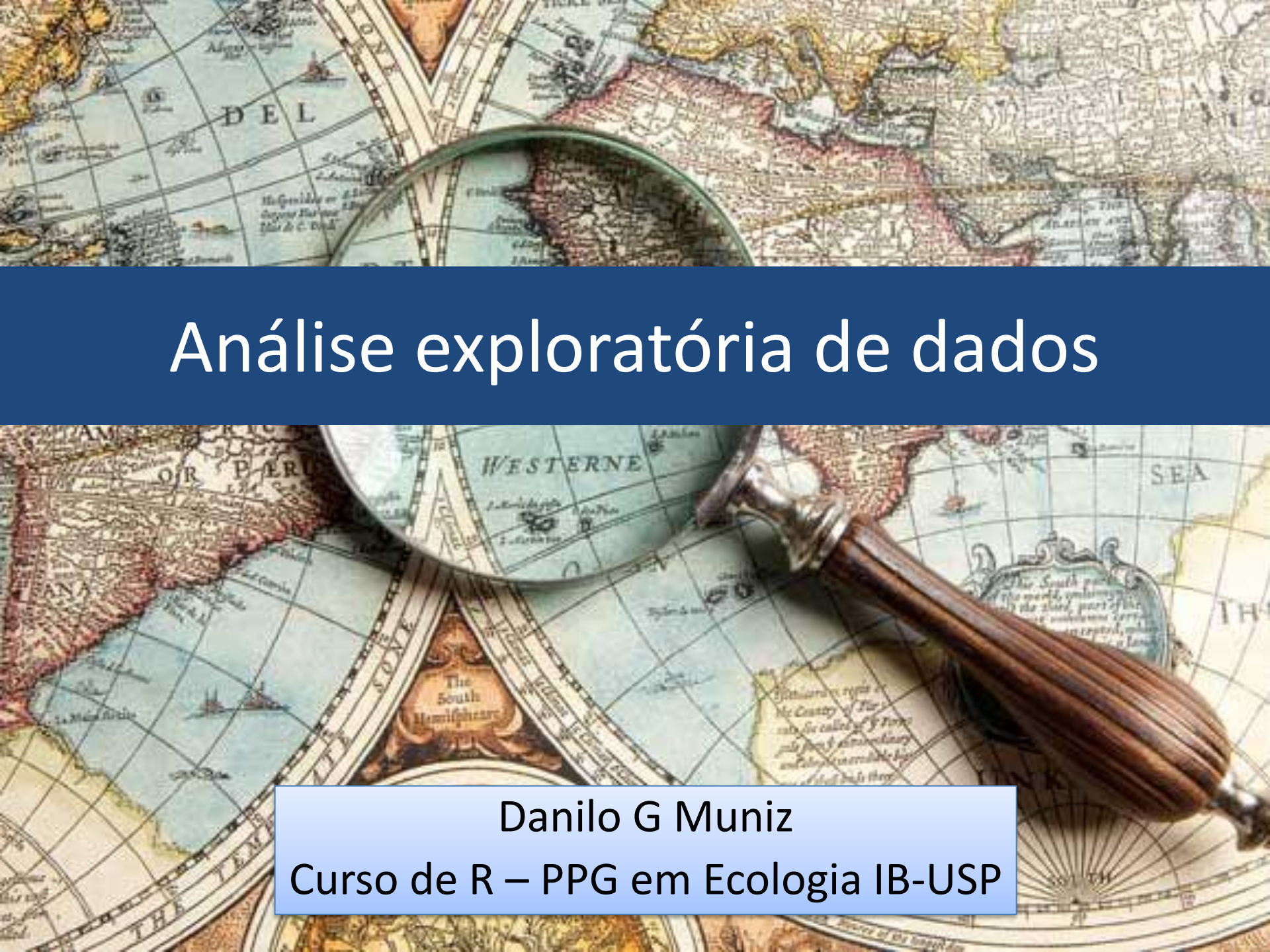


Análise exploratória de dados

Danilo G Muniz

Curso de R – PPG em Ecologia IB-USP

Análise Exploratória de dados

Estatística é muito mais do que fazer testes!

"procedures for analyzing data, techniques for interpreting the results of such procedures, way of planning the gathering of data, and all the machinery and results of (mathematical) statistics which apply to analyzing data"

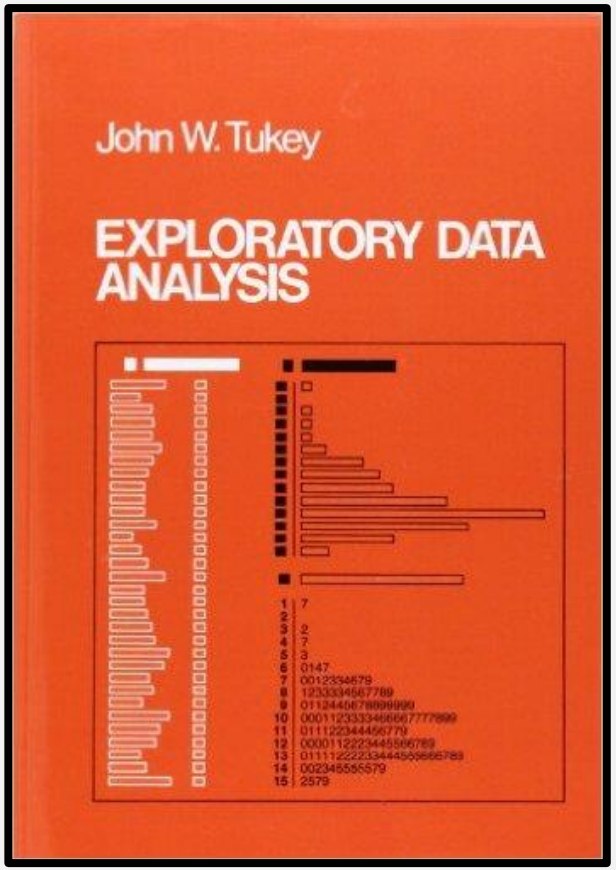
John W Tukey

"Procedimentos para analisar dados, técnicas para interpretar os resultados de tais procedimentos, modo de planejamento da coleta de dados e toda a maquinaria e resultados de estatísticas (matemáticas) que se aplicam ao analisar dados"

Análise Exploratória de dados

Estatística é muito mais do que fazer testes estatísticos!

John W Tukey



O que é análise exploratória?

Do inglês “exploratory data analysis”,
do espanhol “análisis exploratorio de datos”,
do polonês “badania eksploracyjne”...

“Uma abordagem de análise de dados com o objetivo de resumir as principais características de um conjunto de dados, comumente usando gráficos”

Adaptado da Wikipédia (fazer o que né...)

O que é análise exploratória?

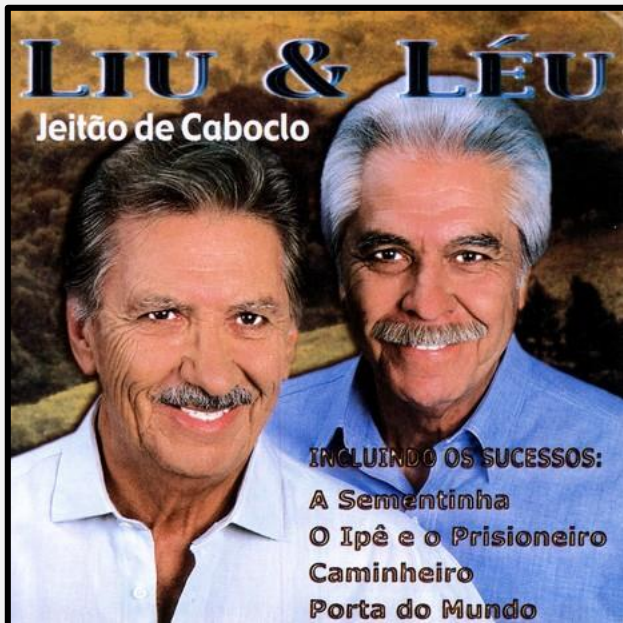
"Análise exploratória de dados nunca pode ser a história completa, mas nada mais serve como a pedra fundamental como o primeiro passo. “

"Análise exploratória é trabalho de detetive, trabalho numérico de detetive.“

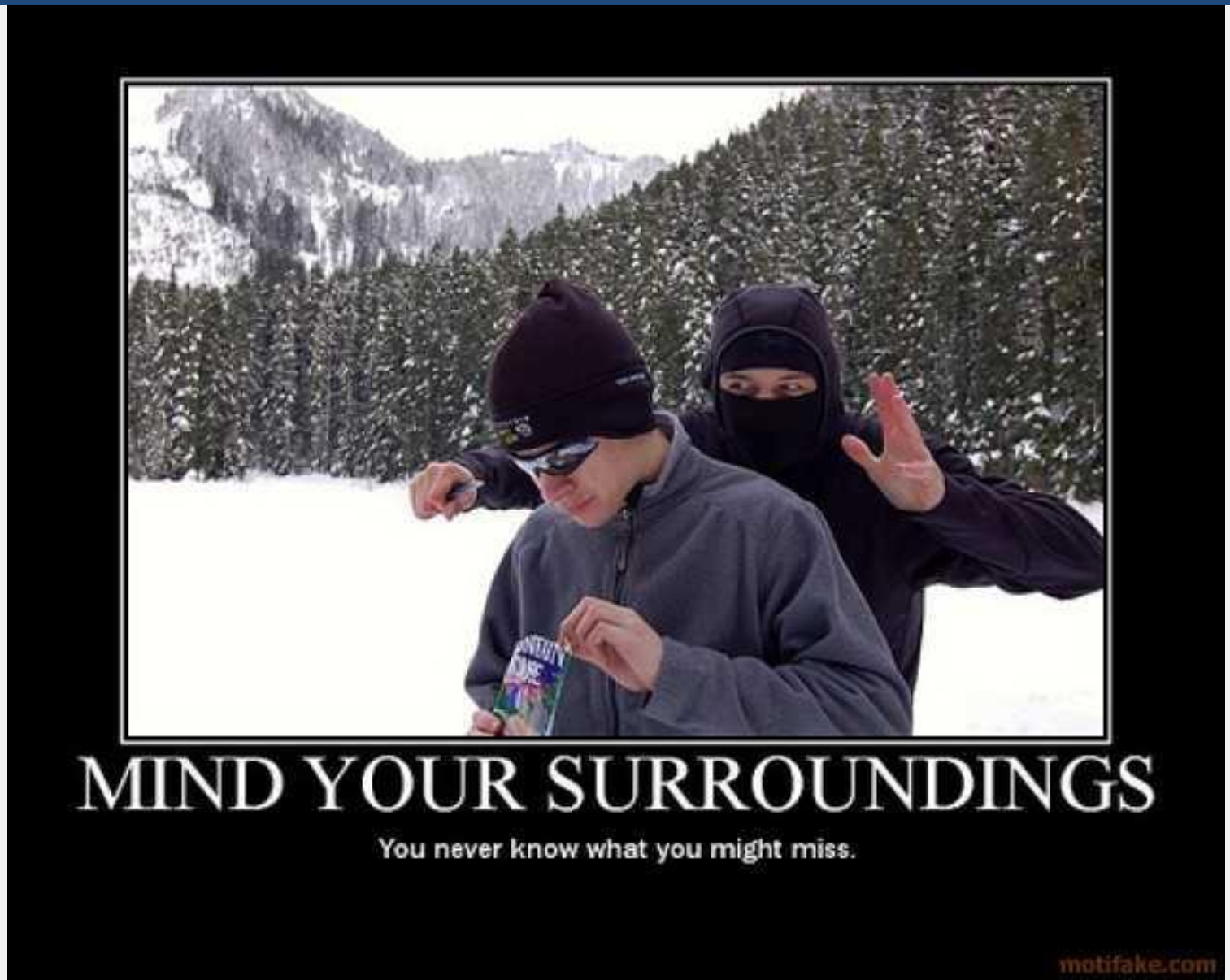
“Análise exploratória é olhar para os dados e ver o que eles dizem”.

Objetivos da análise exploratória

- Controle de qualidade dos dados
- Descobrir padrões e formular hipóteses (para estudos futuros)
- Avaliar premissas dos testes estatísticos planejados
- “Sentir o jeitão” dos dados



Conheça bem seus dados!



Uma anedota do tempo da guerra

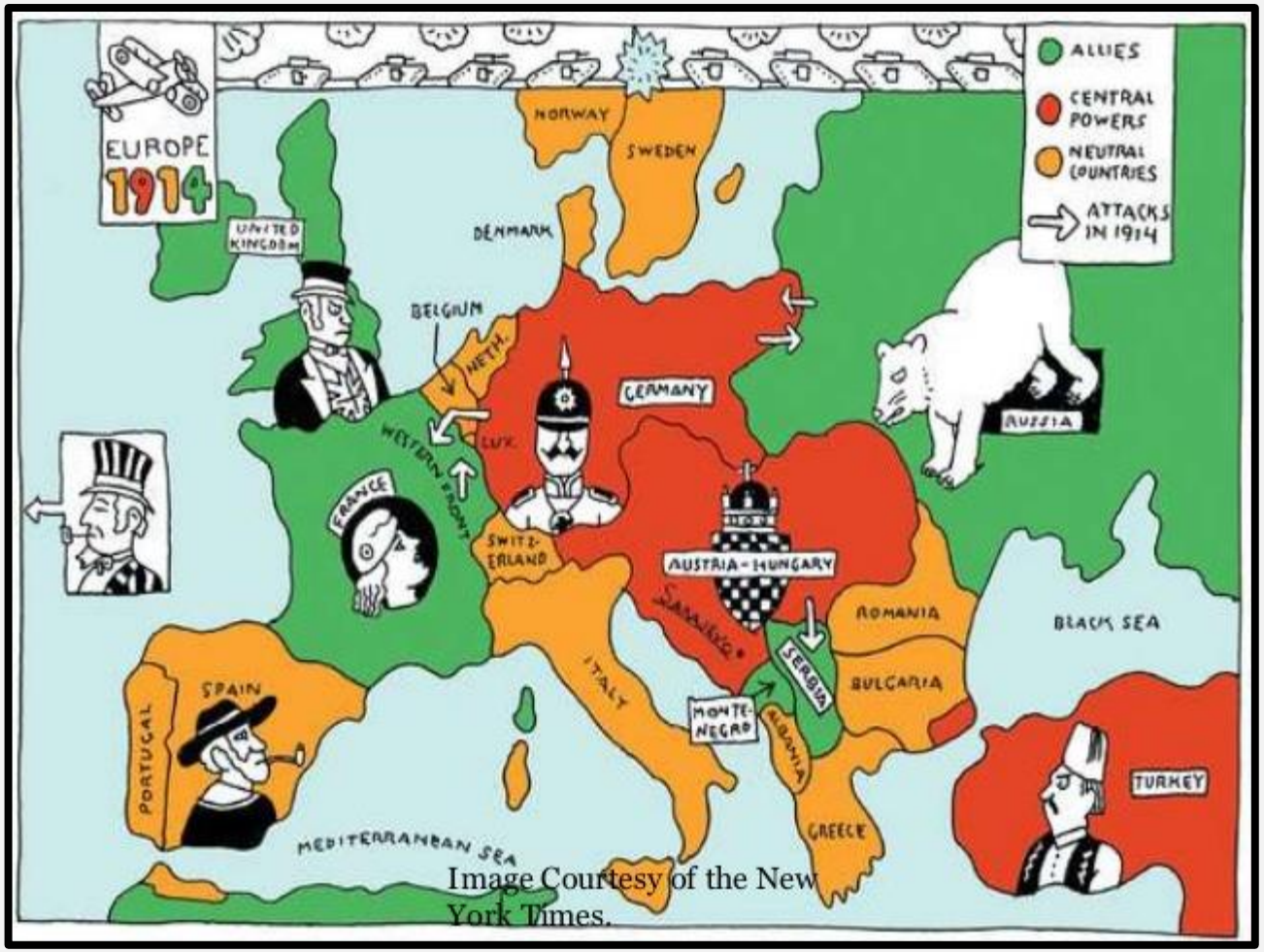
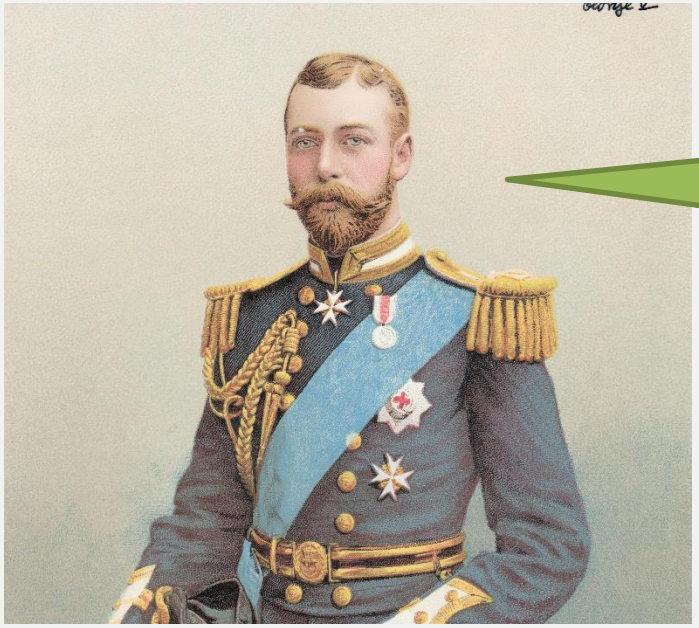


Image Courtesy of the New York Times.

Capacetes britânicos na 1ª Guerra

Depois da adoção do capacete

Aumento gigantesco no número de ferimentos na cabeça!

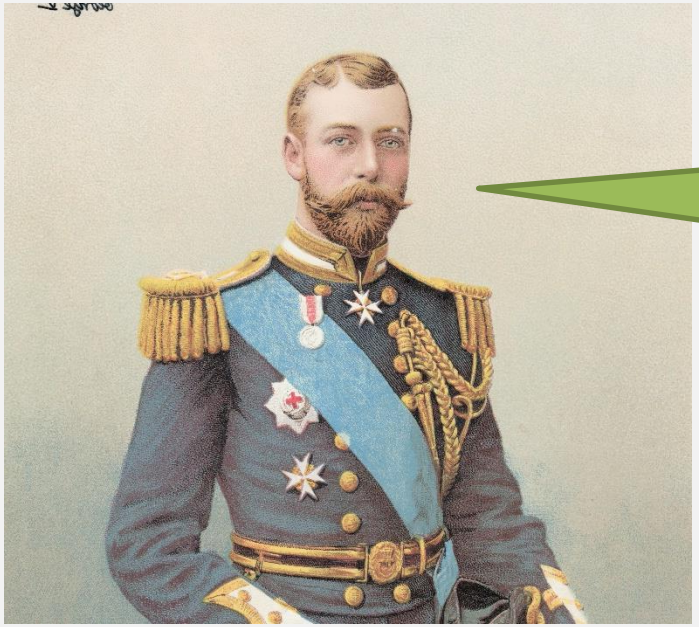


Ora, mas qual a explicação para tão inesperada ocorrência?

Olhando melhor os dados

- O número de FERIMENTOS aumentou

- Mas o número de MORTES caiu!



Ah, sim! Comprem mais capacetes!

Conheça bem os seus dados!

Conheça bem seus dados

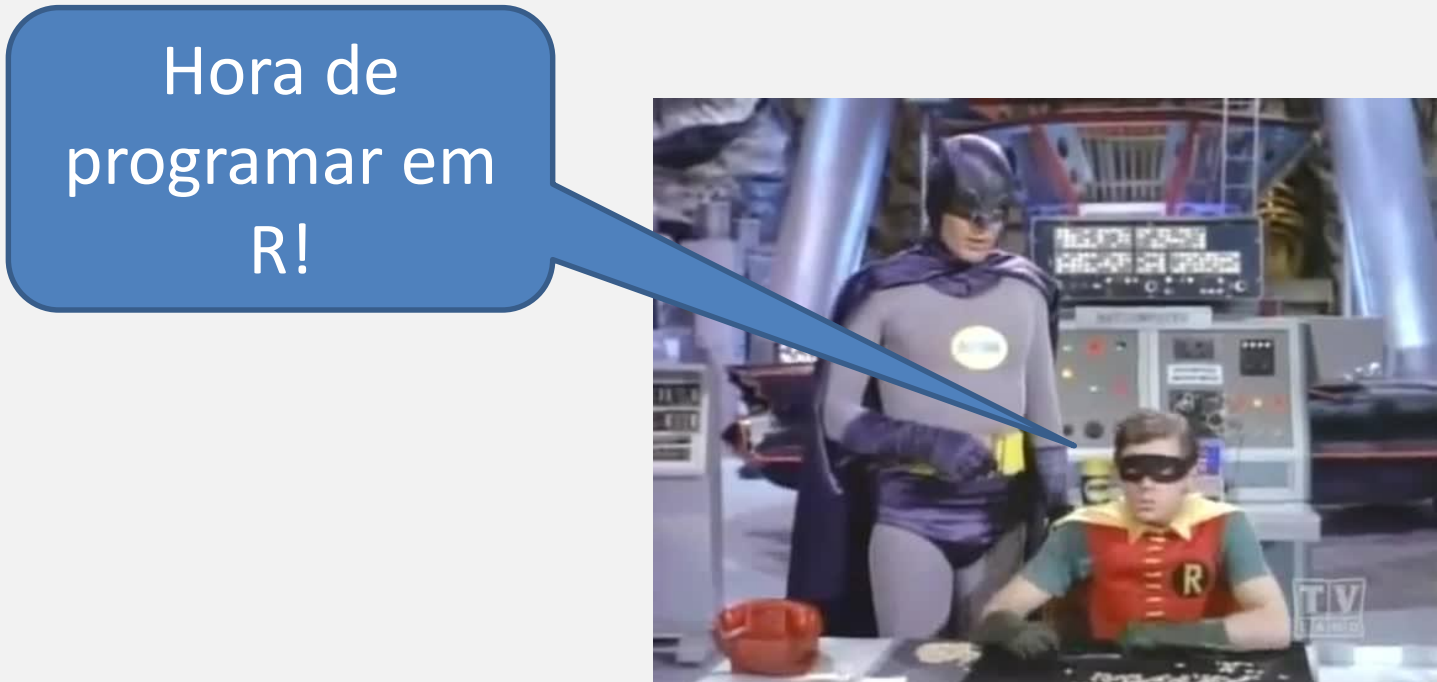
Leve-os pra passear, ouça o que eles tem a dizer.

Primeira coisa: Dê aquela conferida inicial

Dê aquela conferida inicial

Depois da leitura de dados:

funções str(), head() e tail()



Seguindo a primeira conferida

- Existem valores faltantes (NA) ?
  - São NAs mesmo ou são zeros?

- Existem muitos zeros?
  - Especialmente importante para levantamentos

Eliminando NAs

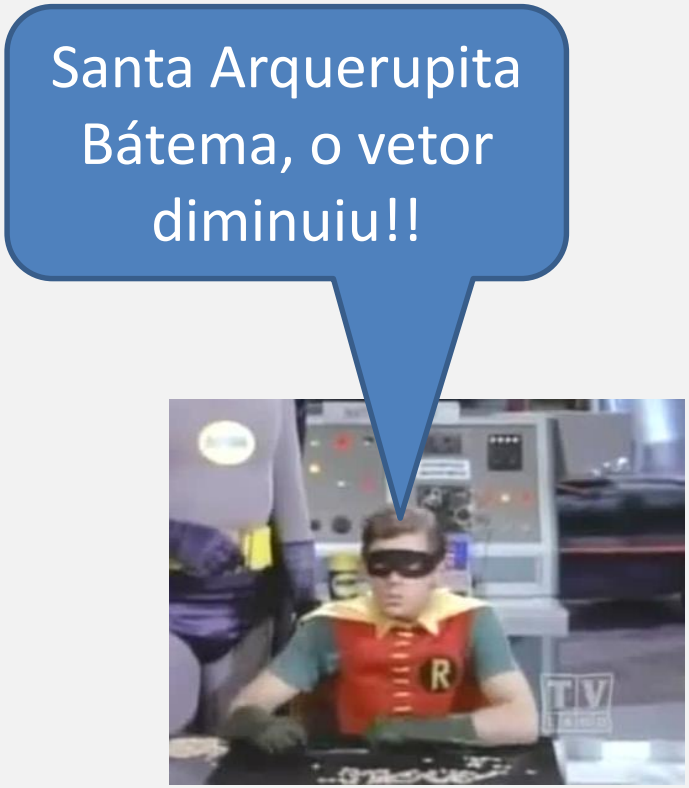
Função is.na()

```
> x = c(12, 14, 15, 76, NA, NA, 0, 9, 7, 7)
>
> which(is.na(x))
[1] 5 6
>
> x[is.na(x)] = 0
> x
 [1] 12 14 15 76  0  0  0  9  7  7
>
```

Eliminando NAs

Função na.omit()

```
> x = c(12, 14, 15, 76, NA, NA, 0, 9, 7, 7)
> x = na.omit(x)
> x
[1] 12 14 15 76  0  9  7  7
attr(,"na.action")
[1] 5 6
attr(,"class")
[1] "omit"
>
```



Contando zeros

O poder dos testes lógicos!

```
> x = c(12, 14, 15, 76,  0,  0,  0,  9,  7,  7)
> x == 0
 [1] FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE
FALSE FALSE FALSE
> sum(x == 0)
[1] 3
> table(x == 0)

FALSE  TRUE
    7     3
```

Explorando dados quantitativos

- Estatísticas descritivas básicas
  - Tendência central
    - mean(), median()
  - Variação
    - sd(), var(), range()

Vamos explorar alguns dados simulados

Usando as funções com início *r*

azul = [redacted]

laranja = [redacted]

verde = [redacted]

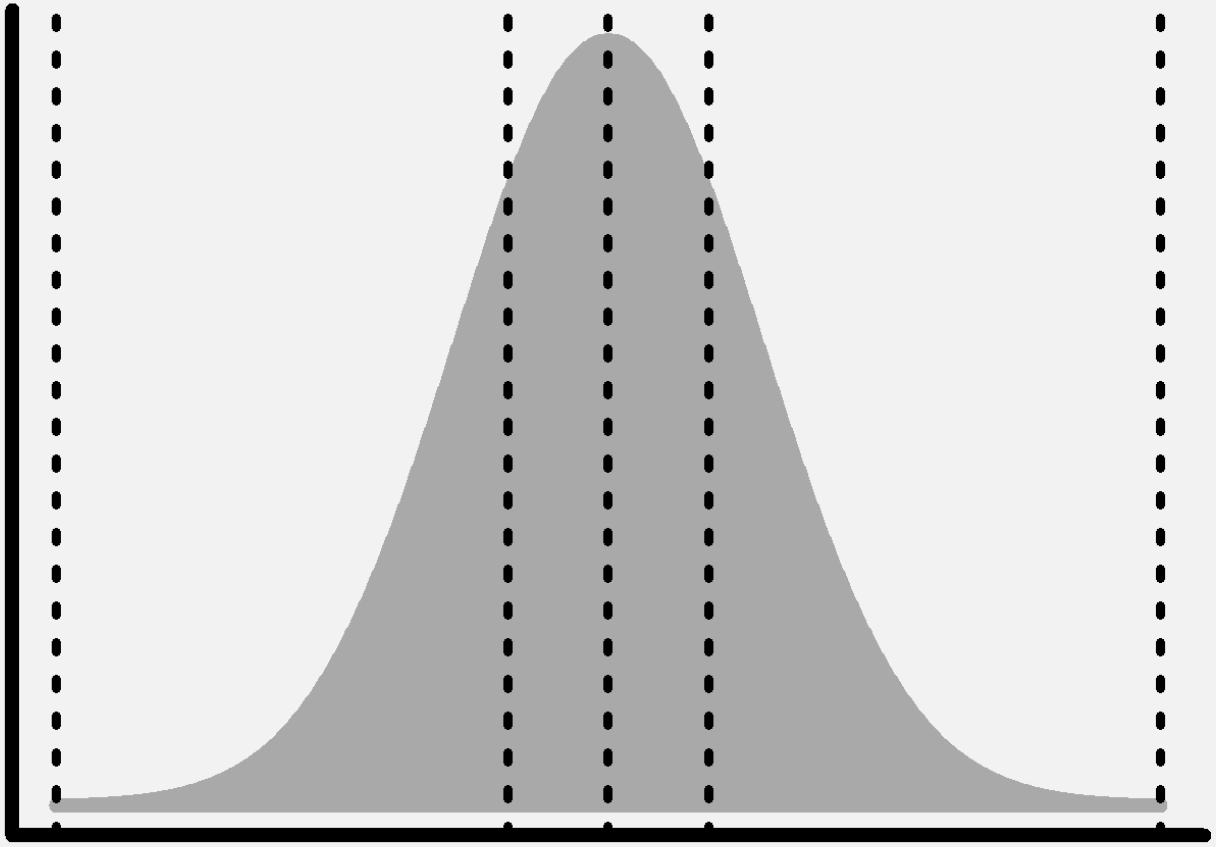
roxo = [redacted]

Tendência central

```
> mean(azul)
[1] 9.981562
> mean(laranja)
[1] 10.7297
> mean(verde)
[1] 10.16197
> mean(roxo)
[1] 9.594
>
```

O resumo de cinco números

Imagine dividir os dados em quatro fatias do mesmo tamanho



O resumo de cinco números

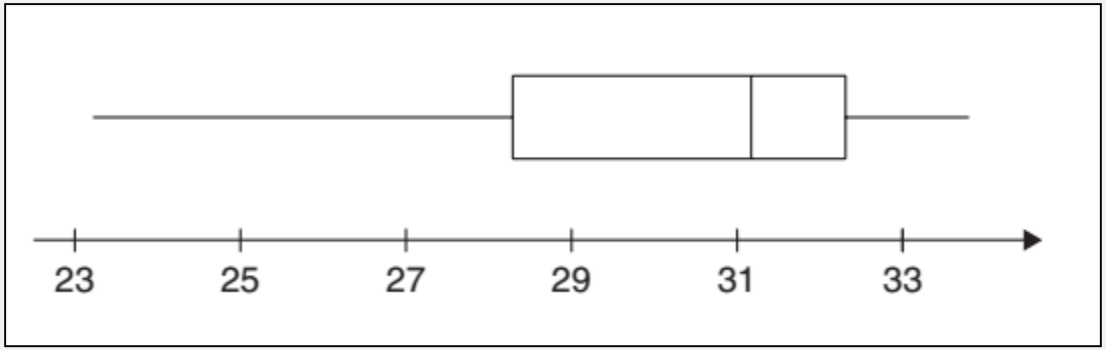
Imagine dividir os dados em quatro fatias do mesmo tamanho

11	13	7	11	9	10	10	7	11	9	9	11	5

5	7	7	9	9	9	10	10	11	11	11	11	13

O resumo de cinco números

Mínimo, 1º quartil, Mediana (2º quartil), 3º quartil, máximo



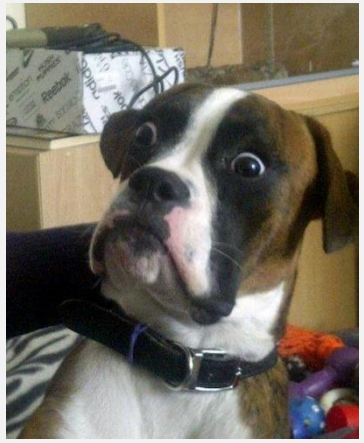
Função summary()

Morgenthaler 2009

Cinco pontos!

Quem vê tendência central, não vê distribuição!

```
> summary(azul)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  4.113   8.421  10.072   9.982  11.454  18.516
> summary(laranja)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.376   7.070   9.572  10.730  12.734  56.405
> summary(verde)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0162  3.1925  7.1034 10.1620 13.9796 82.1361
> summary(roxo)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  5.023   7.677   8.874   9.594  11.747  15.298
> |
```

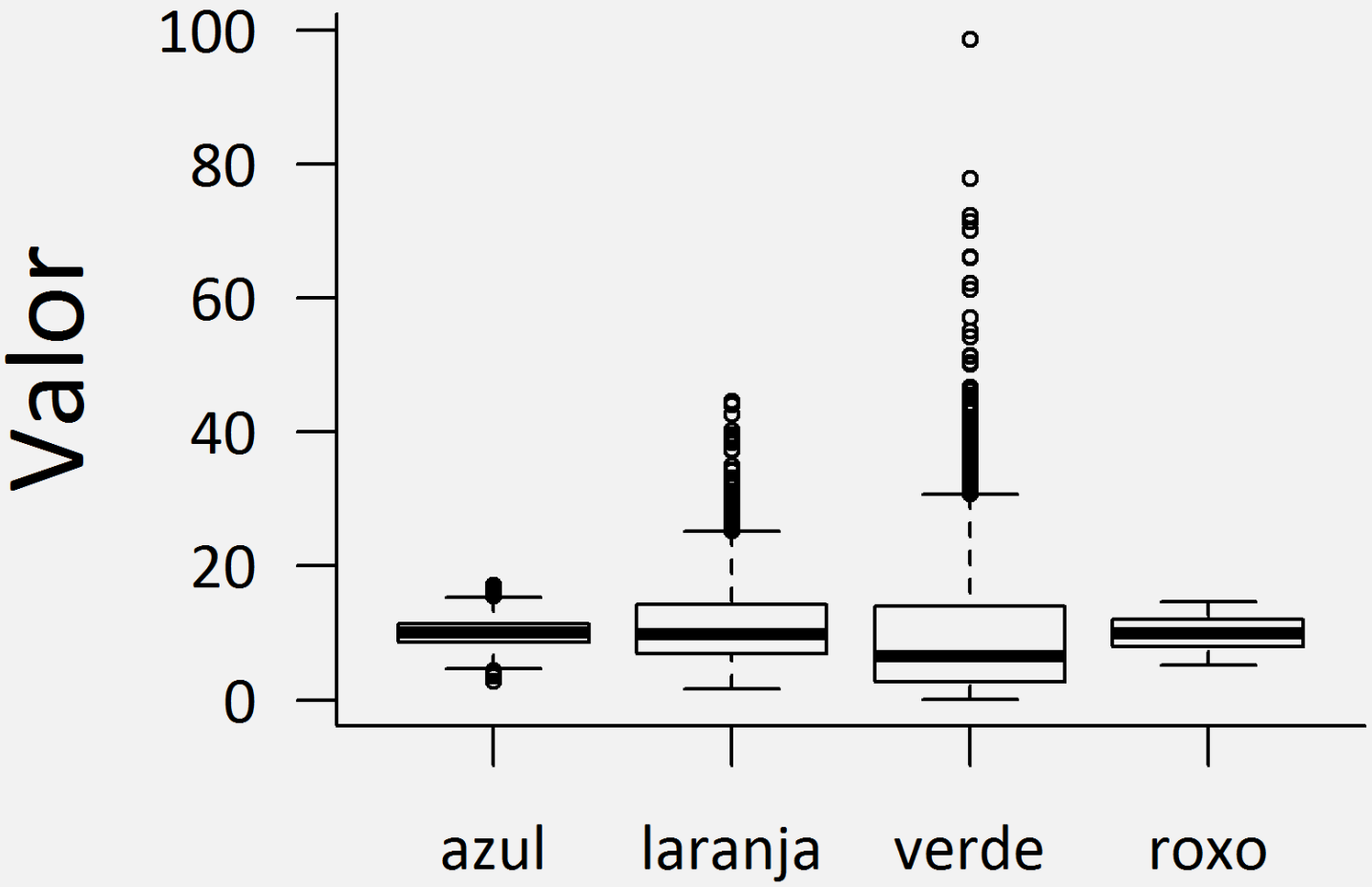


Tem como fazer um desenho?

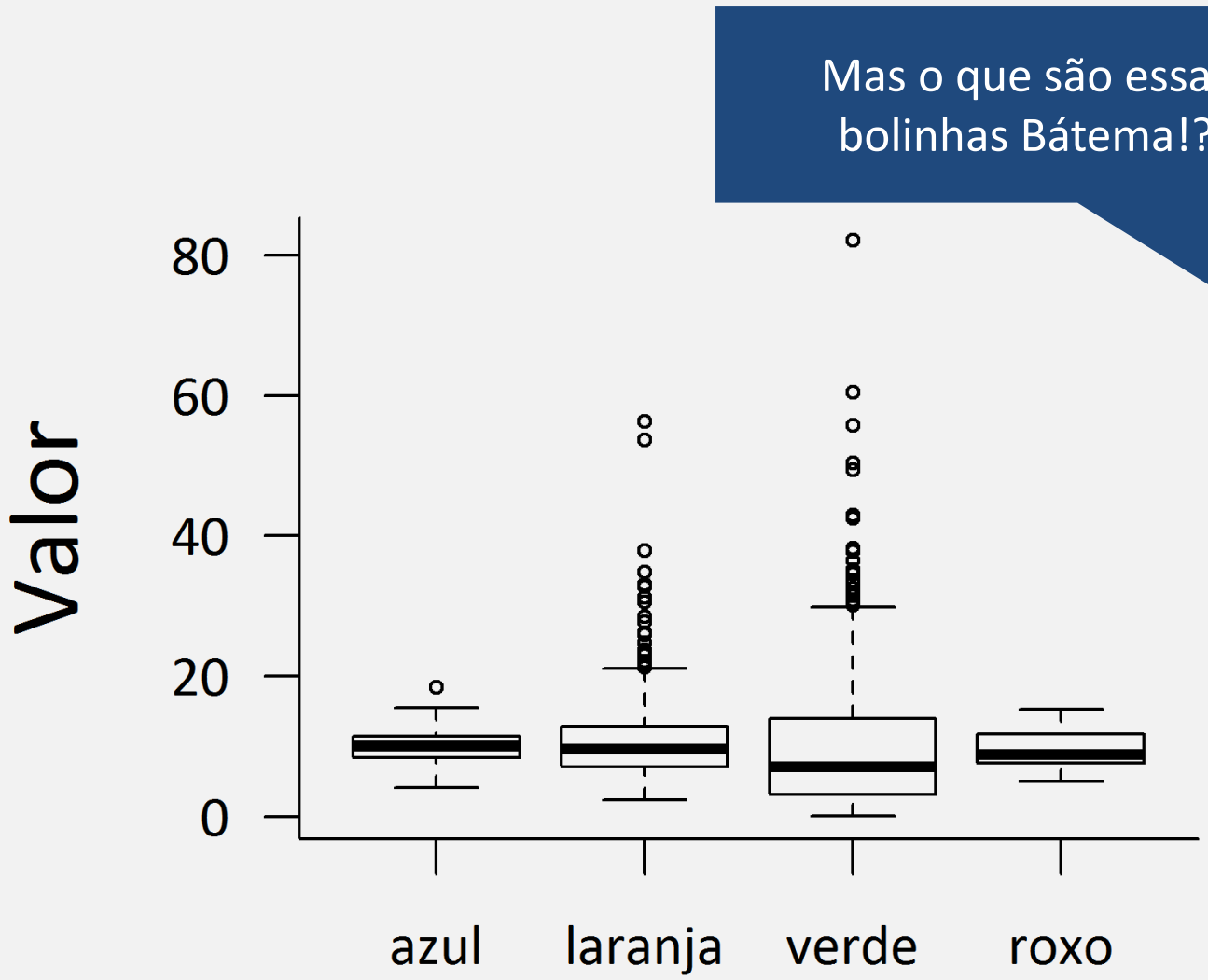
Claro, no R é tranquilo!

Explorando gráficos univariados

O famoso boxplot!

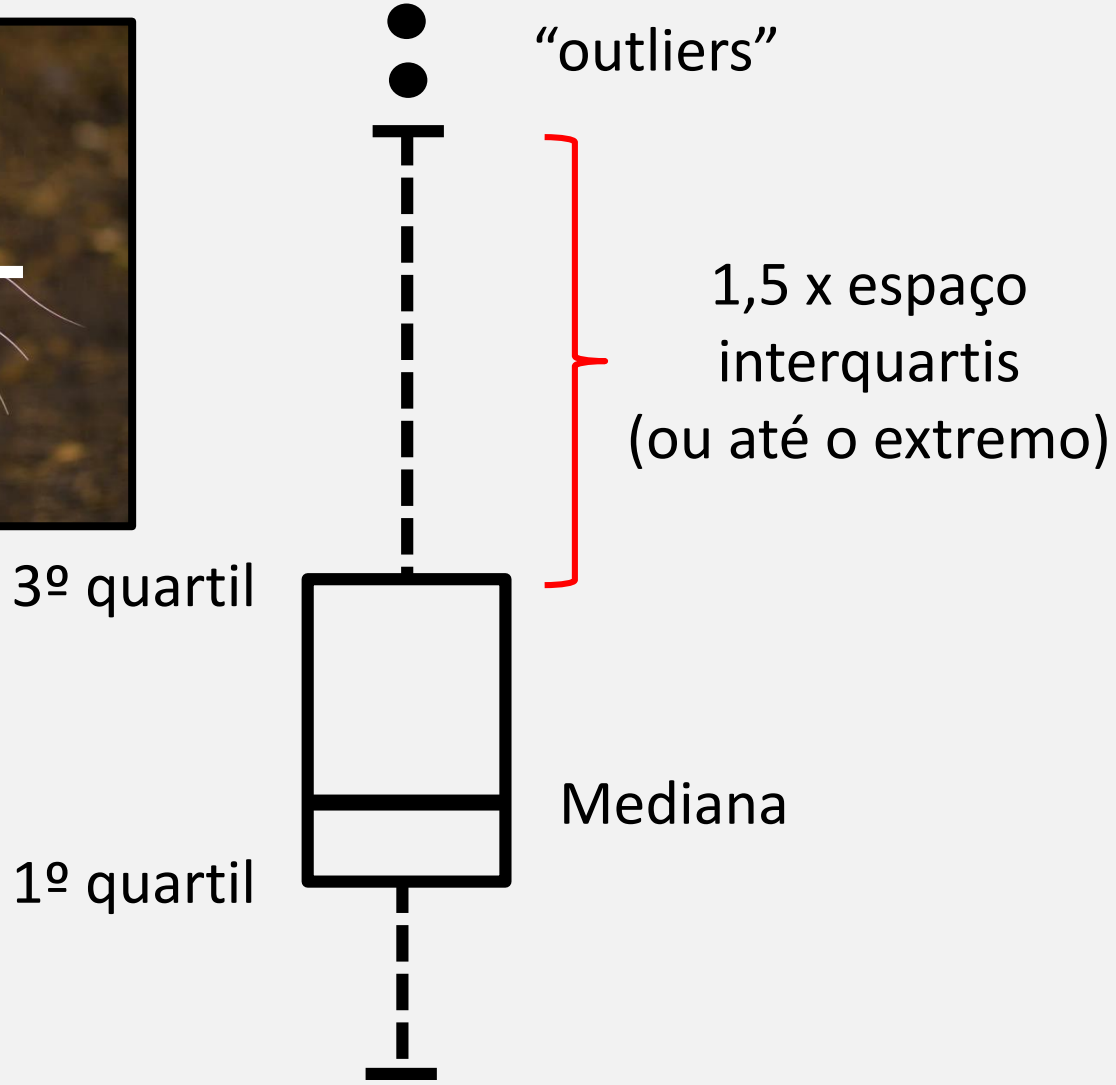


O famoso boxplot!



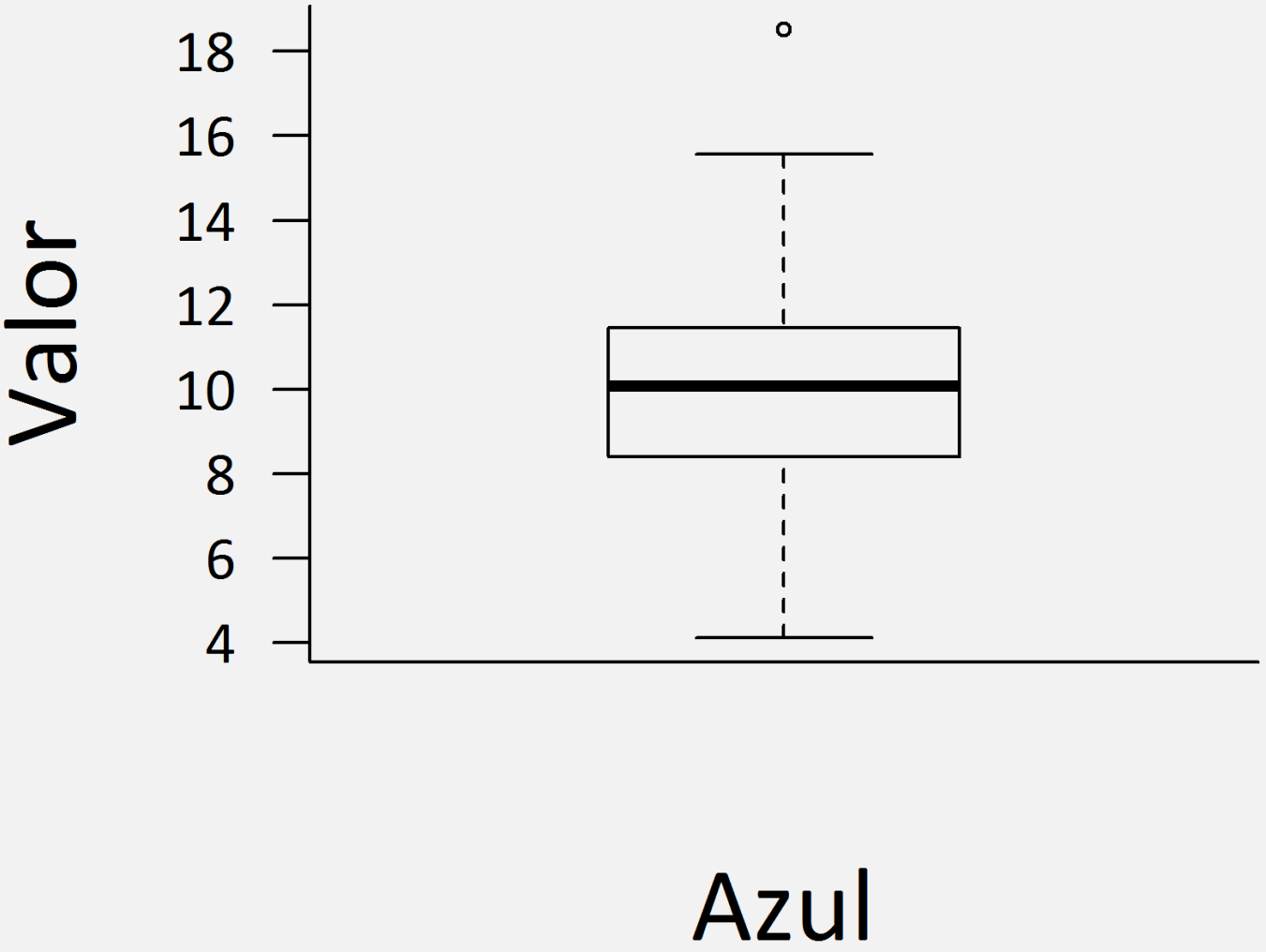
Mas o que são essas bolinhas Bátema!?

Entendendo a caixa e os bigodes

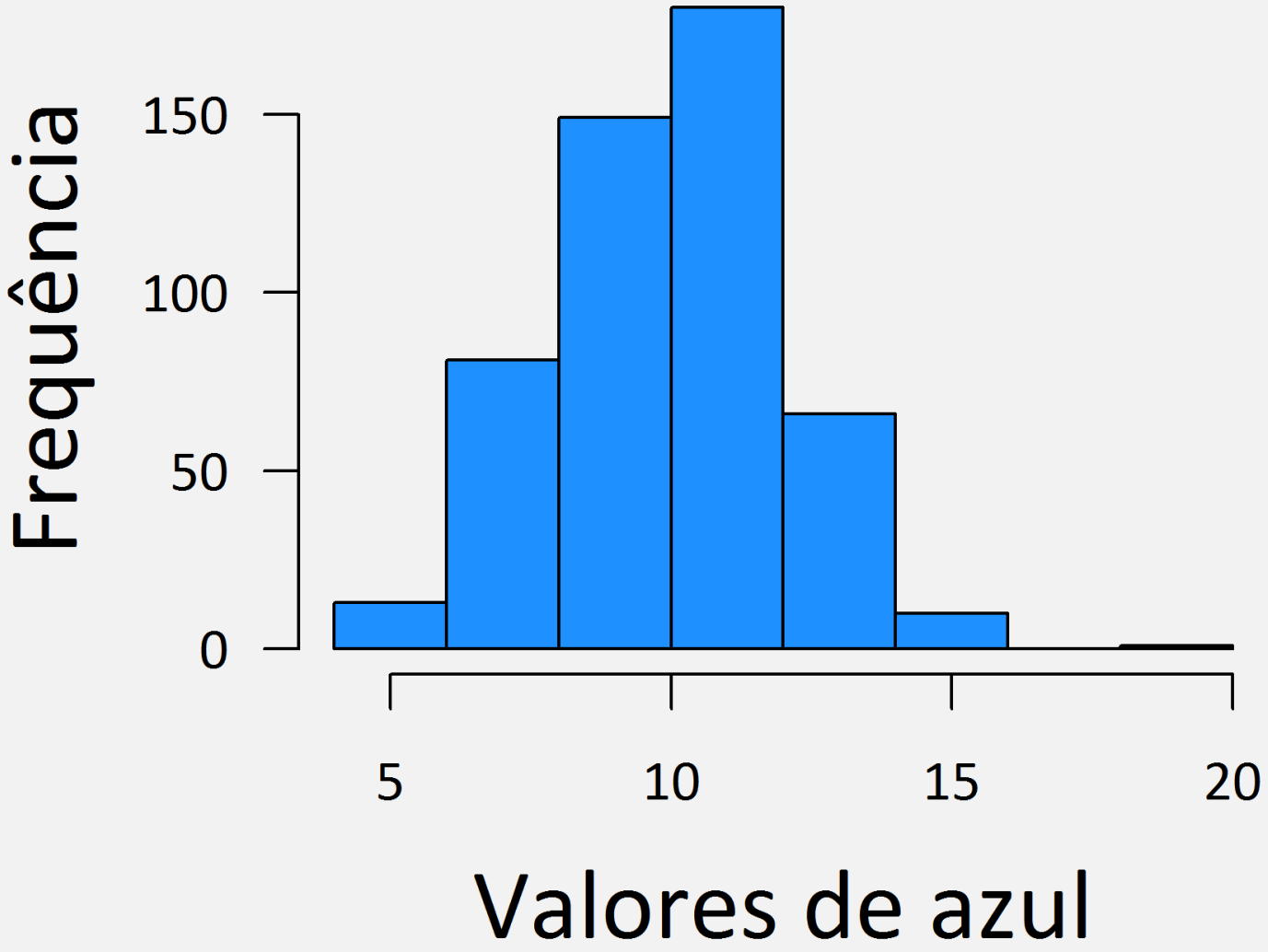


Boxplot solitário?

Função boxplot()



Histograma

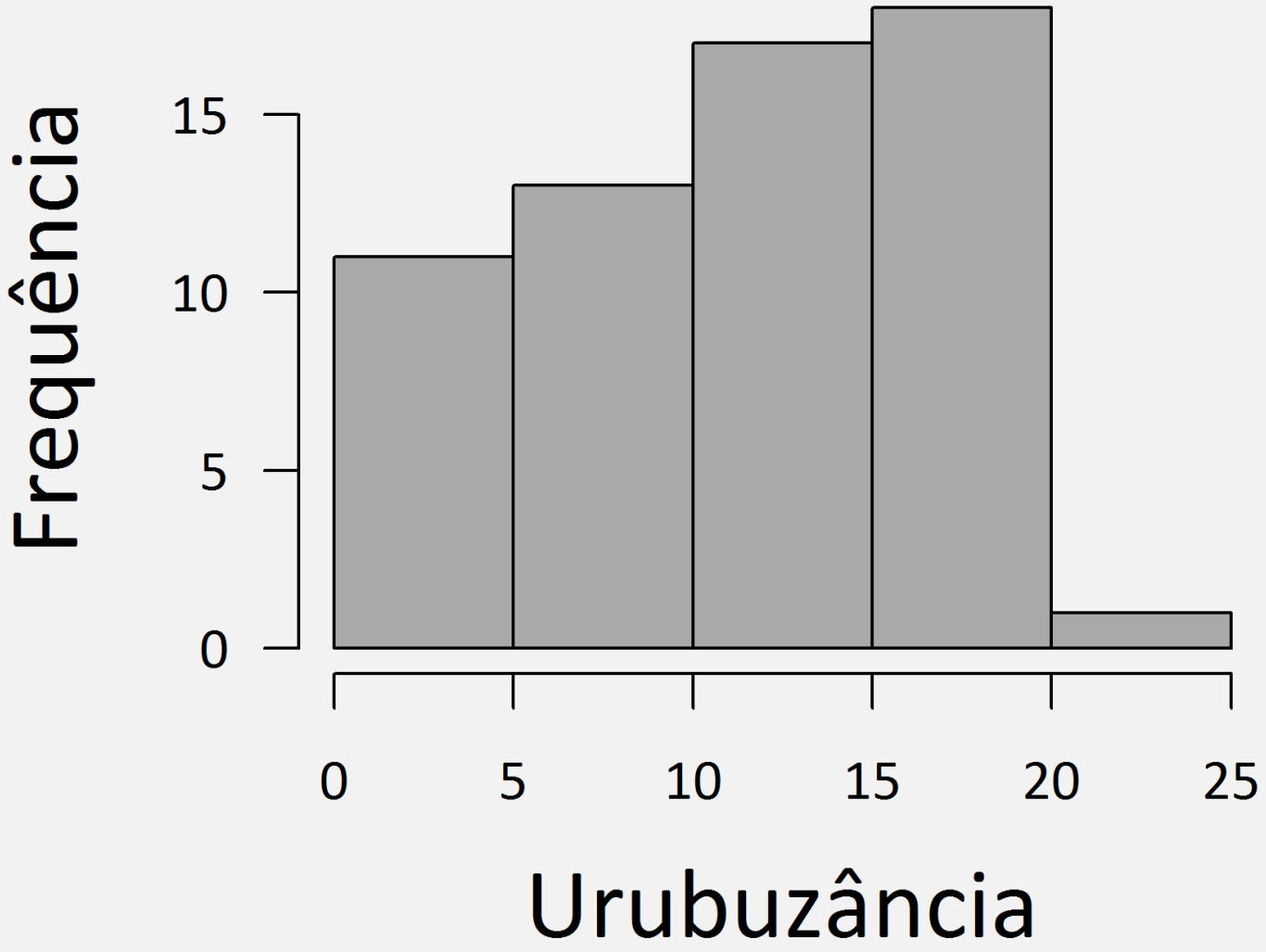


O problema do número de barras!

Vamos para as contagens de urubus em áreas de cerrado.

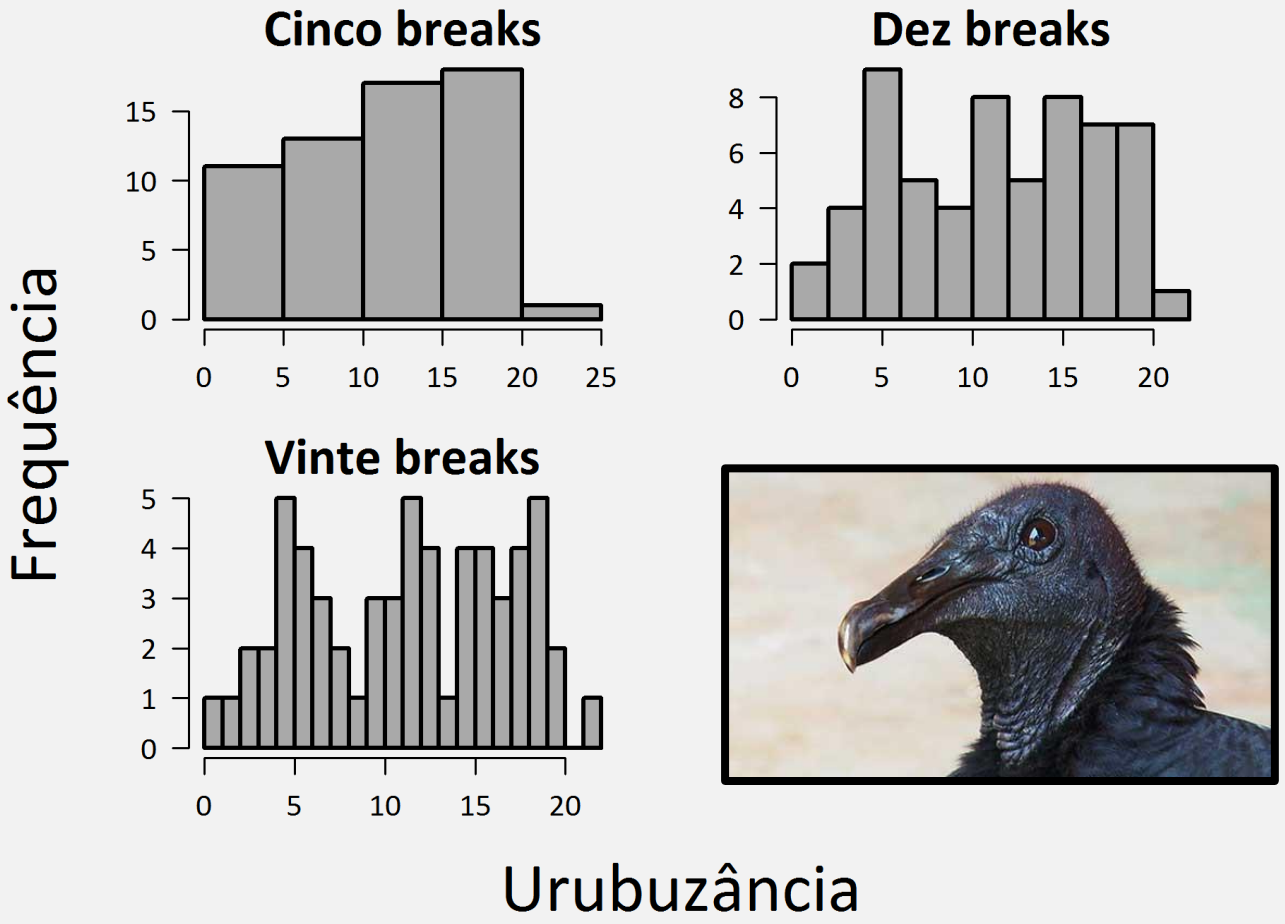
Urubuzando o histograma

Função hist()



Mas, e o número de barras?

Argumento breaks na função hist()



Tem como fugir das barras?

Sinto-me oprimido por essas barras...

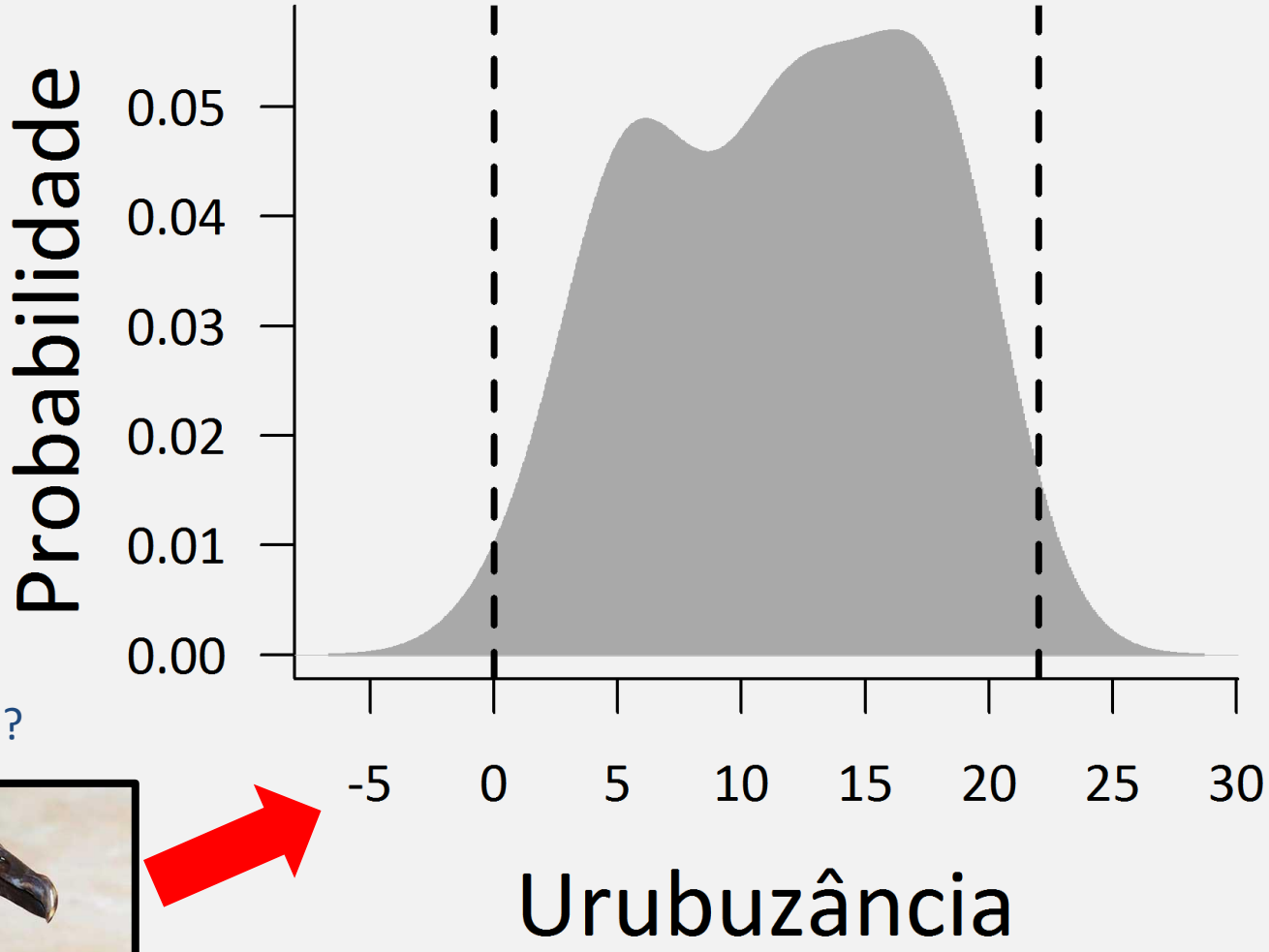
Gráfico de densidade!

plot(density(x, ...))

Função density() – estima uma função de densidade probabilística a partir do conjunto de dados (*kernel density*)

Função plot() – plota o gráfico de densidade

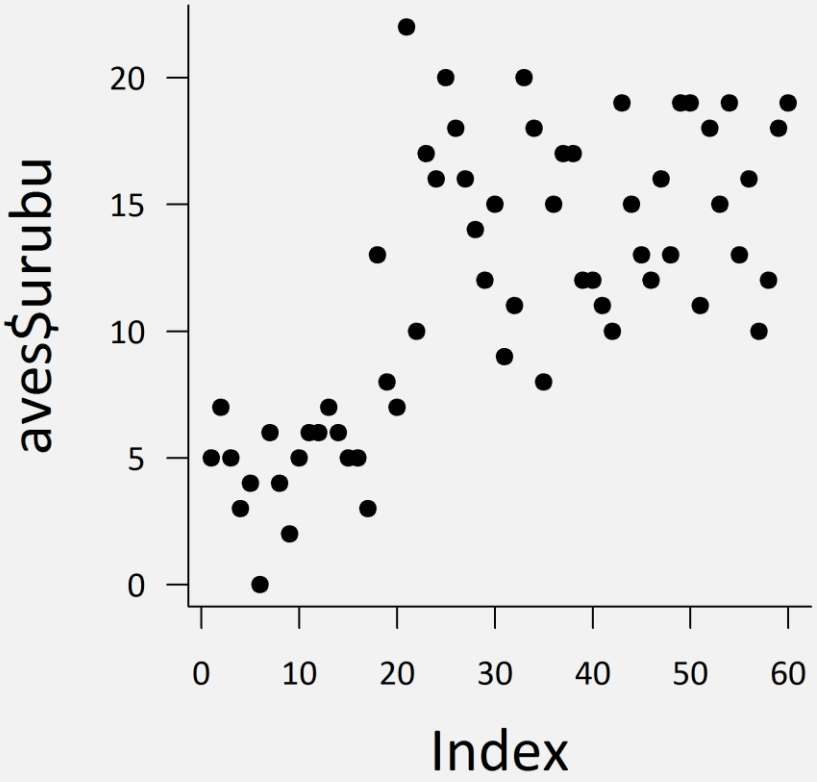
Densidade de urubus

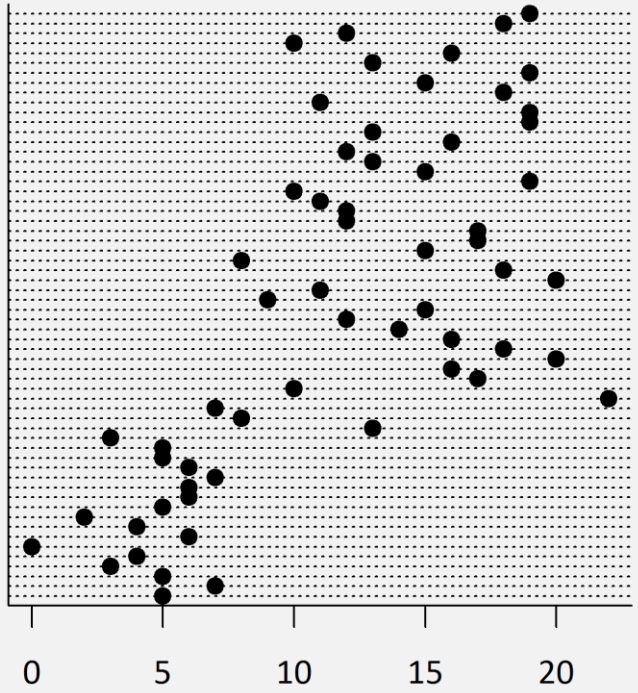


???????

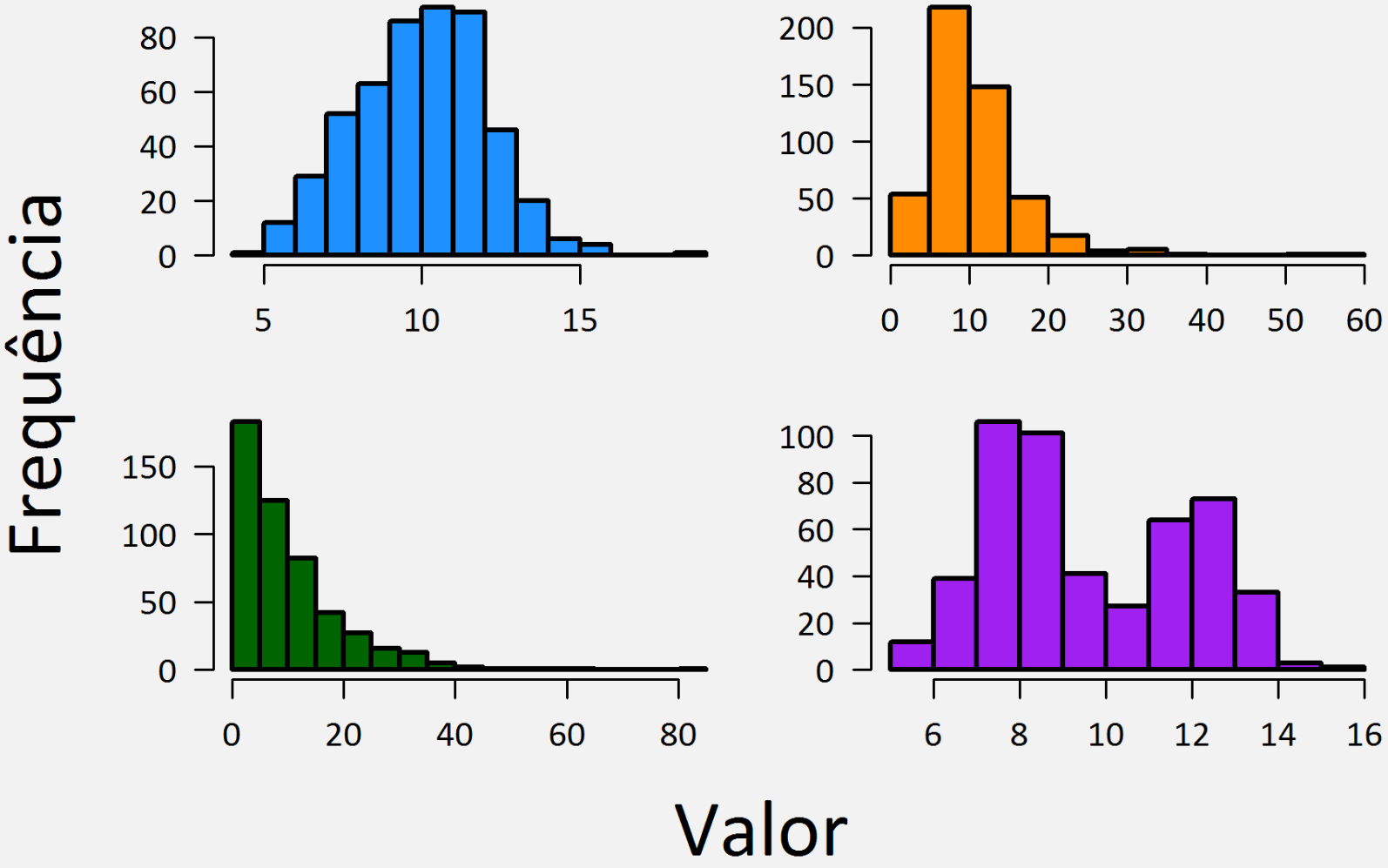
Que tal gráficos mais simples?

Funções plot() e dotchart()





Uma última olhada nos dados simulados



O segredo dos dados simulados

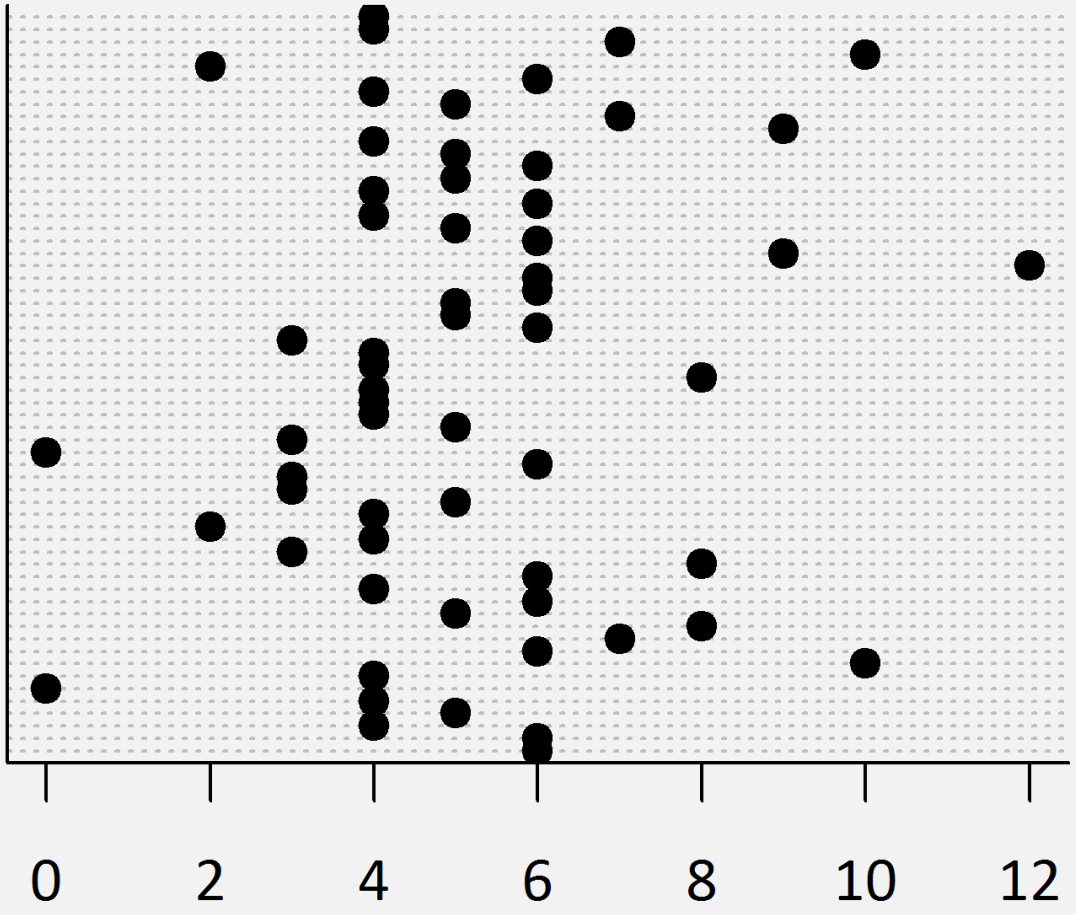
Usando as funções com início *r*

```
azul = rnorm(500, 10, 2)

laranja = rlnorm(500, log(10), 1/2)

verde = rexp(500, 1/10)

roxo = c(rnorm(300, 8, 1), rnorm(200, 12, 1))
```

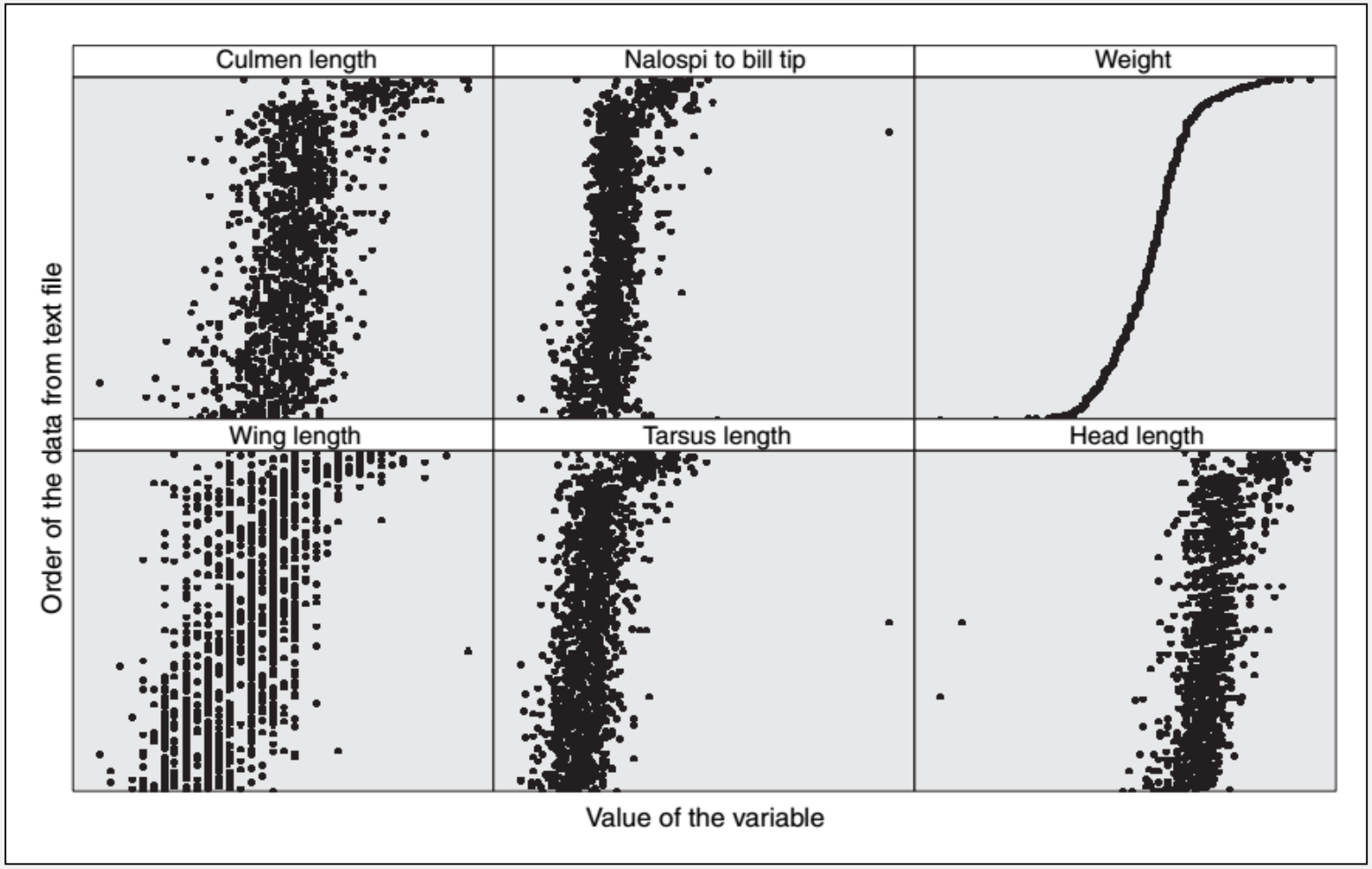
Valores extremos

Outliers!?

Valores extremos?



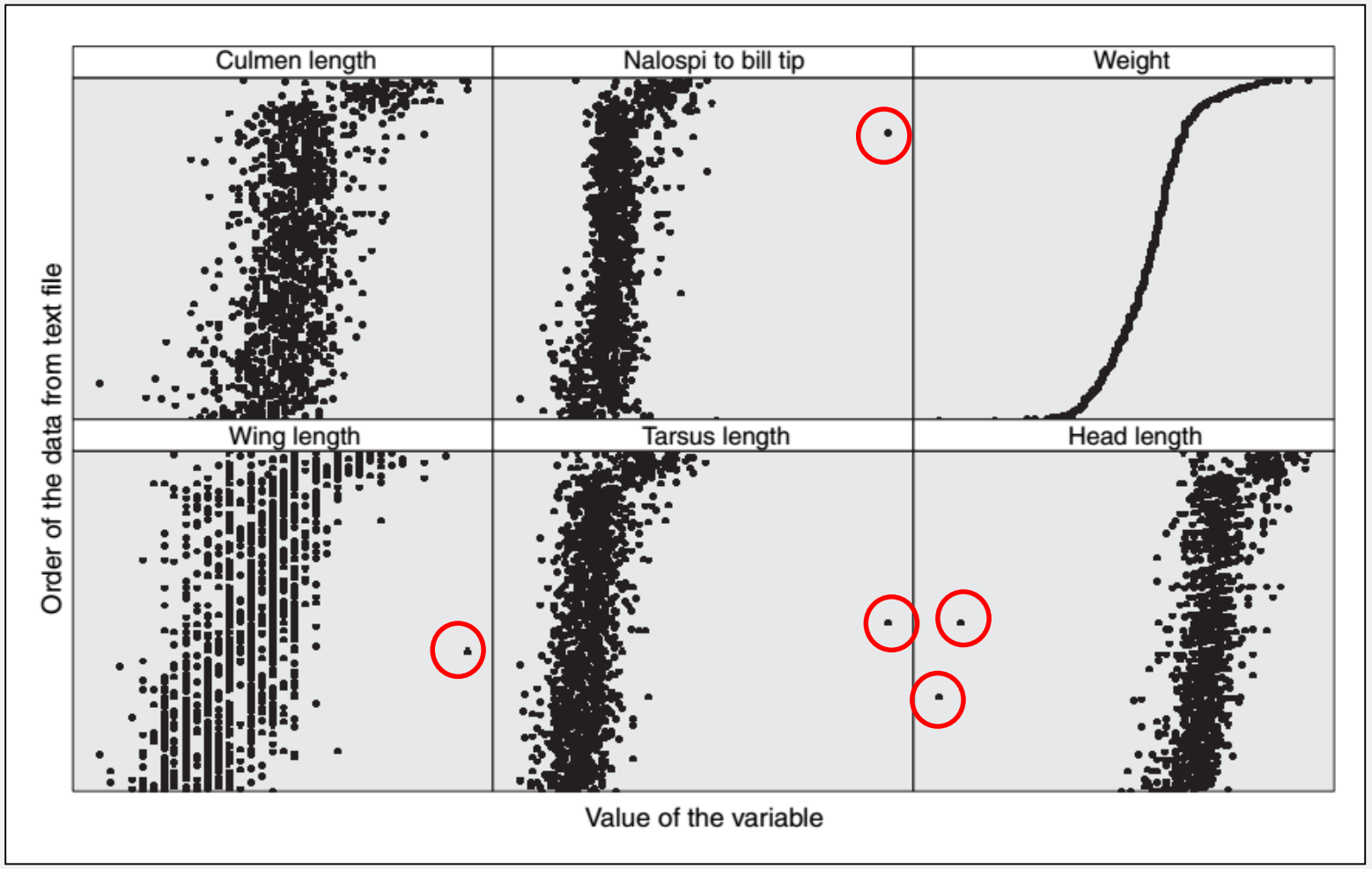
Valores extremos!



Zuur et al. 2010.

Medidas morfométricas de pardais

Valores extremos!



Zuur et al. 2010.

Medidas morfométricas de pardais

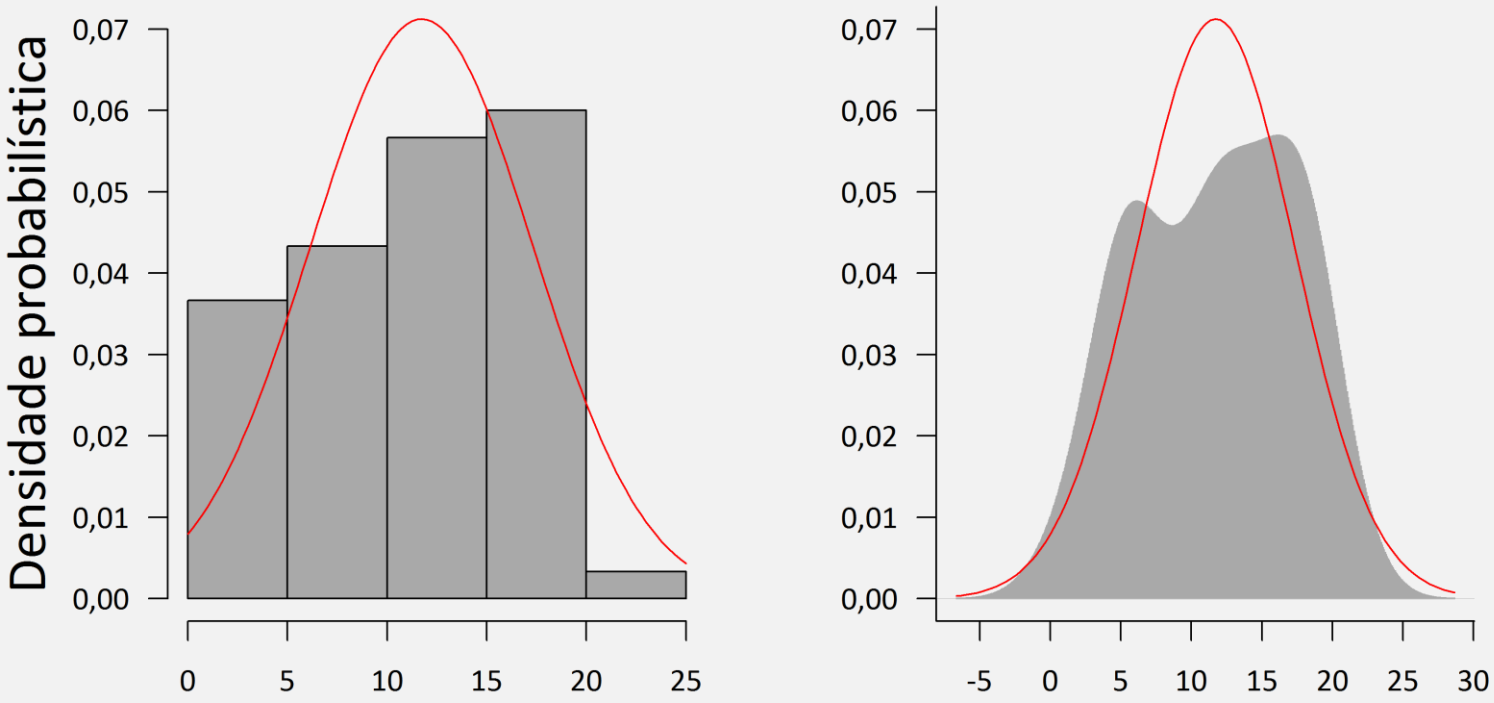
Valores extremos – o que fazer?

- Verificar erros de medida ou digitação (se possível)
- Considerar a exclusão de valores “impossíveis”
- Análise de sensibilidade

Meus dados são normais?

Curva empírica x teórica (normal)

curve(dnorm(x,...))



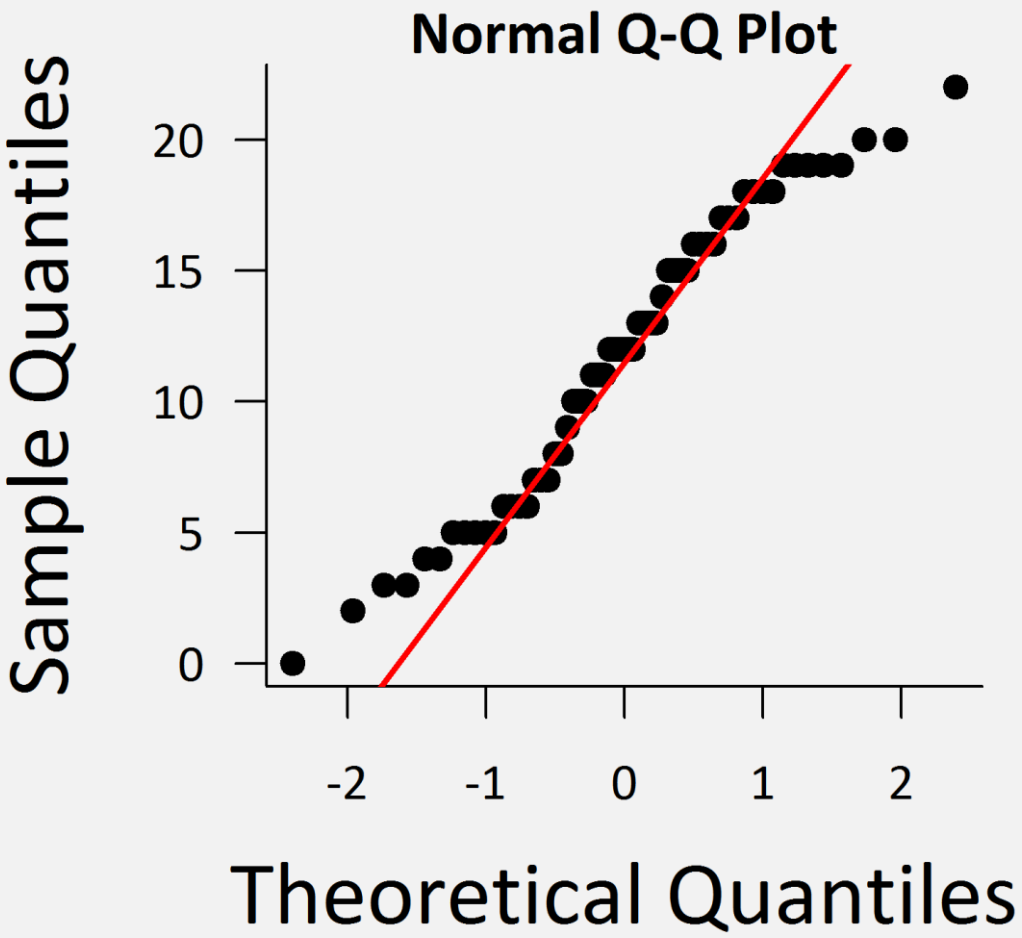
Funções qqnorm() e qqline()

- qqnorm()
  - "plota" os quantis empíricos (dos dados) contra os quantis teóricos (esperados)

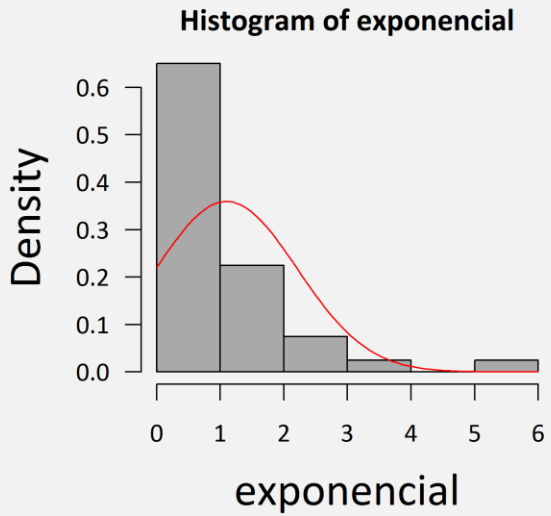
- qqline()
  - Adiciona uma linha com os quantis esperados

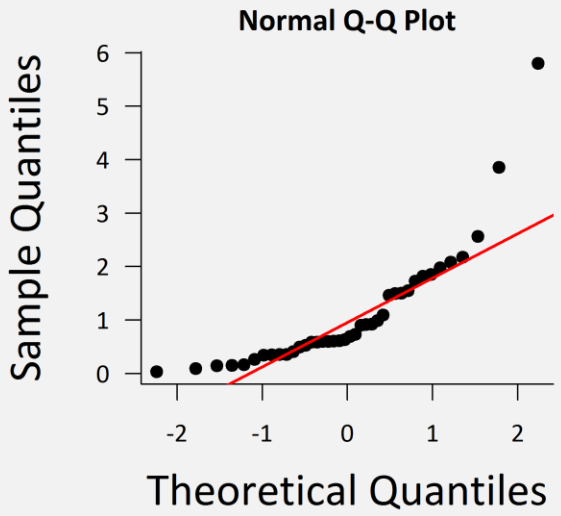
Funções qqnorm() e qqline()

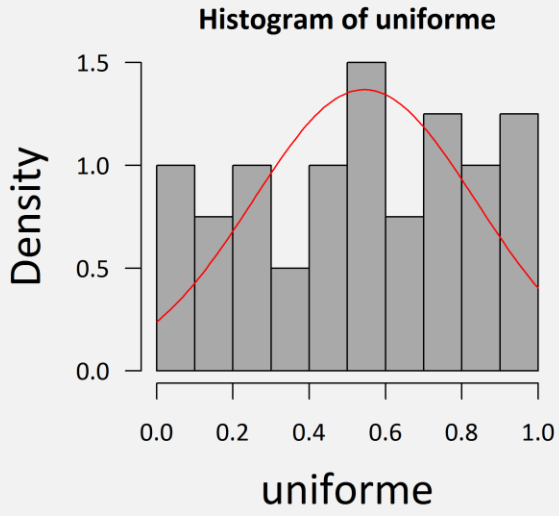
De volta aos dados dos urubus

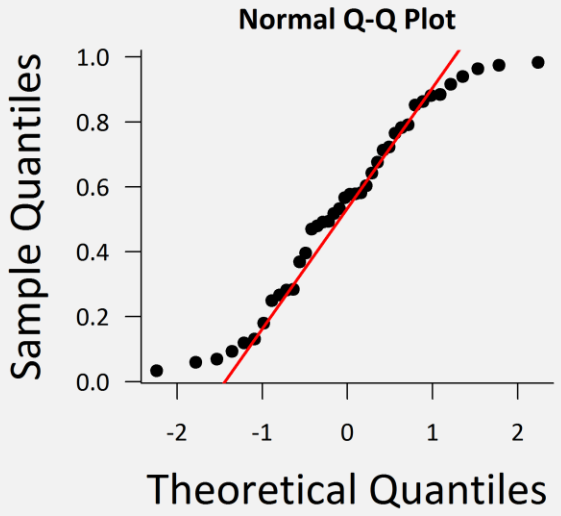


O melhor teste de normalidade do mundo!!

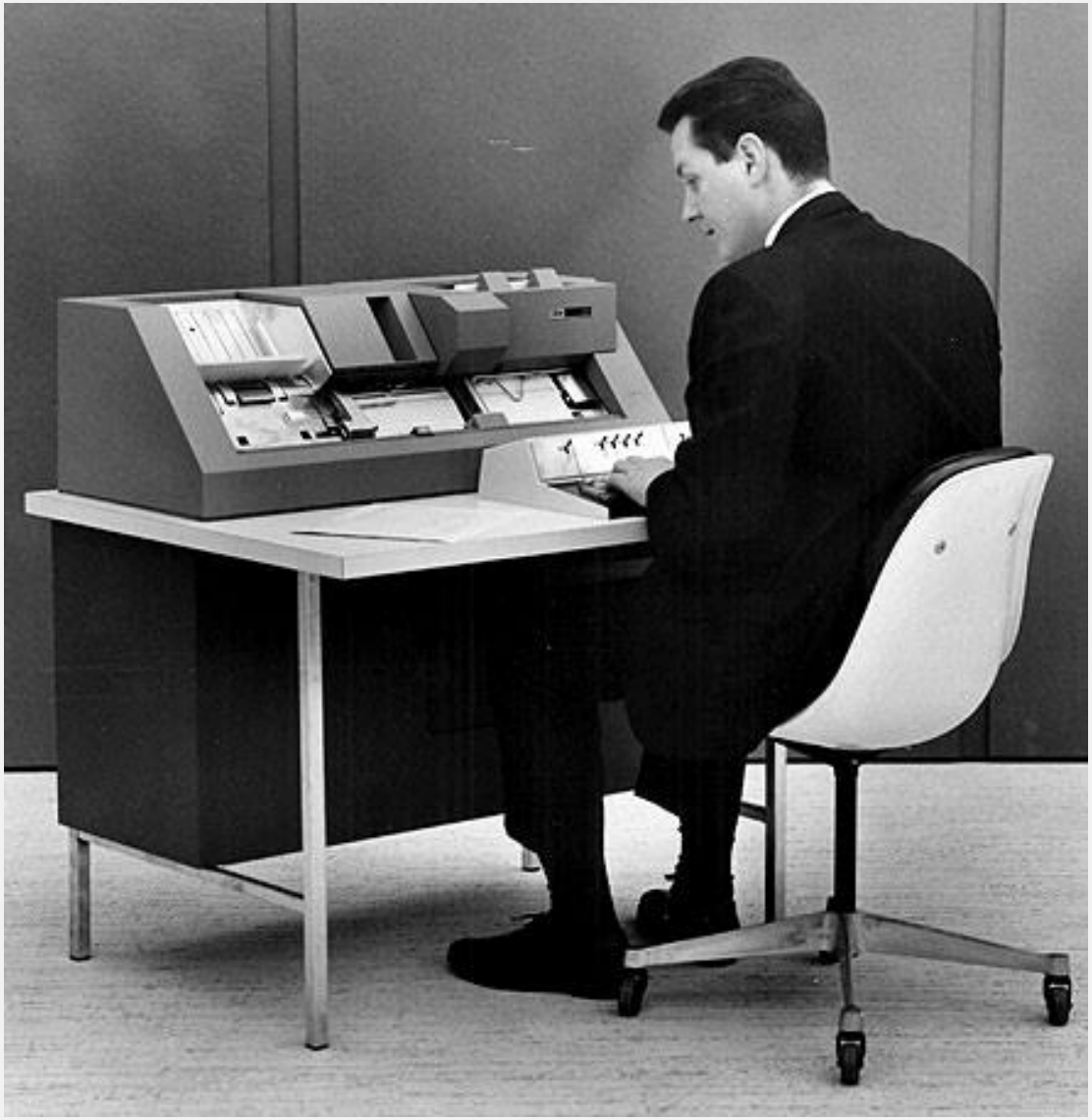








Vamos para o R!



Explorando dados qualitativos

Dados qualitativos?

- Também conhecidos como categóricos
- Dados que não podem ser expressos como números ou quantidades
- Coisas como identidade da espécie, tipo de ambiente e “tipo” de qualquer coisa

A incrível função table()!!

Direto do arquivo caixeta.csv

```
> table(cax$local)
chauas jureia retiro
426      241     360
>
```

barplot(table(x,...))

```
barplot(table(cax$local),
xlab="Local", ylab="Número de observações")
```

Tabelas multidimensionais

```
> table(cax$especie,cax$local)

                          chauas jureia retiro
  Alchornea triplinervia       0       3     12
  Andira fraxinifolia          0       4      0
  bombacaceae                  0       1      0
  Cabralea canjerana           0       4      0
  Callophyllum brasiliensis    7       0      0
  Calophyllum brasiliensis     0       4      0
  Cecropia sp                  0       0      1
  Coussapoa macrocarpa         0       3
  Coussapoa micropoda          2       0
  Cryptocaria moschata         0       2
```

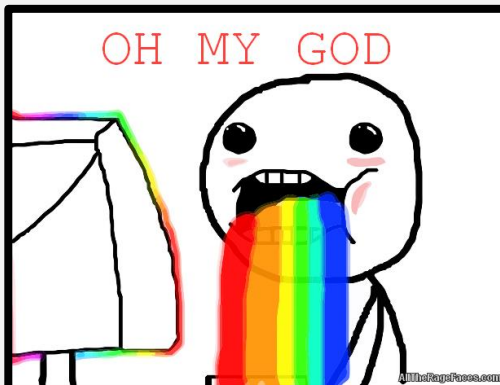
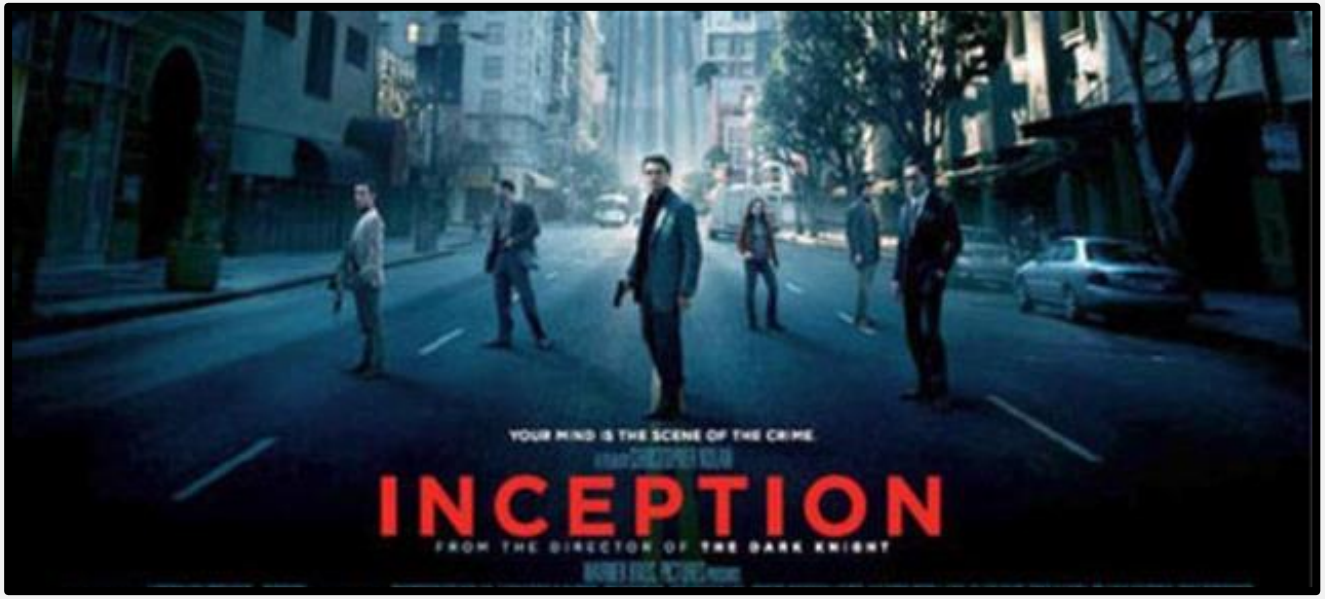


Table Inception!!!

```
> table(table(cax$especie))

  1   2   3   4   7   8   9  10  15  20  37  63  96 698
 15  10   3   5   1   1   1   1   1   1   1   1   1   1
```



Juntando contagens com xtabs()

```
> TitanicDF = data.frame(Titanic)
> head(TitanicDF)
  Class    Sex   Age Survived Freq
1   1st   Male Child       No    0
2   2nd   Male Child       No    0
3   3rd   Male Child       No   35
4  Crew   Male Child       No    0
5   1st Female Child       No    0
6   2nd Female Child       No    0
>
```

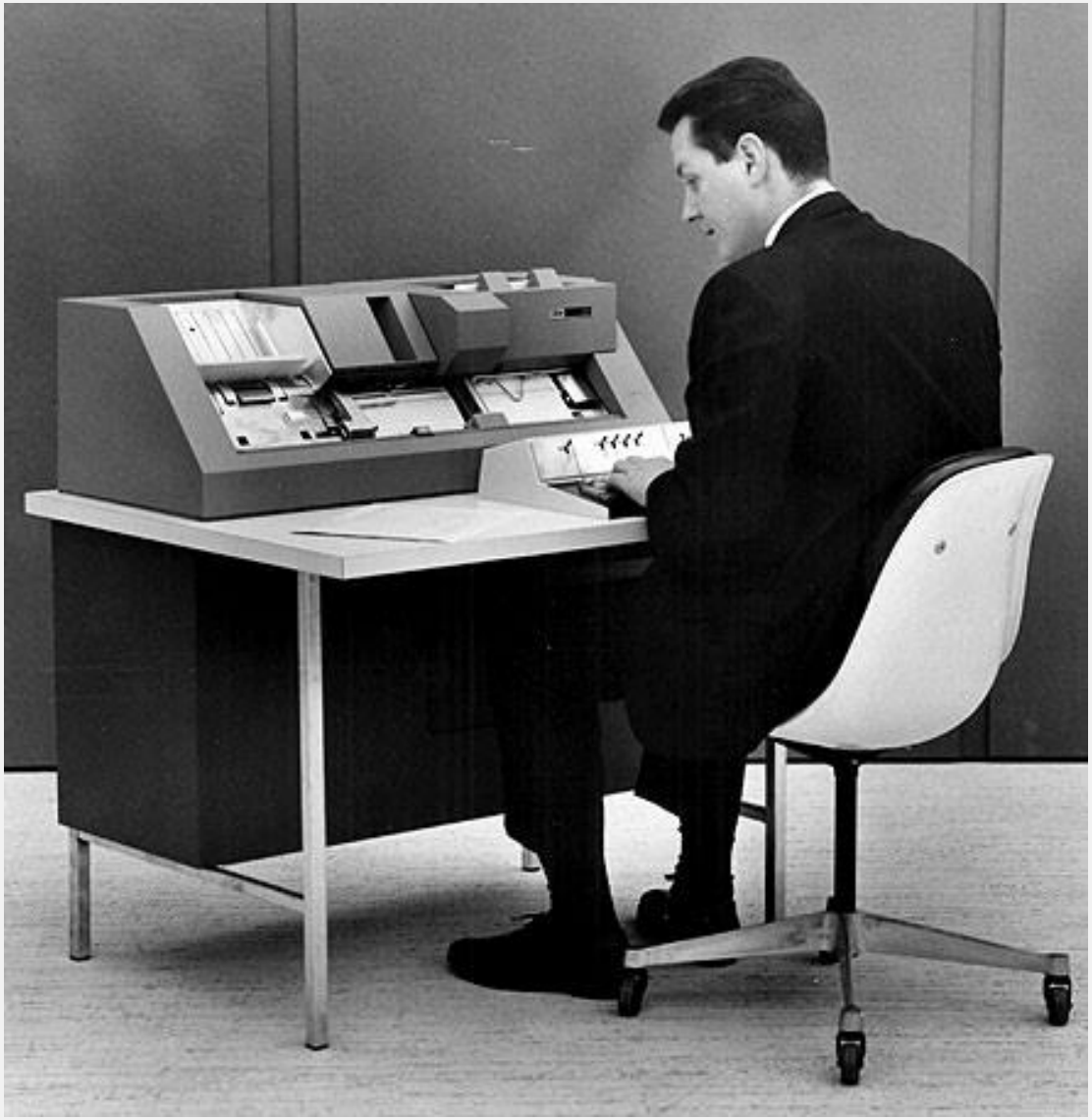
Juntando contagens com xtabs()

```
> xtabs(Freq~Survived+Sex,
data=TitanicDF)
          Sex
Survived Male Female
      No  1364    126
      Yes  367    344
```

Juntando contagens com xtabs()

```
> xtabs(Freq~Survived+Sex+Age, data=TitanicDF)
, , Age = Child

        Sex
Survived Male Female
     No    35     17
     Yes   29     28

, , Age = Adult

        Sex
Survived Male Female
     No  1329    109
     Yes  338    316
```

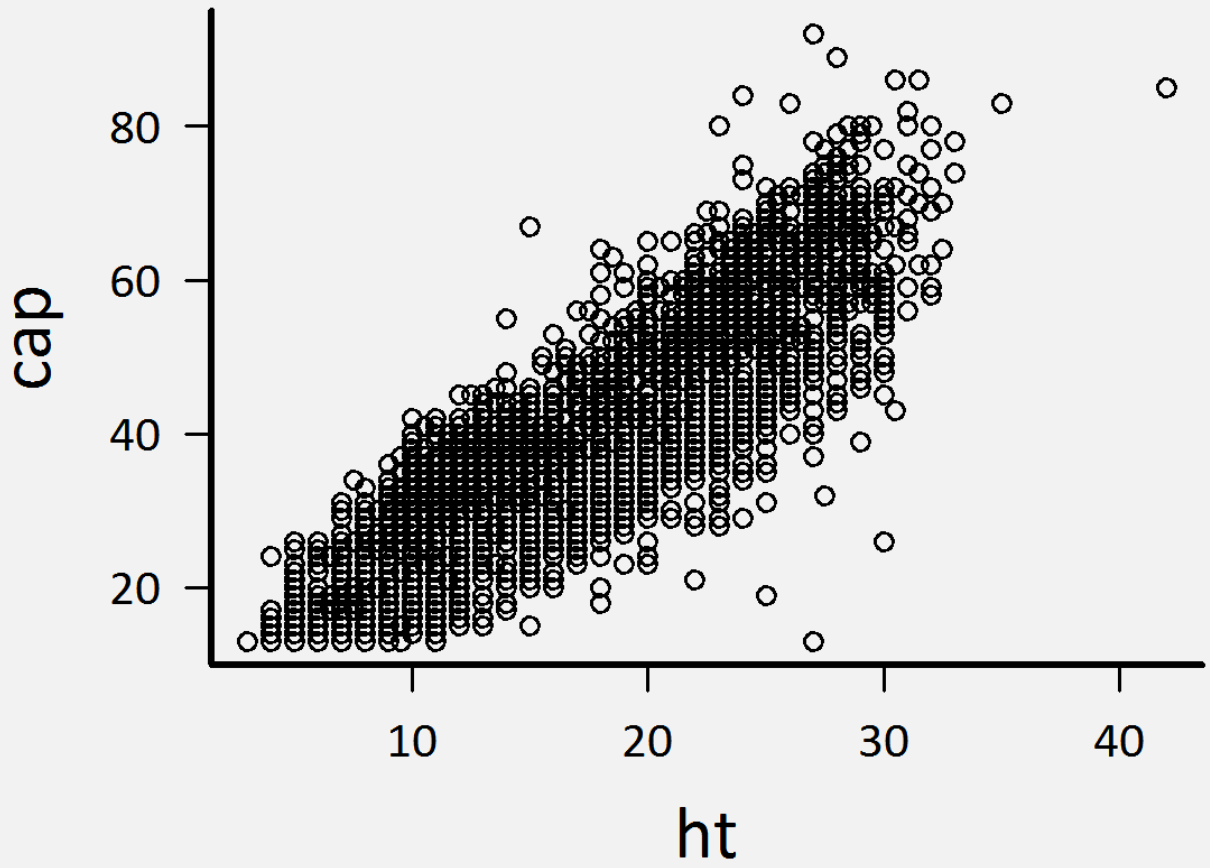
Vamos para o R!



Exploração bivariada de dados

De volta à multiflexível função plot()

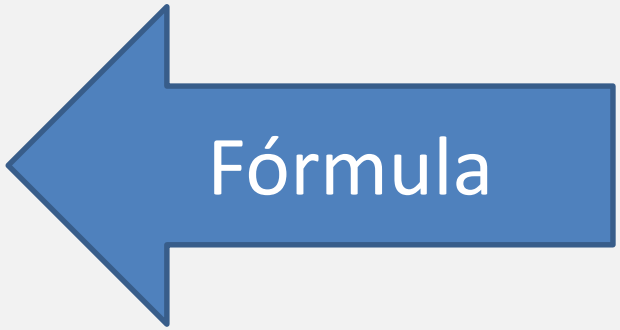
Duas variáveis contínuas
Dados de egrandis.csv



Muitas formas de chamar um plot

```
plot(x=egr$ht, y=egr$cap)
```

```
plot(cap~ht, data=egr)
```




```
plot(egr[,c("ht","cap")])
```

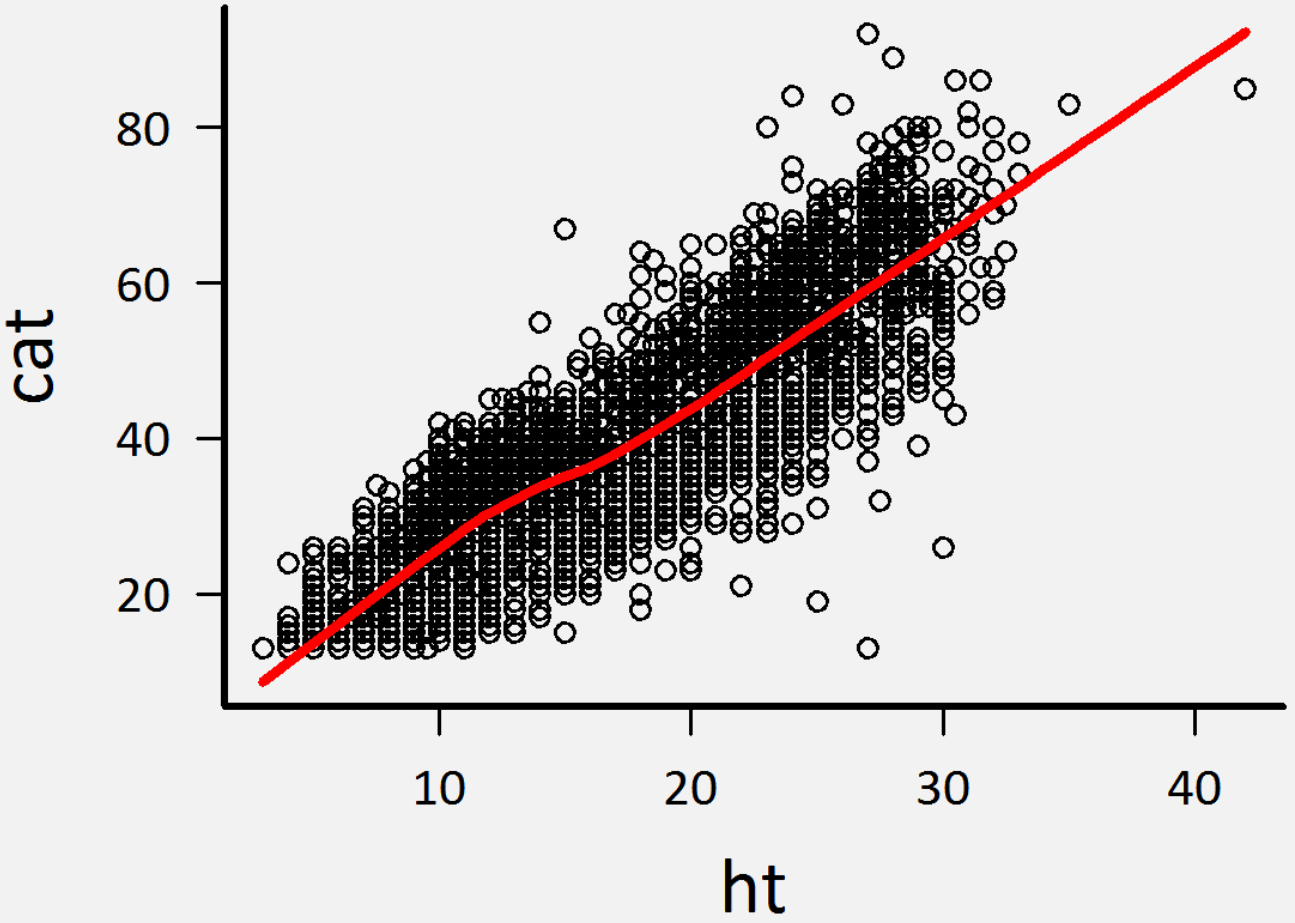
Existe relação entre as minhas variáveis?

- Coeficientes de correlação
  - função cor()

```
> cor(egr$cap, egr$ht)
[1] 0.8770064
```

Existe relação entre as minhas variáveis?

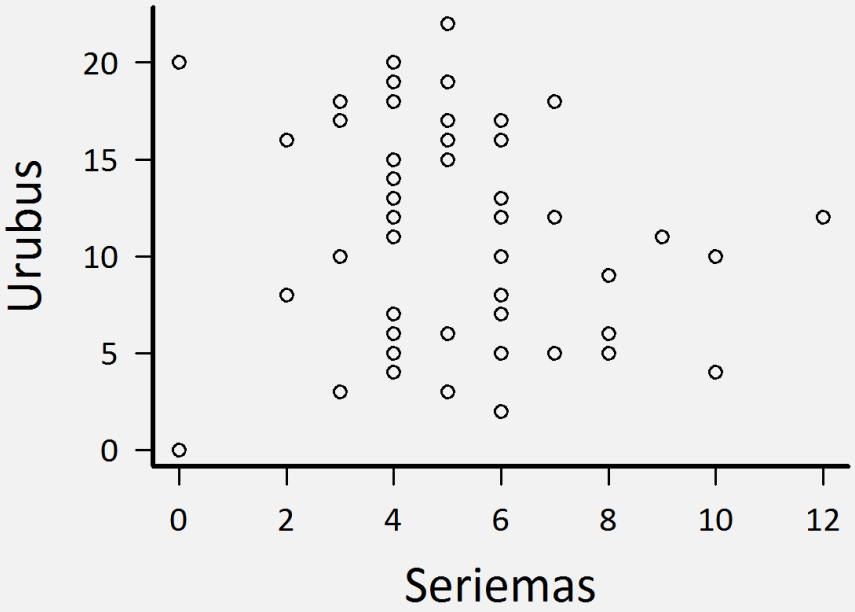
Função scatter.smooth()

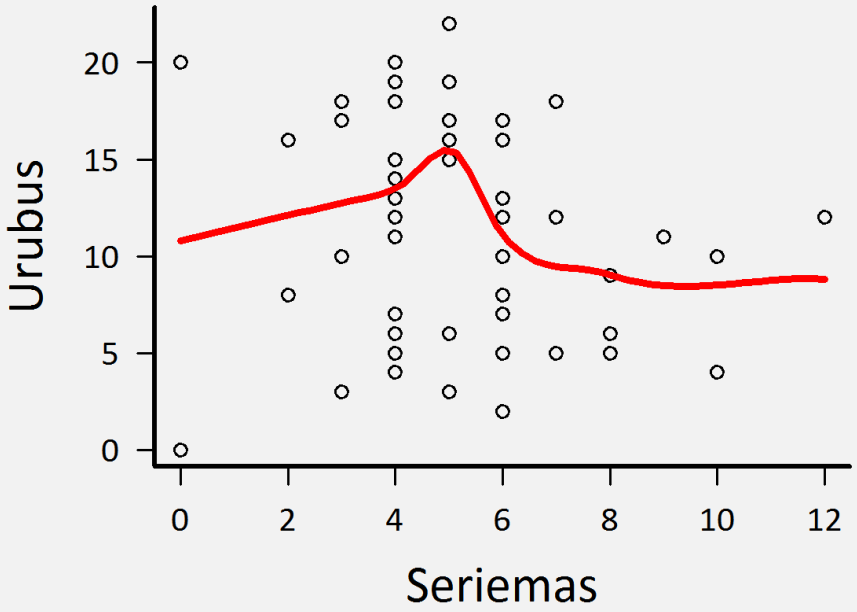


Mas a correlação é significativa?

Espera a aula de modelos lineares!

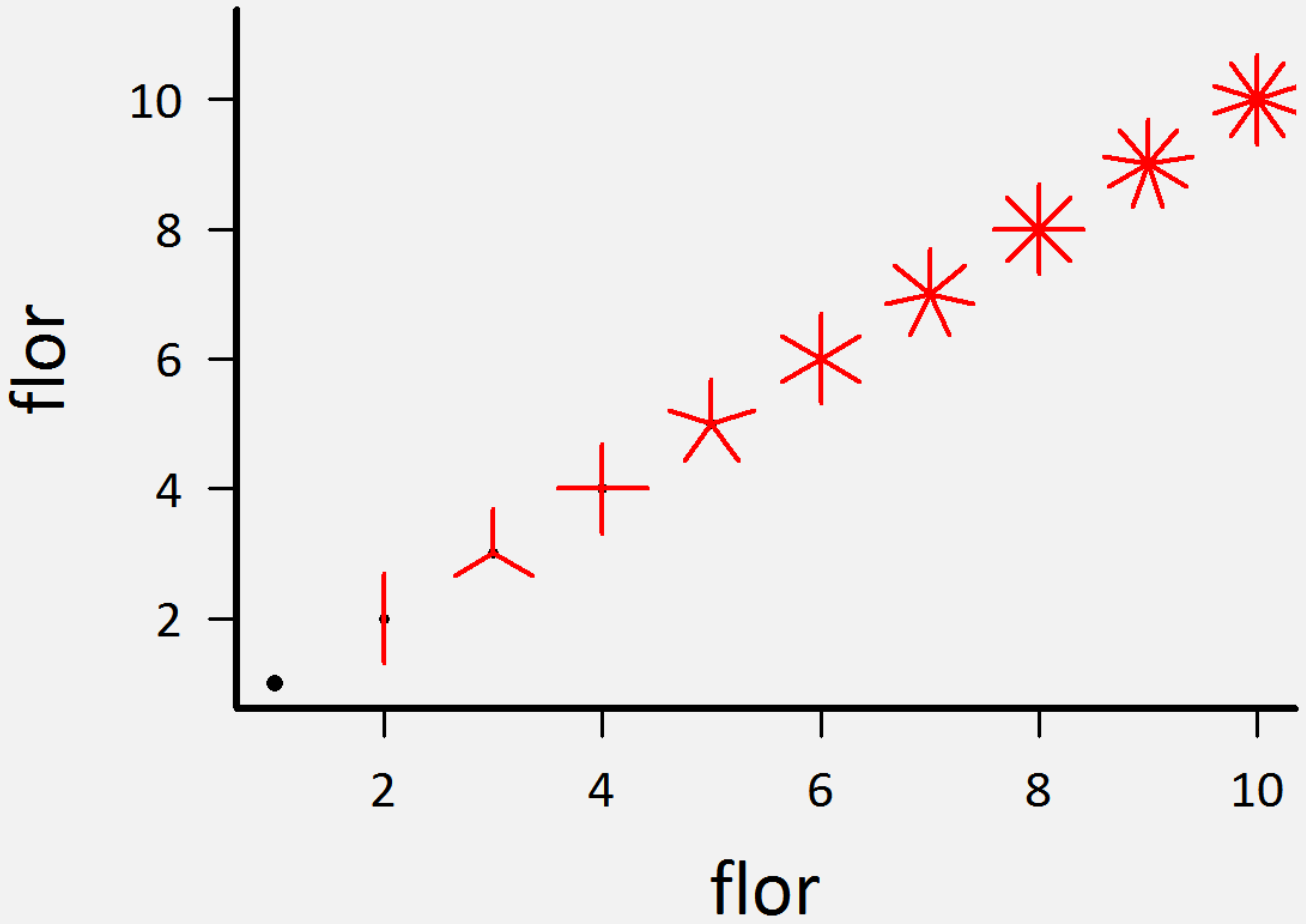
Cuidado com scatter.smooth()





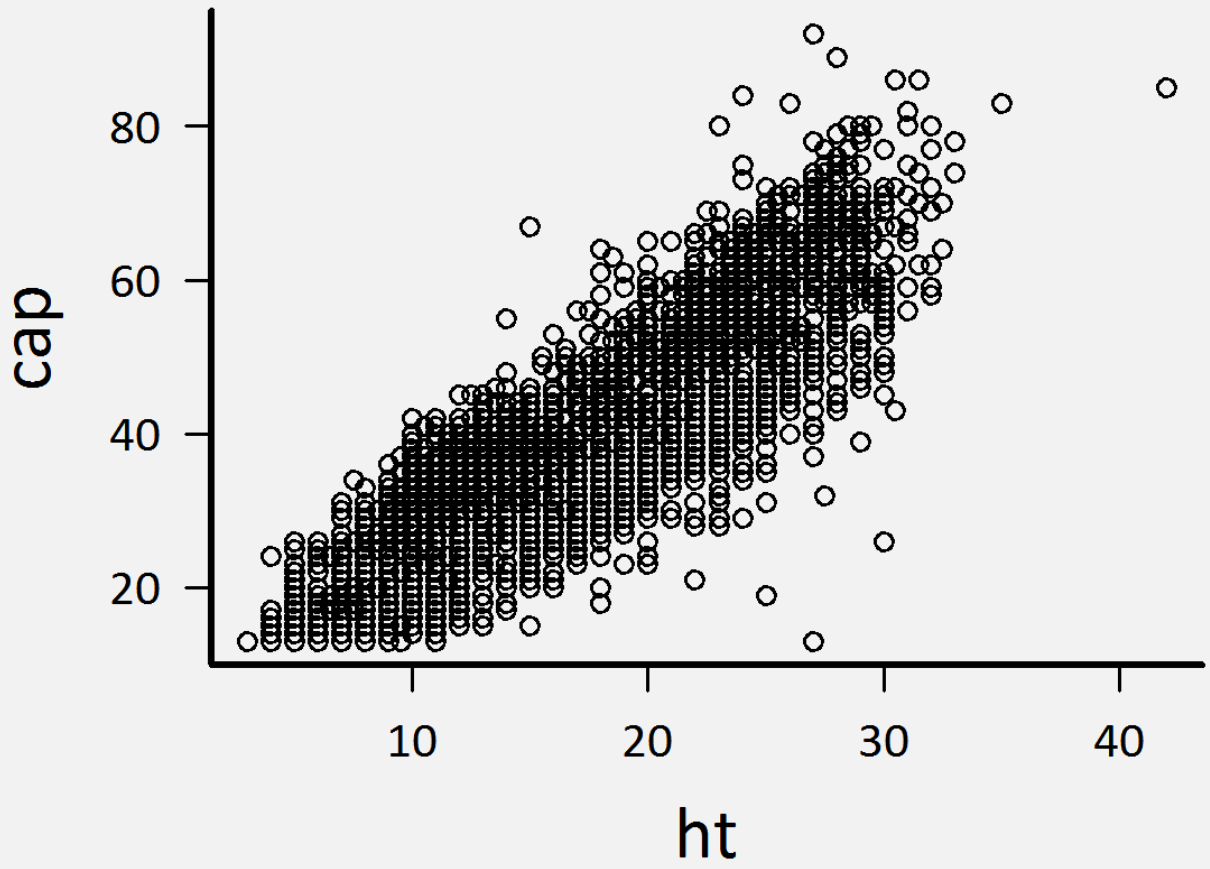
Gráficos de florzinha com sunflowerplot()

```
flor = rep(1:10, 1:10)
sunflowerplot(flor,flor)
```

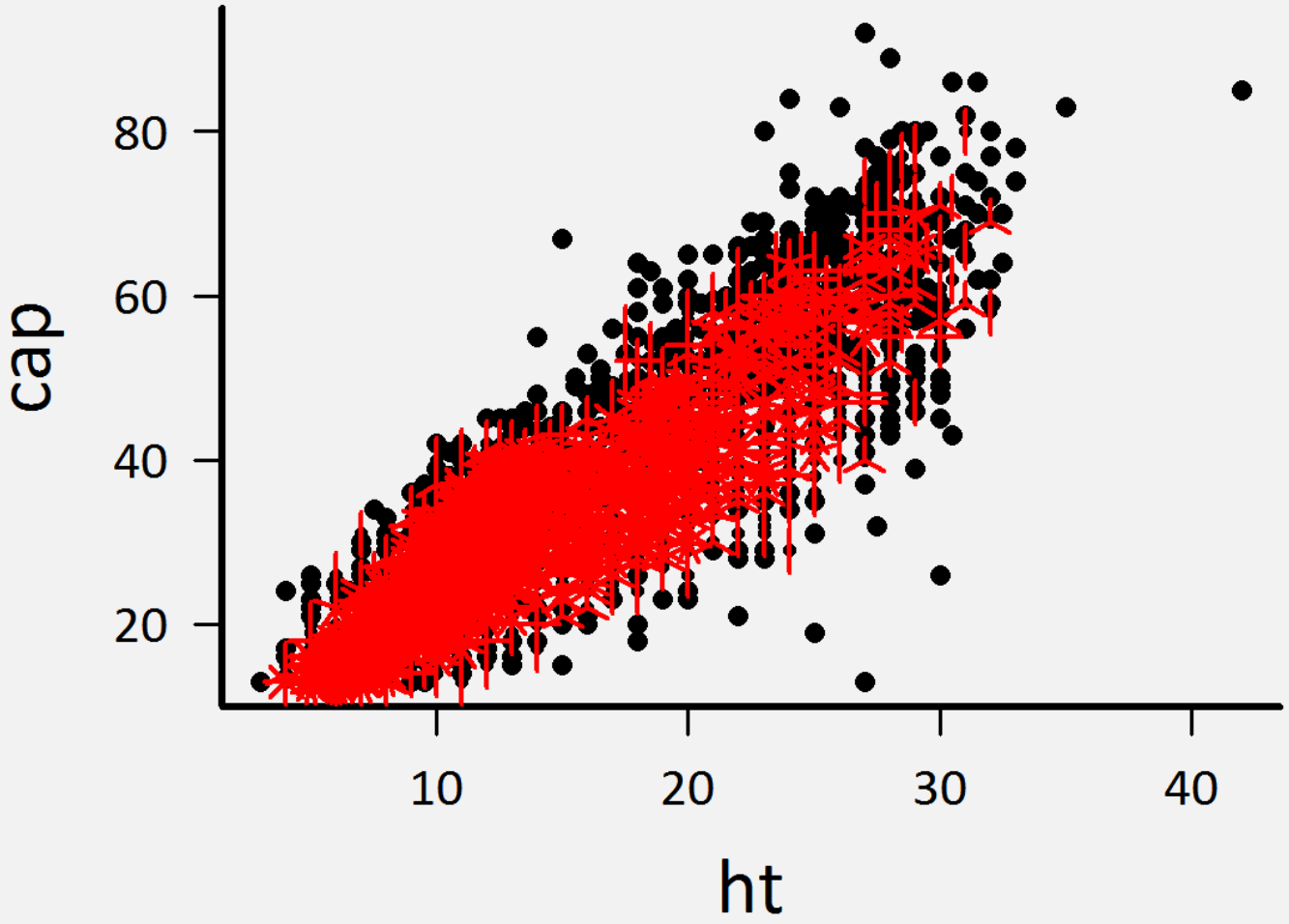


De volta à multiflexível função plot()

Duas variáveis contínuas



Milhões de pontos sobrepostos!



Quarteto de Anscombe

```
> data(anscombe)
> anscombe
   x1 x2 x3 x4    y1   y2    y3    y4
1  10 10 10  8  8.04 9.14  7.46  6.58
2   8  8  8  8  6.95 8.14  6.77  5.76
3  13 13 13  8  7.58 8.74 12.74  7.71
4   9  9  9  8  8.81 8.77  7.11  8.84
5  11 11 11  8  8.33 9.26  7.81  8.47
6  14 14 14  8  9.96 8.10  8.84  7.04
7   6  6  6  8  7.24 6.13  6.08  5.25
8   4  4  4 19  4.26 3.10  5.39 12.50
9  12 12 12  8 10.84 9.13  8.15  5.56
10  7  7  7  8  4.82 7.26  6.42  7.91
11  5  5  5  8  5.68 4.74  5.73  6.89
```

F.J. Anscombe

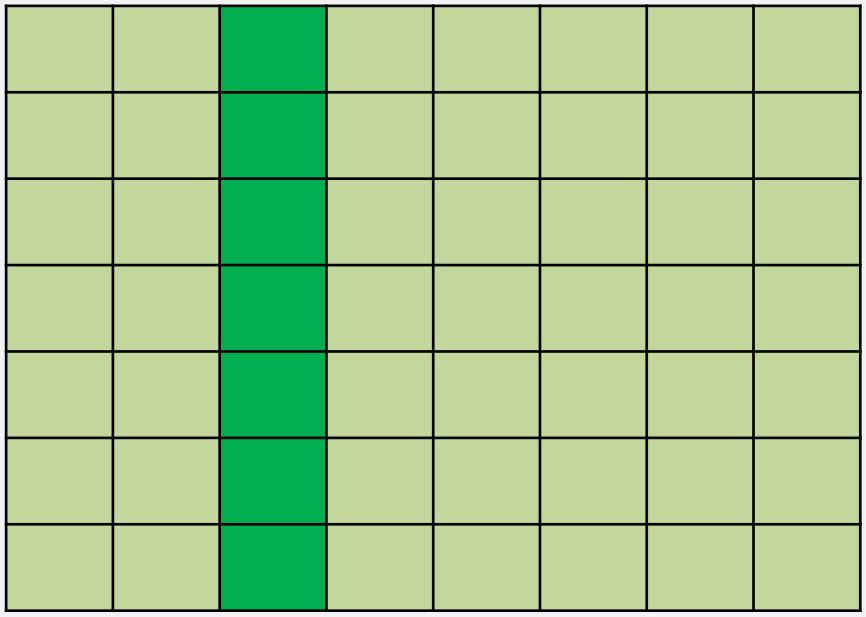
Jamais de esqueça da família (apply)

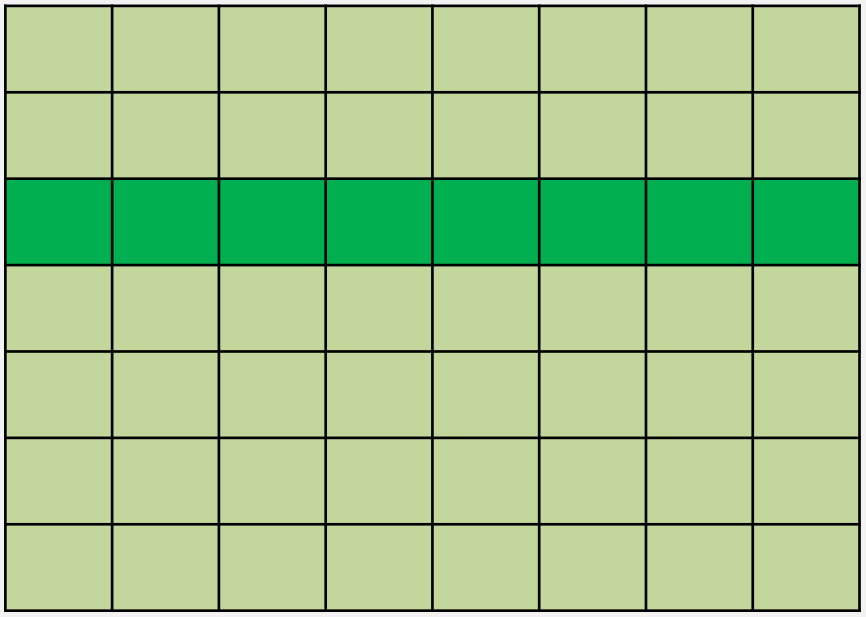
Função *apply()*

Função *apply*()

Opera sobre matrizes

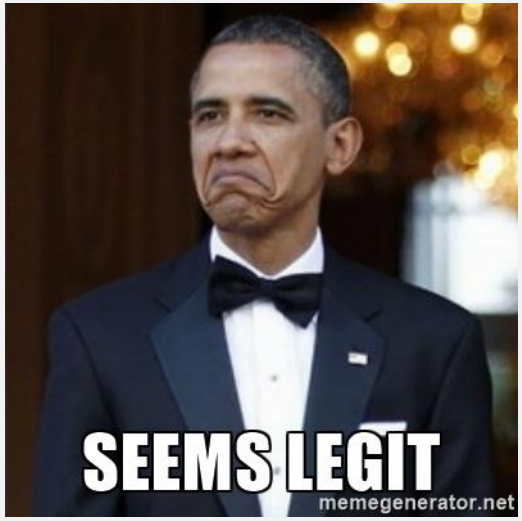
Realiza uma operação sobre todas as linhas ou colunas





Explorando o quarteto

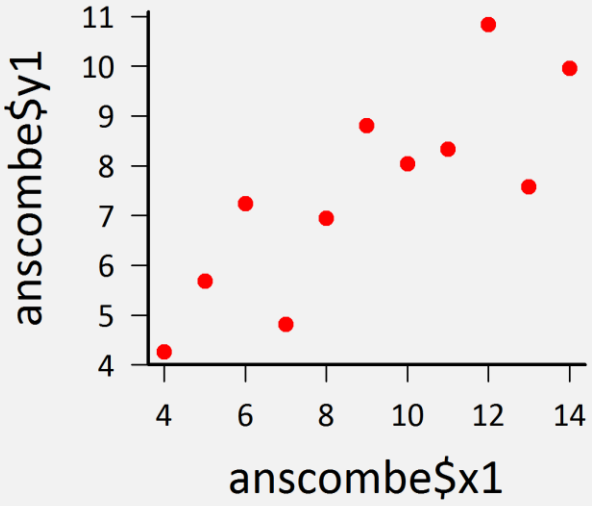
```
> apply(anscombe[,1:4], 2, mean)
x1 x2 x3 x4
 9  9  9  9
> apply(anscombe[,1:4], 2, sd)
       x1        x2        x3        x4
3.316625 3.316625 3.316625 3.316625
>
> apply(anscombe[,5:8], 2, mean)
       y1        y2        y3        y4
7.500909 7.500909 7.500000 7.500909
> apply(anscombe[,5:8], 2, sd)
       y1        y2        y3        y4
2.031568 2.031657 2.030424 2.030579
```

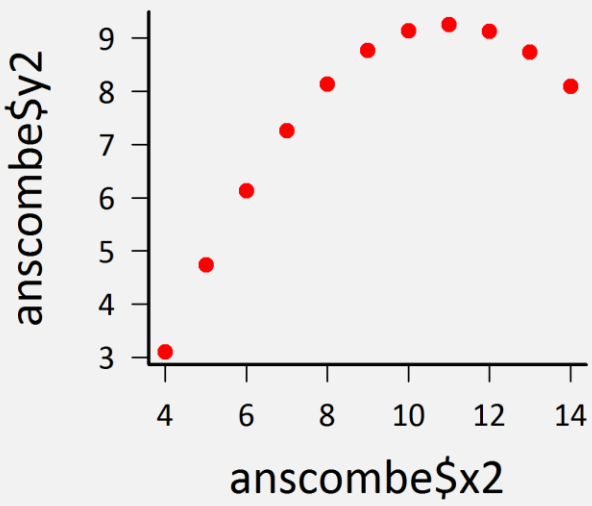


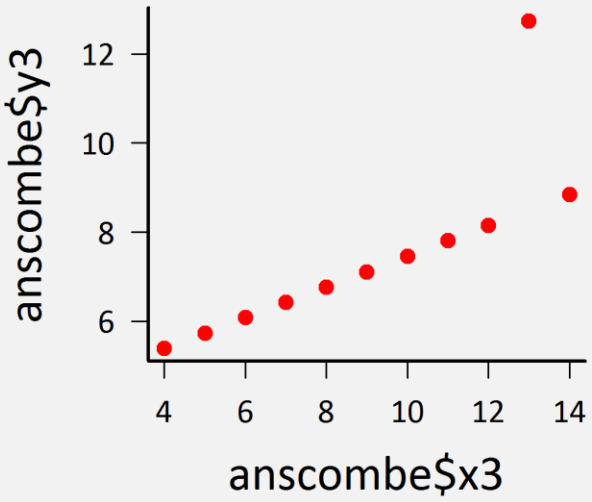
Explorando o quarteto

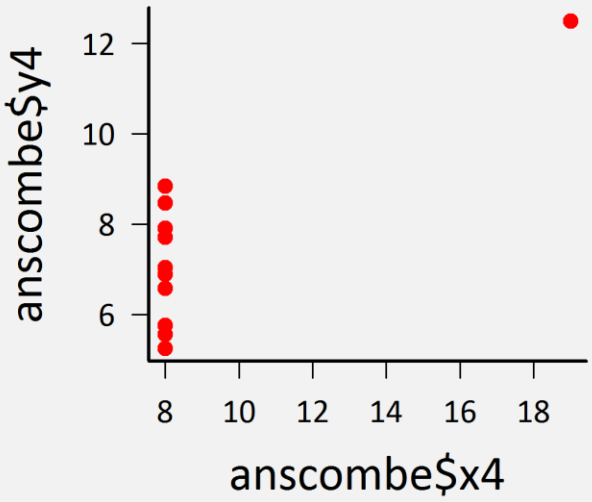
```
> cor(anscombe$x1, anscombe$y1)
[1] 0.8164205
> cor(anscombe$x2, anscombe$y2)
[1] 0.8162365
> cor(anscombe$x3, anscombe$y3)
[1] 0.8162867
> cor(anscombe$x4, anscombe$y4)
[1] 0.8165214
```

Explorando o quarteto

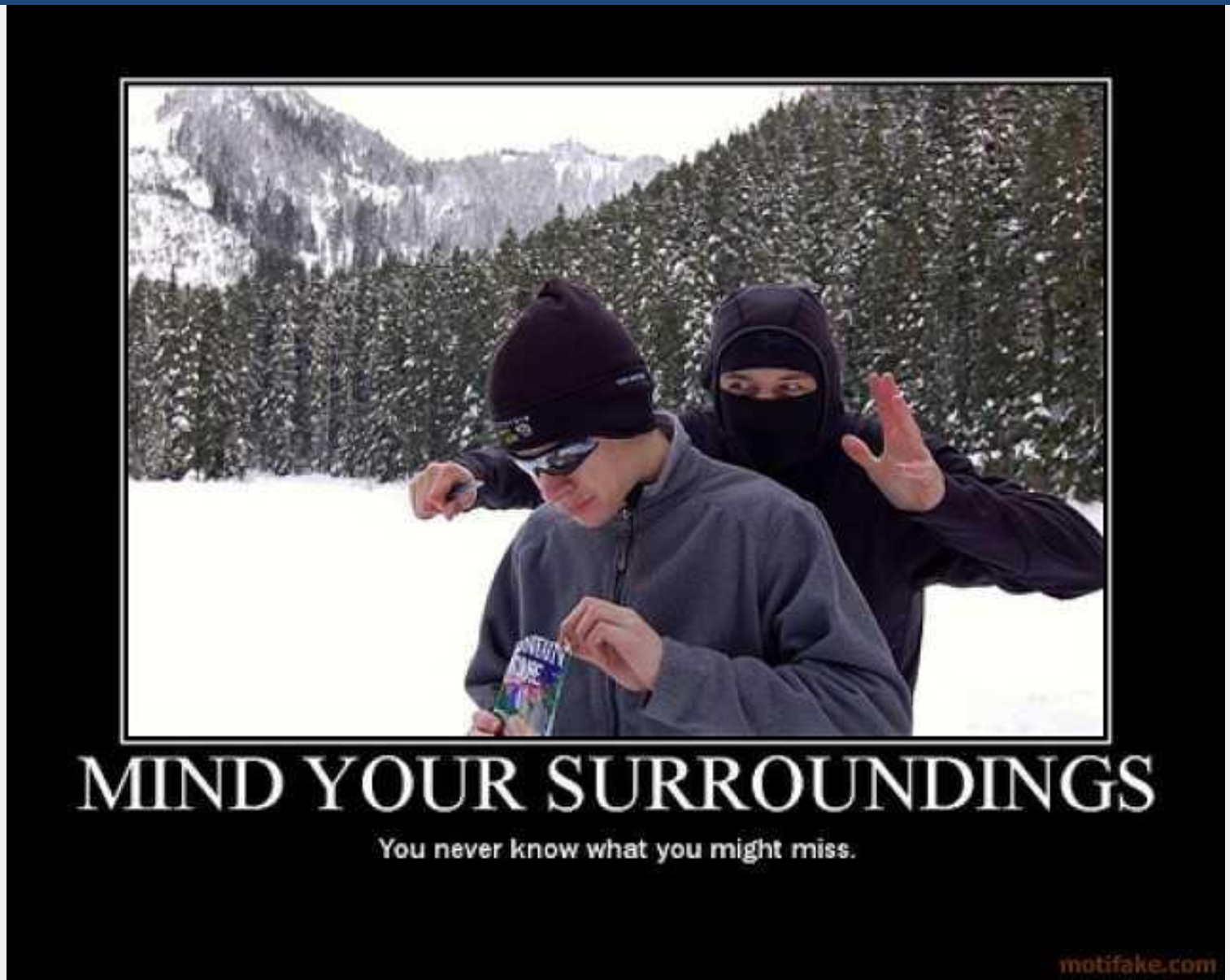








Conheça bem seus dados!



Variável contínua x categórica

Jamais se esqueça da família (apply)!

Funções *tapply*() e *aggregate*()

Realiza uma operação sobre conjuntos de uma variável, separados por um índice (outra variável)

Funções *tapply*() e *aggregate*()

Ideais para calcular médias (ou coisa parecida) de acordo com sexo, classe de idade, espécie ou coisa assim.

Sumário de variáveis descritoras

Funções tapply() e aggregate()

```
> tapply(aves$urubu, aves$fisionomia, mean)
   CC    Ce    CL
14.95  5.35 14.90
> aggregate(aves$urubu, list(aves$fisionomia), mean)
  Group.1     x
1      CC 14.95
2      Ce  5.35
3      CL 14.90
```

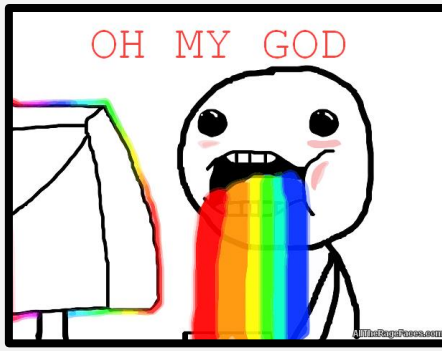
Sumário de variáveis descritoras

Funções tapply() e aggregate()

```
>tapply(aves$urubu, aves$fisionomia, sd)
       CC        Ce        CL
3.872644 2.621269 3.338768
> aggregate(aves$urubu, list(aves$fisionomia), sd)
  Group.1        x
1      CC 3.872644
2      Ce 2.621269
3      CL 3.338768
```

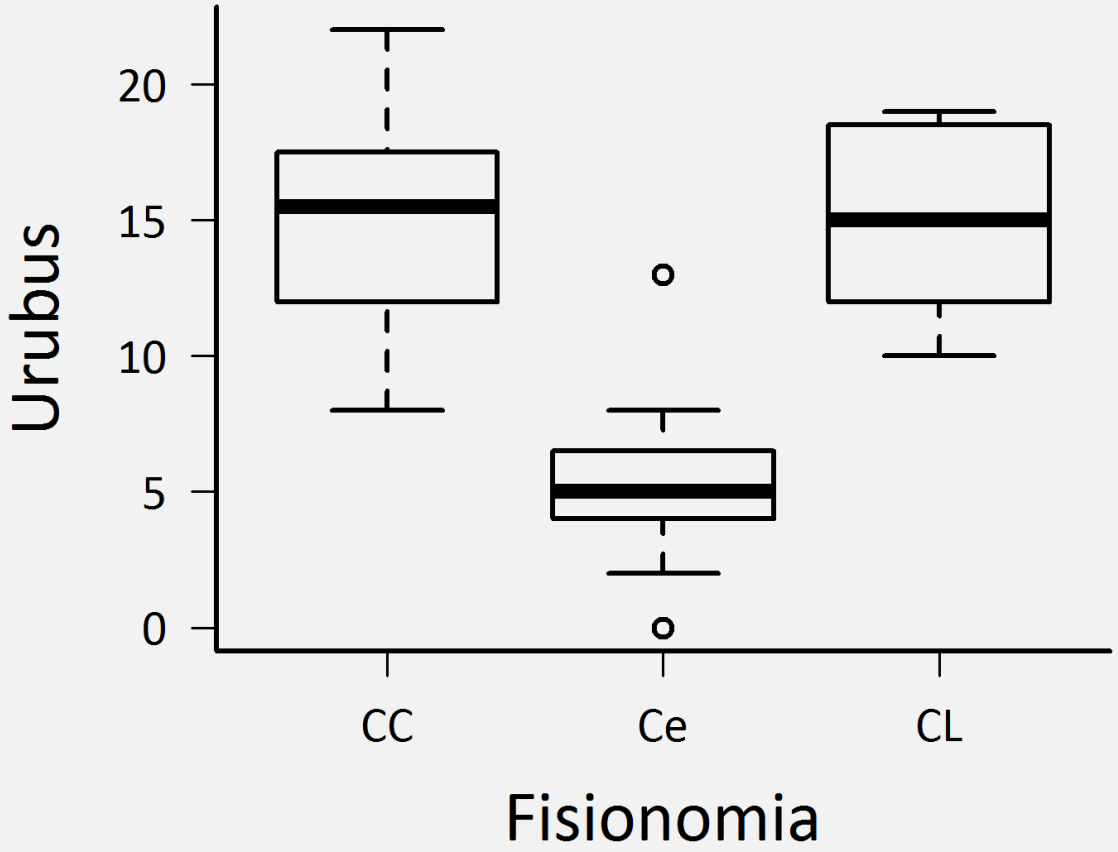
Mais de uma dimensão!

```
> aggregate(cax$h, list(cax$local,cax$especie), mean)
   Group.1                    Group.2         x
1   jureia    Alchornea triplinervia 130.00000
2   retiro    Alchornea triplinervia  60.83333
3   jureia       Andira fraxinifolia  67.50000
4   jureia                bombacaceae 150.00000
5   jureia        Cabralea canjerana 122.50000
6   chauas Callophyllum brasiliensis 142.85714
7   jureia  Calophyllum brasiliensis 117.50000
8   retiro               Cecropia sp  70.00000
9   jureia      Coussapoa macrocarpa  86.66667
10  chauas       Coussapoa micropoda  80.00000
11  retiro       Coussapoa micropoda  88.57143
```



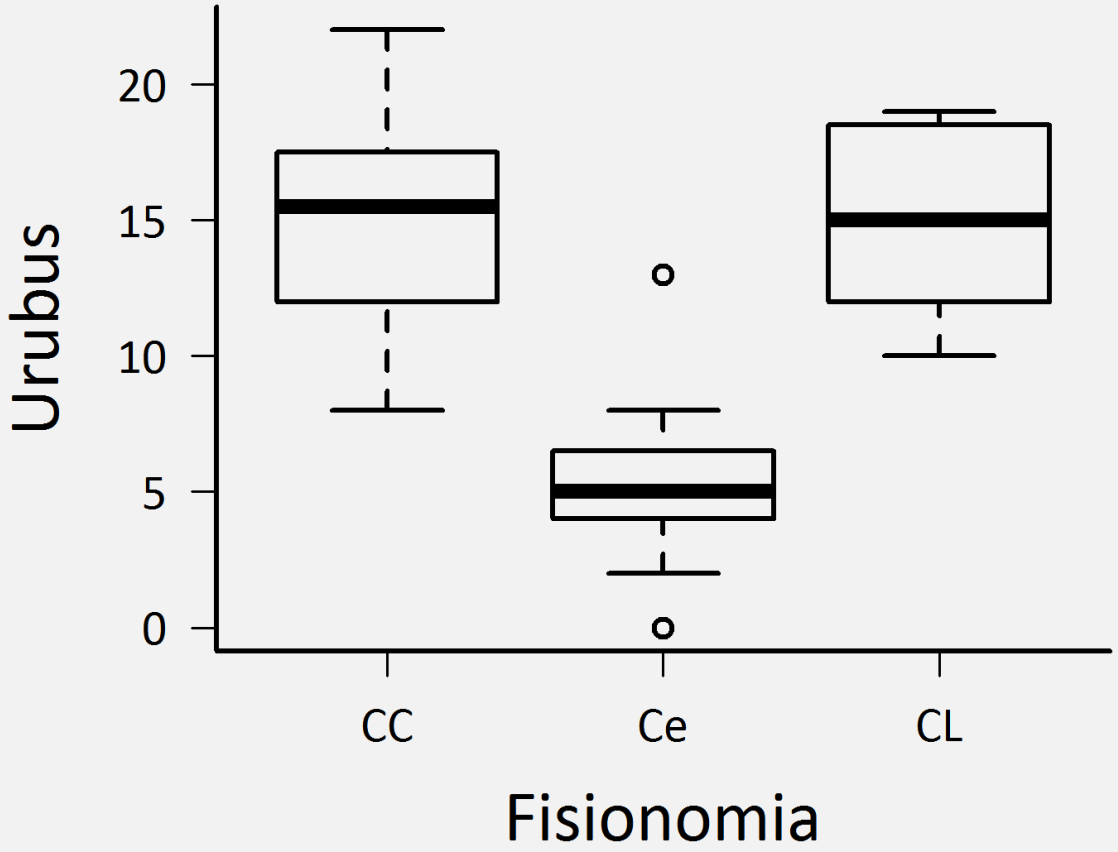
O retorno do boxplot

```
boxplot(urubu~fisionomia, data=aves,
        xlab="Fisionomia", ylab="Urubus")
```



O retorno do boxplot

```
plot(urubu~fisionomia, data=aves,
     xlab="Fisionomia", ylab="Urubus")
```



O poderoso pacote lattice

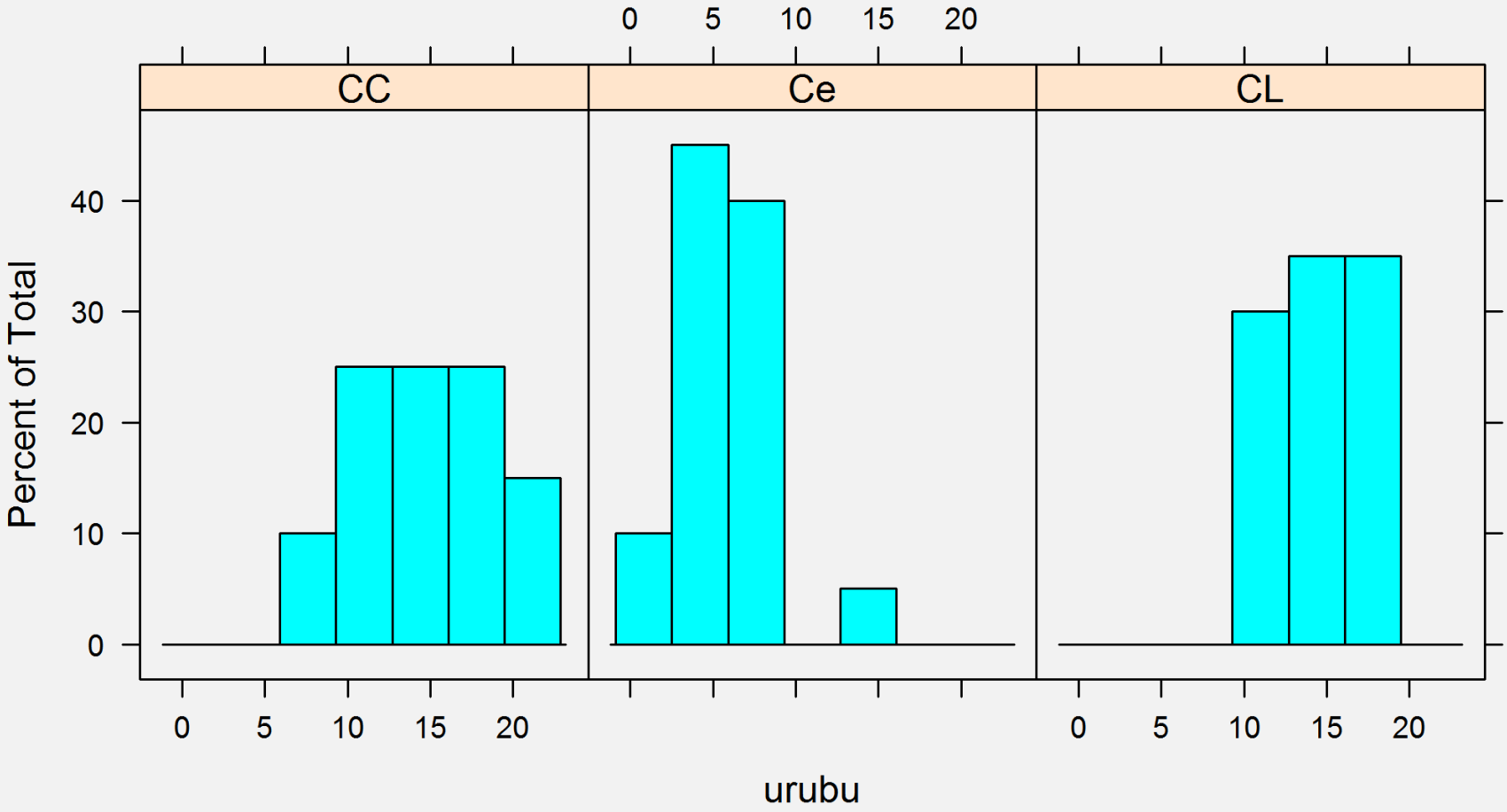
Fórmulas com o formato

$$y \sim x \mid z$$

“y em função de x separado por z”

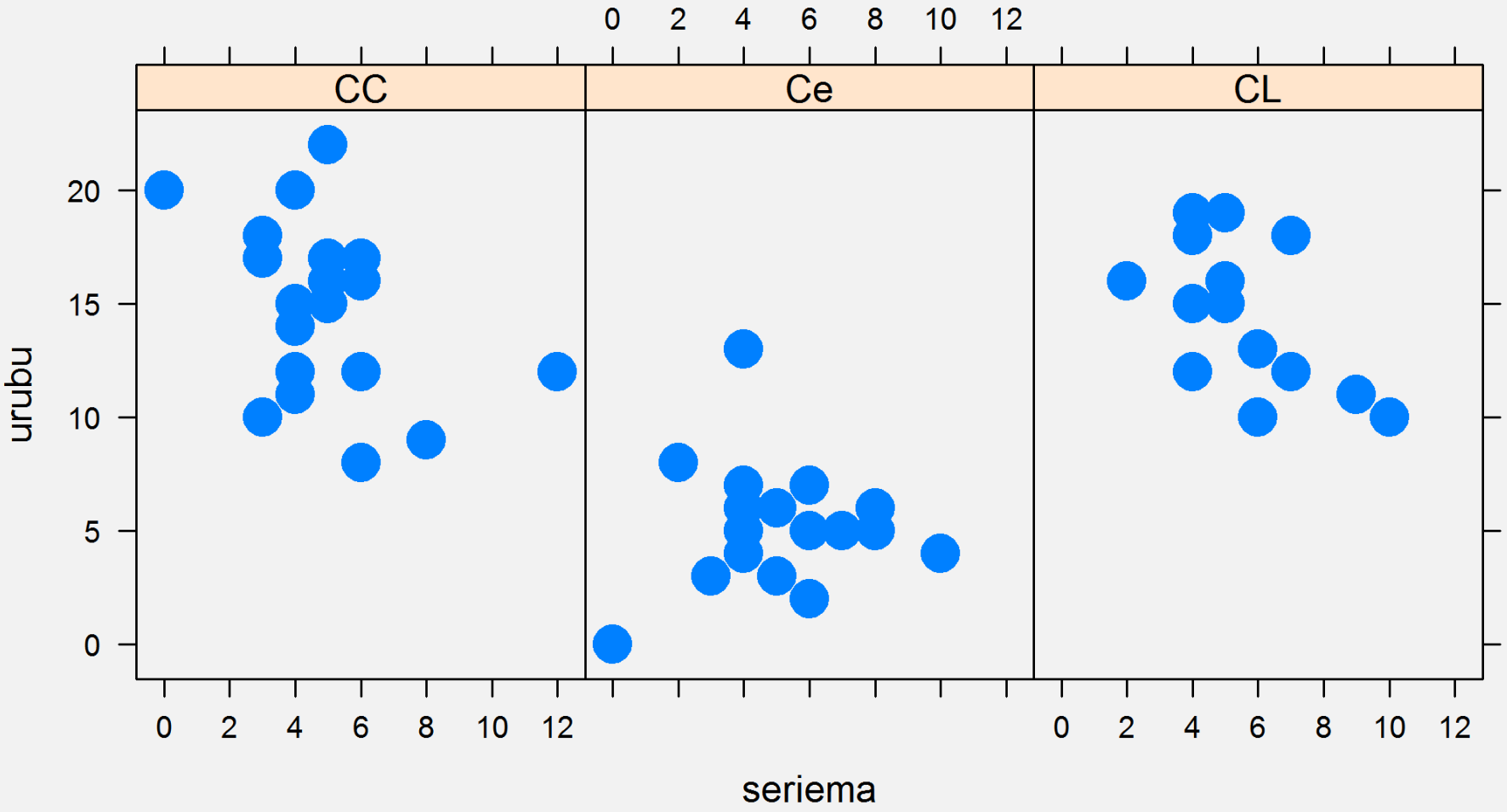
lattice::histogram()

histogram(~urubu|fisionomia, data=aves)

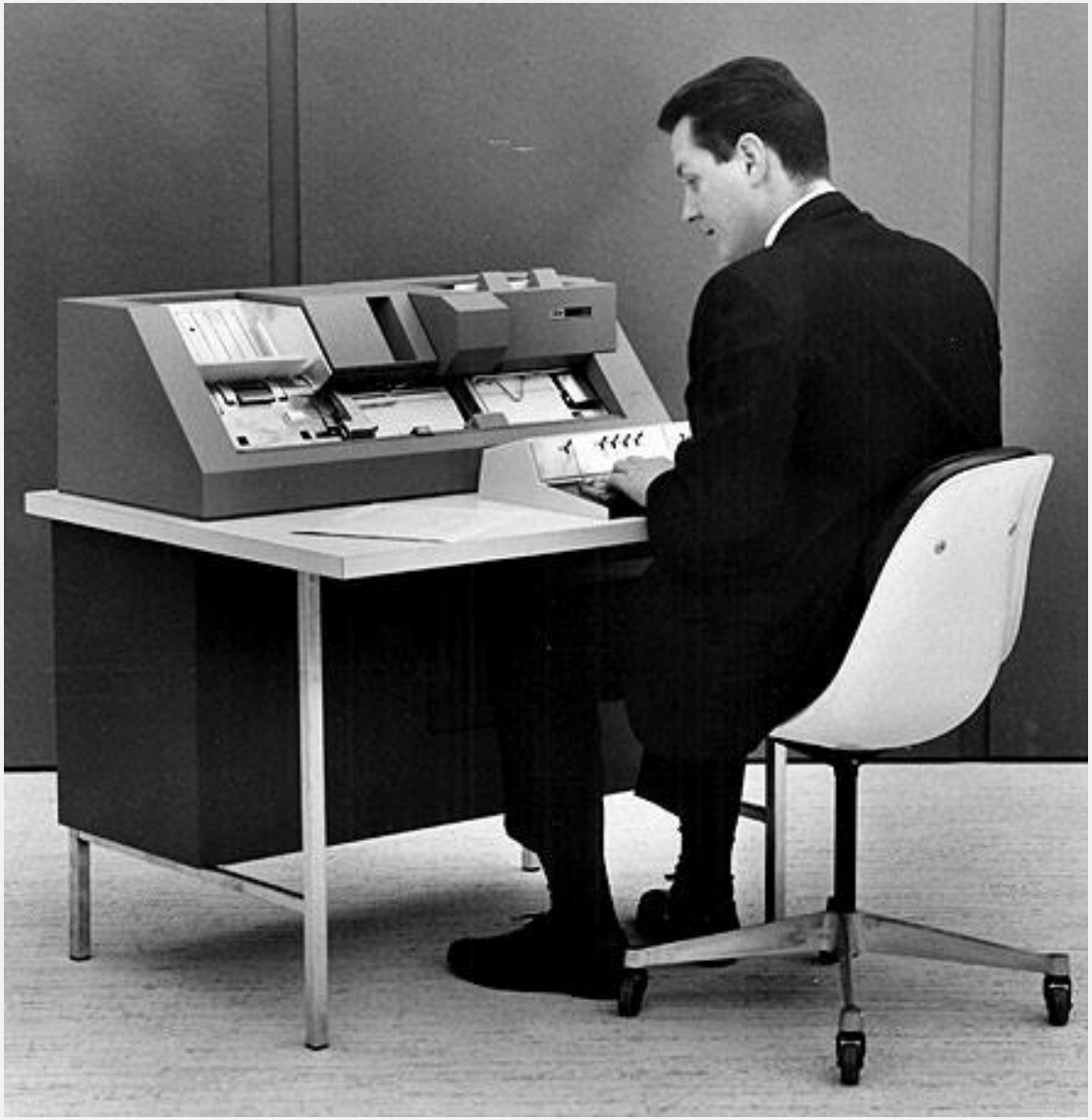


lattice::xyplot()

xyplot(urubu~seriema|fisionomia, data=aves)

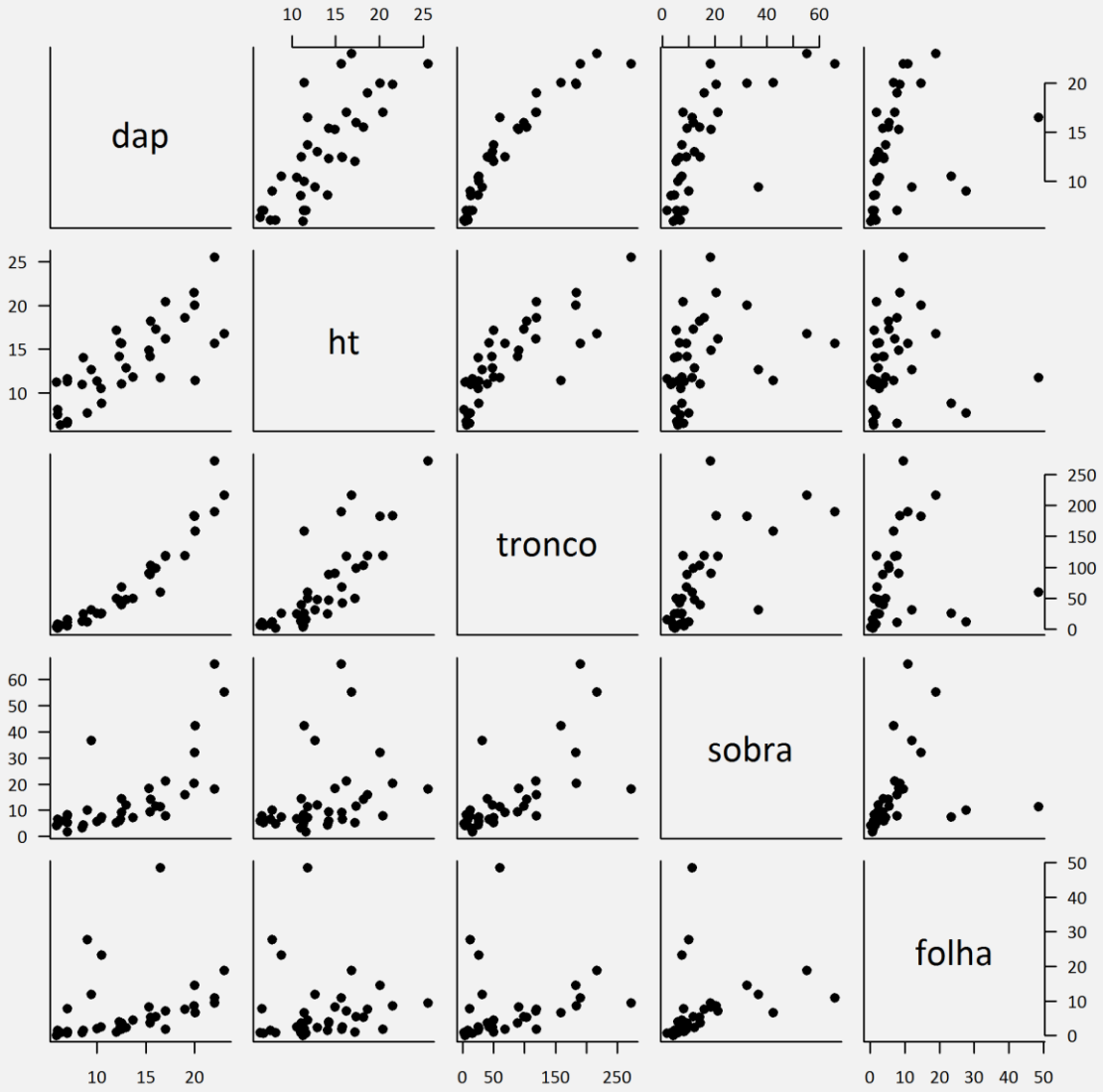


Bora para o R!



Explorando dados multivariados

Múltiplas variáveis contínuas



plot(esa[,4:8], pch=19)

Matriz de correlação automática

```
> cor(esa[,4:8])
              dap          ht    tronco     sobra       folha
dap    1.0000000 0.774516726 0.9407805 0.6863613 0.332482850
ht     0.7745167 1.000000000 0.8054810 0.3204422 0.007789986
tronco 0.9407805 0.805480985 1.0000000 0.6933458 0.217992398
sobra  0.6863613 0.320442164 0.6933458 1.0000000 0.291215948
folha  0.3324829 0.007789986 0.2179924 0.2912159 1.000000000
```

Lembrando...

Estamos conhecendo os dados

Matrizes de distância (ou similaridade)

Dados de árvores da ilha de BarroColorado

Carregando os dados

```
> library(vegan)
> data(BCI)
> str(BCI)
'data.frame':  50 obs. of  225 variables:
 $ Abarema.macradenia            : int  0 0 0 0 0 0 0 0 0 1 ...
 $ Vachellia.melanoceras         : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Acalypha.diversifolia         : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Acalypha.macrostachya         : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Adelia.triloba                : int  0 0 0 3 1 0 0 0 5 0 ...
 $ Aegiphila.panamensis          : int  0 0 0 0 1 0 1 0 0 1 ...
 $ Alchornea.costaricensis       : int  2 1 2 18 3 2 0 2 2 2 ...
 $ Alchornea.latifolia           : int  0 0 0 0 0 1 0 0 0 0 ...
 $ Alibertia.edulis              : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Allophylus.psilospermus       : int  0 0 0 0 1 0 0 0 0 0 ...
 $ Alseis.blackiana              : int  25 26 18 23 16 14 18 14 16 14
...
```

Matrizes de distância (ou similaridade)

```
barroDist = vegdist(x = BCI[1:10,],
method =  "jaccard")
```

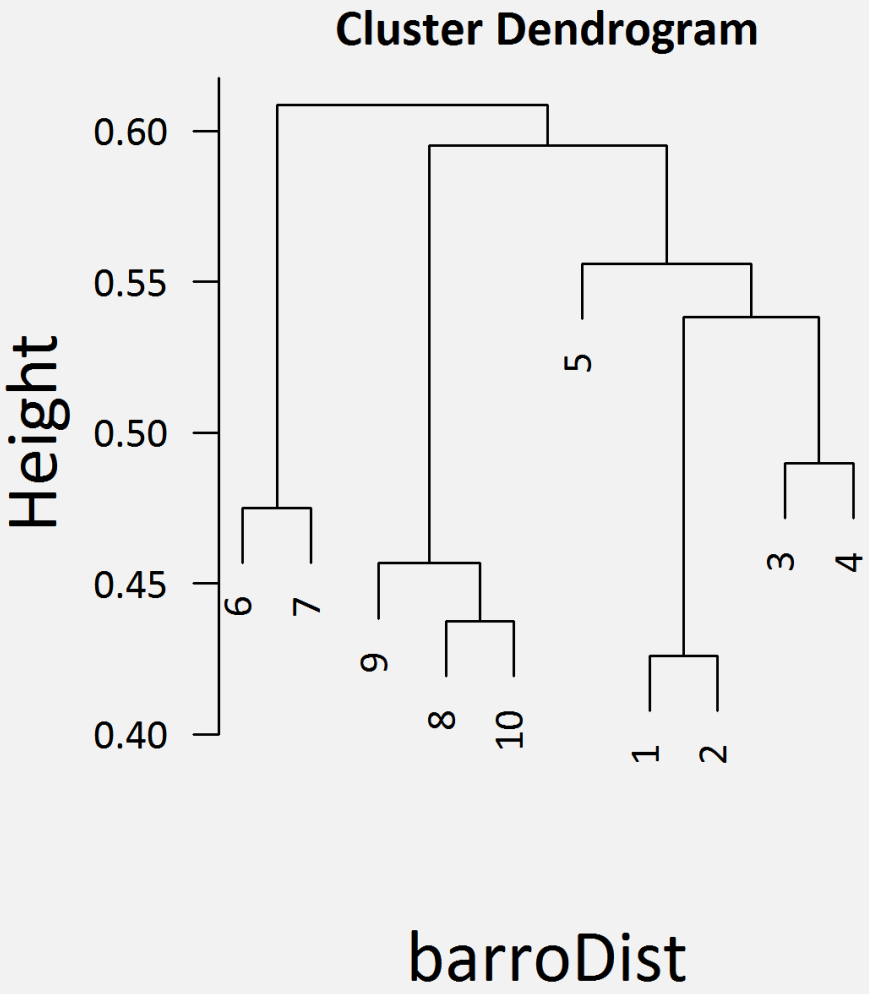
O índice de Jaccard mede a proporção de espécies em comum entre duas amostras/localidades

A matriz de similaridade

	1	2	3	4	5	6	7	8	9
2	0.426								
3	0.519	0.446							
4	0.538	0.479	0.490						
5	0.543	0.556	0.529	0.542					
6	0.545	0.522	0.547	0.595	0.547				
7	0.521	0.453	0.490	0.547	0.609	0.475			
8	0.510	0.431	0.448	0.478	0.542	0.526	0.479		
9	0.595	0.545	0.539	0.549	0.594	0.584	0.537	0.447	
10	0.548	0.502	0.457	0.475	0.556	0.599	0.559	0.438	0.457

Análise de agrupamento

plot(hclust(barroDist))



Análise multivariada *stricto sensu*

```
> pardal = read.table("ZuurSparrows.txt", sep="\t", dec=".", head=TRUE)
> head(pardal)
  wingcrd flatwing tarsus head culmen nalospi   wt bandstat
1    59.0     60.0   22.3 31.2   12.3    13.0  9.5        1
2    54.0     55.0   20.3 28.3   10.8     7.8 12.2        1
3    53.0     54.0   21.6 30.2   12.5     8.5 13.8        1
4    55.0     56.0   19.7 30.4   12.1     8.3 13.8        1
5    55.0     56.0   20.3 28.7   11.2     8.0 14.1        1
6    53.5     54.5   20.8 30.6   12.8     8.6 14.8        1
  initials Year Month Day Location SpeciesCode Sex Age
1        2 2002     9  19        4           1   0   2
2        2 2002    10   4        4           3   0   2
3        2 2002    10   4        4           3   0   2
4        8 2002     7  30        9           1   0   2
5        3 2002    10   4        4           3   0   2
6        7 2004     8   2        1           1   0   2
```

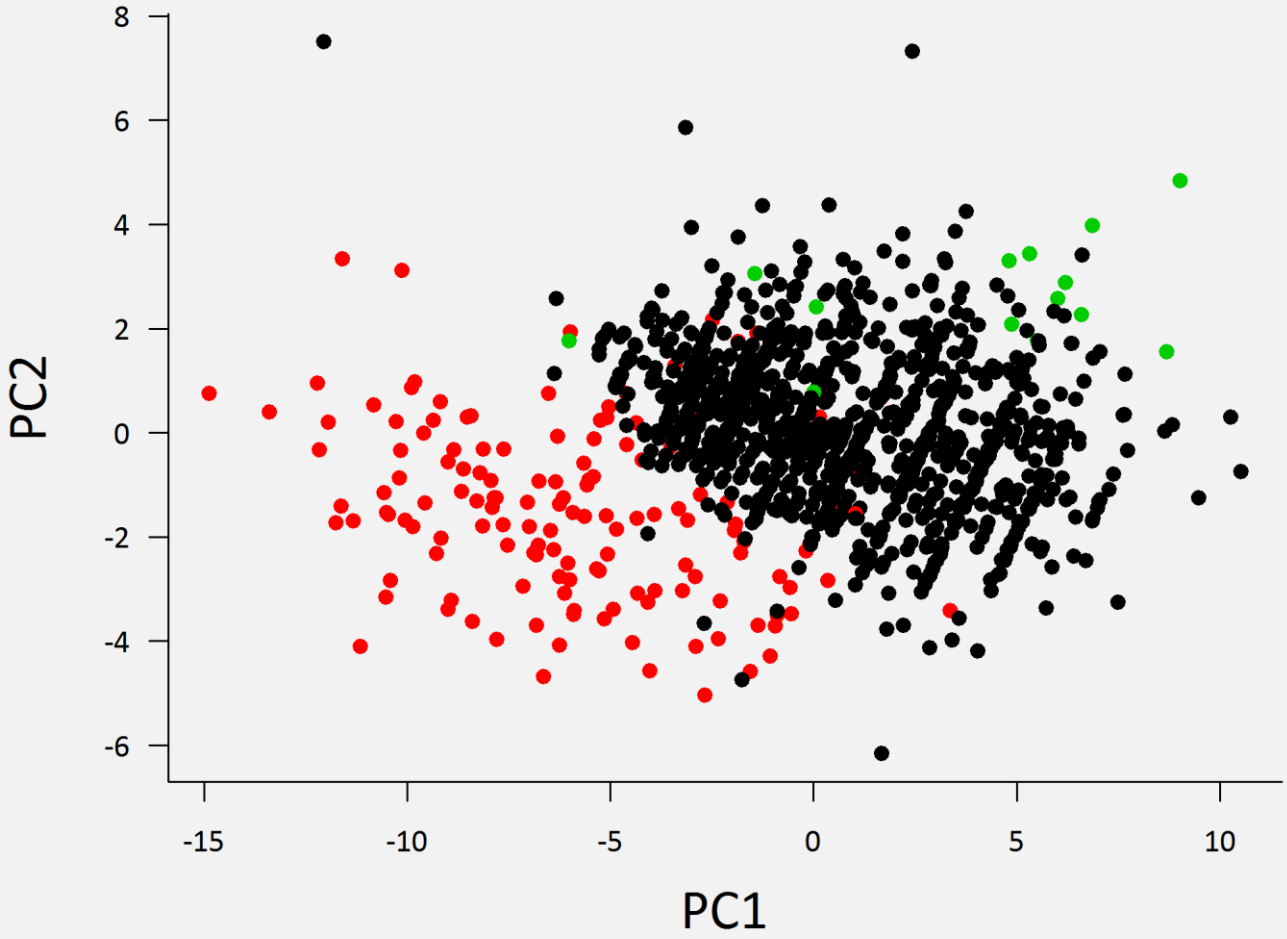
Dados de Zuur 2010

Nosso primeiro PCA!

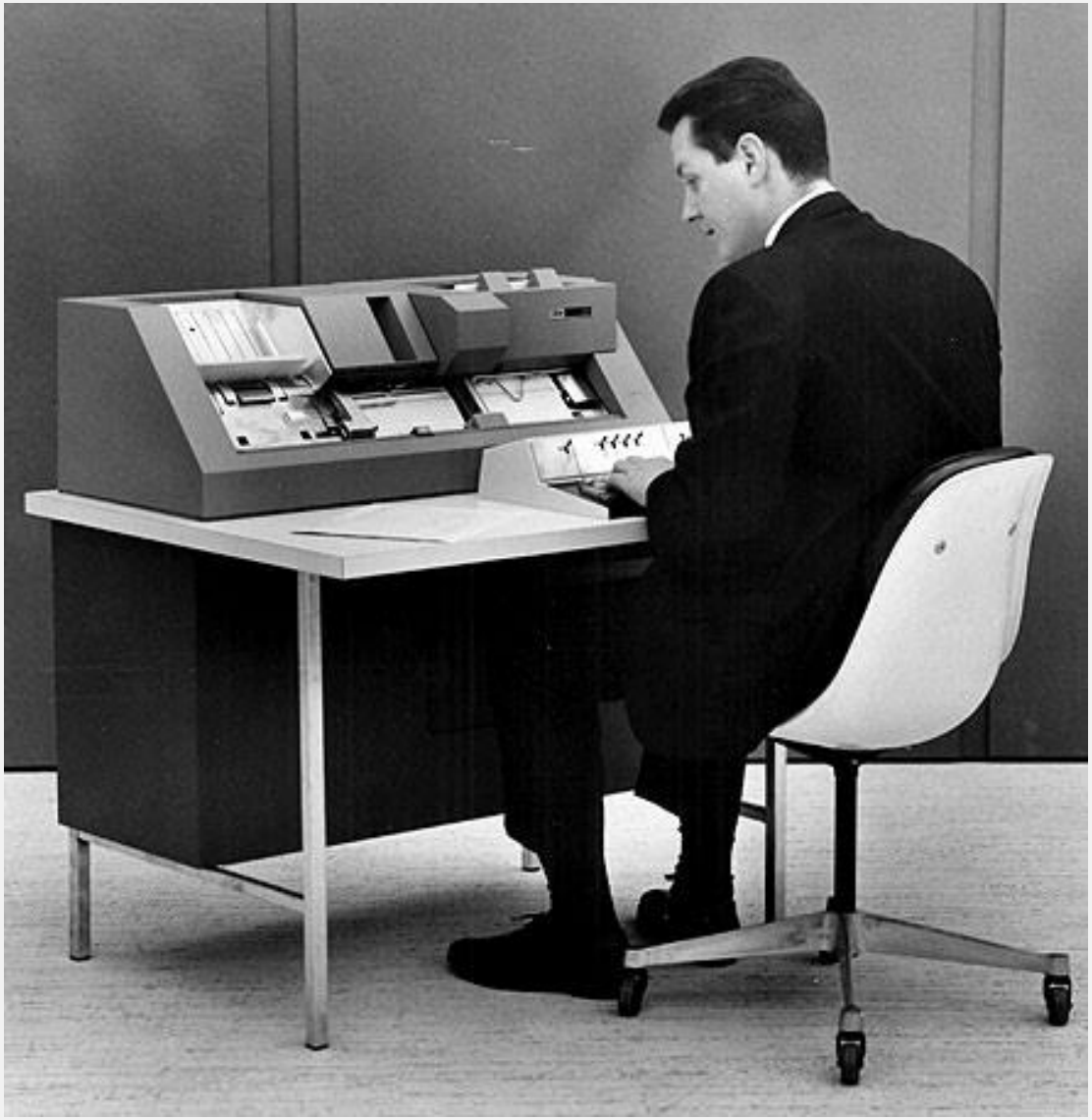
```
> pcapardal = princomp(pardal[,1:7])
> summary(pcapardal)
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4
Standard deviation     3.722563 1.5555237 1.00150901 0.72026550
Proportion of Variance 0.755477 0.1319138 0.05468231 0.02828279
Cumulative Proportion  0.755477 0.8873908 0.94207314 0.97035594
                           Comp.5     Comp.6      Comp.7
Standard deviation     0.53041049 0.43892098 0.264130229
Proportion of Variance 0.01533774 0.01050291 0.003803411
Cumulative Proportion  0.98569368 0.99619659 1.000000000
```

Vamos por no gráfico

```
plot(pcapardal$scores[,1], pcapardal$scores[,2],
     col=pardal$SpeciesCode, pch=19, xlab='PC1', ylab="PC2")
```



Vamos ao R!



Resumo

- Confira seus dados (NAs e erros de digitação)
- Verifique valores extremos e zeros
- Preste atenção na distribuição das variáveis
- Verifique a relação entre variáveis

Principais referências

Cleveland, W. 1993. **Visualizing data**. Hobart Press.

Ellison, A. M. 1993. Exploratory data analysis and graphic display. In: Scheiner, S. M. (ed.), ***Design and analysis of ecological experiments***. Chapman & Hall, pp. 14-45.

Zuur, A., Ieno, E. N. & Smith G. M. 2007. **Analysing ecological data**. Springer. Capítulo 4.

Morgenthaler, S. (2009). Exploratory data analysis. **Wiley Interdisciplinary Reviews: Computational Statistics,** 1:33-44.

Zuur, A., Ieno, E. N. & Elphick, C. S. 2010. A protocol for data exploration to avoid common statistical problems. **Methods in Ecology & Evolution**, 1: 3-14.

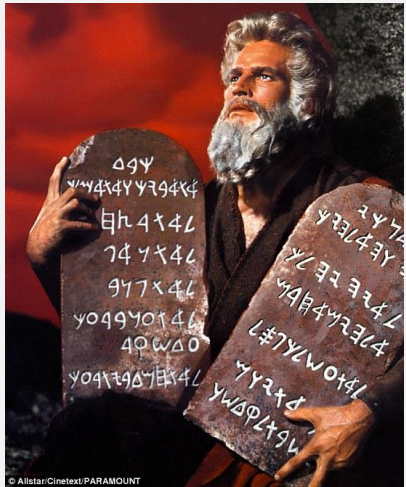
Take home messages

Conheça bem seus dados

Jamais se esqueça da família (apply)

A análise exploratória só é limitada pela imaginação e habilidade do analista

(não se reprima!)



Nos vemos no plantão!