

35



✓ Variáveis:

- title: título do periódico
- publisher: nome do editor (fator com 52 níveis)
- society: periódico é publicado por uma sociedade acadêmica? ('no' = não, 'yes' = sim)
- price: preço da assinatura da biblioteca
- pages: número de páginas
- charpp: caracteres por página
- citations: número total de citações
- foundingyear: ano de fundação do jornal
- subs: número de assinaturas da biblioteca
- field: descrição do campo da Economia (fator, com 24 níveis).



Laboratório de Estatística - 2020

36



• 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",...
 18 18 13 18 28 11 ...
 $ society     : Factor w/ 2 levels "no","yes": 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
```

37

Laboratório de Estatística - 2020



• 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",...
 166 45 144 ...
 $ title      : Factor w/ 180 levels "Agricultural Economics",...
 137 47 168 43 143 ...
 $ publisher   : Factor w/ 52 levels "Academic Press",...
 28 11 ...
 $ society     : Factor w/ 2 levels "no","yes": 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 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
```

Laboratório de Estatística - 2020

38



• 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
```

39

Laboratório de Estatística - 2020

• Preparação do banco

✓ Preço unitário por citação

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

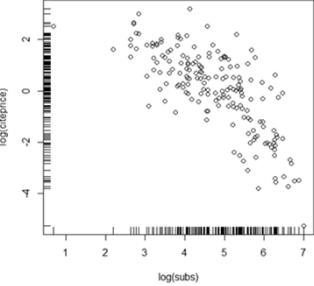
40

Laboratório de Estatística - 2020

• Plot com distribuições marginais:

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

✓ Comando rug: marcas para cada dado



41

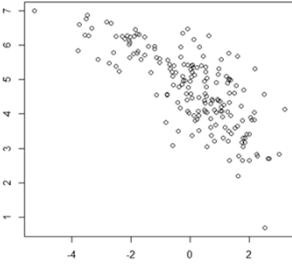
Laboratório de Estatística - 2020

• Plot – comando alternativo

✓ Sem o arquivo estar anexado

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

✓ Formato Y ~ X



42

Laboratório de Estatística - 2020

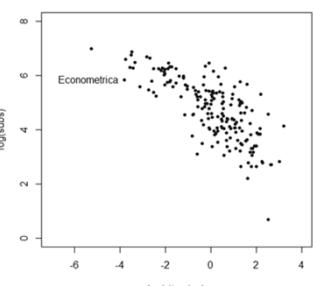
• Plot – parâmetros gráficos:

```
> 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



43

Laboratório de Estatística - 2020

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:

44

Laboratório de Estatística - 2020

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

45

Laboratório de Estatística - 2020

Sugestão

- Experimentar o comando:
✓ `demo("plotmath")`

46

Laboratório de Estatística - 2020

Família apply

11

 **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,] 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

48

Laboratório de Estatística - 2020

 • 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
> lapply(dados, FUN = median)
$x
[1] 3
$y
[1] 8
$z
[1] 13
```

49

Laboratório de Estatística - 2020

 • 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
```

50

Laboratório de Estatística - 2020

 • 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

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

51

Laboratório de Estatística - 2020

 • 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
```

Laboratório de Estatística - 2020



52

 • 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
```

Laboratório de Estatística - 2020



53

 • 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
```

Laboratório de Estatística - 2020



54

 • 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
```

Laboratório de Estatística - 2020



55



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}*

Laboratório de Estatística - 2020

57



✓ 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)

Laboratório de Estatística - 2020

58



• Importação dos dados:



```
> # carregamento dos dados
> dados <- ChickWeight
> help(ChickWeight)
> str(dados)
Classes 'nfnGroupedData', 'nfGroupedData', 'groupedData' and 'data.frame':
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 ...
>
> head(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
```

Laboratório de Estatística - 2020

59



• 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
```

Laboratório de Estatística - 2020

60

• 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?

61

Laboratório de Estatística - 2020

• 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
```

62

Laboratório de Estatística - 2020

• 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
```

63

Laboratório de Estatística - 2020

• 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

64

Laboratório de Estatística - 2020

Referências

Bibliografia Recomendada



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