MAC 113 — Introdução à Ciência da Computação

Aula 26

Nelson Lago

1°/2025

Previously on MAC113...

Dataframes

- **Em geral, as operações vetoriais que fazemos em dataframes/tibbles processam as colunas**
  - Como cada coluna é um vector, sabemos que os elementos são todos do mesmo tipo
  - Em geral, coisas como `max()`, `mean()`, `sum()` etc., fazem sentido com as colunas
- **Para processar as linhas, podemos extrair uma linha específica por sua coordenada:**
  - `df[3,]`
- **ou, para processar todas, fazer um laço:**
  - `for (i in seq_len(nrow(df))) { df[i,] }`
- **ou usar `paste()`**
- **...**

Dataframes

Dataframes/tibbles: Como vivem? Onde moram? De que se alimentam?

- **Não tem muita graça digitar os dados de um dataframe/tibble como parte do programa**
- **O mais razoável é carregar de um arquivo**
- **Dataframes são tabelas $\longrightarrow$ o arquivo deve representar uma tabela**
- **O formato mais comumente usado é o CSV (comma-separated values)**

Nome	matemática	português	física	história
Alan Turing	9.7	1.4	9.2	8.7
Ada Lovelace	9.8	1.2	10.0	9.2

Dataframes

```
Nome,matemática,português,física,história
Alan Turing,9.7,1.4,9.2,8.7
Ada Lovelace,9.8,1.2,10.0,9.2
```

```
Nome          ,matemática,português,física,história
Alan Turing   ,9.7       ,1.4      ,9.2   ,8.7
Ada Lovelace  ,9.8       ,1.2      ,10.0  ,9.2
```

- **CSV: "Comma" pode ser qualquer coisa (espaços, tabs...)**

```
Nome;matemática;português;física;história
Alan Turing;9,7;1,4;9,2;8,7
Ada Lovelace;9,8;1,2;10,0;9,2
```

Dataframes

- **E como ler um arquivo desses?**

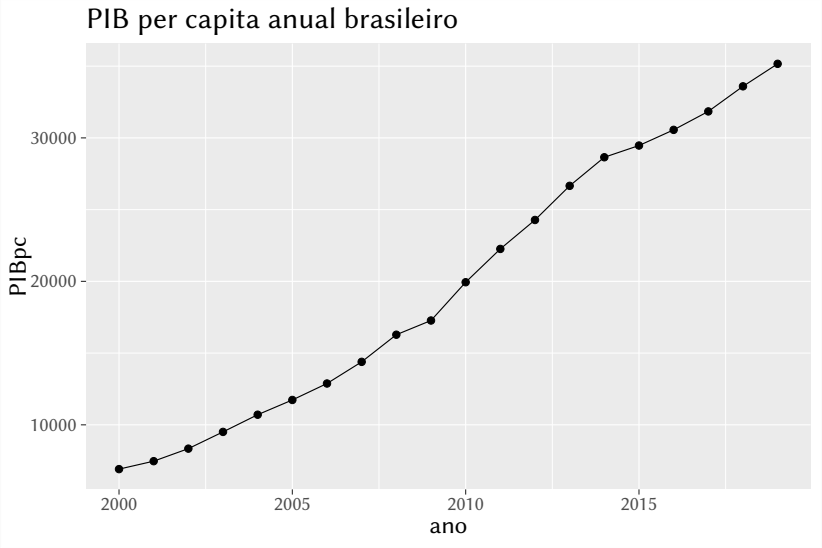
```
endereço <- 'http://www.ime.usp.br/~lago/dadinhos/renda_vs_inflacao.csv'
#df <- read.table(endereço, sep=",", header=TRUE) # dataframe
library(readr)
df <- read_delim(endereço, delim=",", col_types = "idd")
names(df)[2] <- "PIBpc"
print(head(df))
```

```
# A tibble: 6 × 3
    ano  PIBpc  IPCA
  <int>  <dbl> <dbl>
1  2000  6913.   7
2  2001  7467    6.8
3  2002  8341.   8.4
4  2003  9507.  14.7
5  2004 10706    6.6
6  2005 11734.   6.9
```

Dataframes

```
endereço <- 'http://www.ime.usp.br/~lago/dadinhos/renda_vs_inflacao.csv'
#df <- read.table(endereço, sep=",", header=TRUE) # dataframe
library(readr)
df <- read_delim(endereço, delim=",", col_types = "idd")
names(df)[2] <- "PIBpc"
p <- ggplot(df, aes(x=ano, y=PIBpc)) +
         geom_line() +
         geom_point() +
         labs(title="PIB per capita anual brasileiro")
print(p)
```

Dataframes


PIB per capita anual brasileiro
PIBpc
30000
20000
10000
2000
2005
2010
2015
ano

Recortes com filtros

- **Dado um dataframe `df` (ou um vector/lista), podemos fazer um "recorte" usando coordenadas: `df[linhas,colunas]`**
  - `linhas` e `colunas` podem ser valores únicos ou vectors
- **Também podemos usar um vector do tipo `LOGICAL`**

```
vec <- 1:5
cat(vec[c(TRUE, FALSE, FALSE, TRUE, FALSE)], "\n")
1 4
grandes <- vec > 3
cat(grandes, "\n")
FALSE FALSE FALSE TRUE TRUE
cat(vec[grandes], "\n")
4 5
cat(vec[vec > 3], "\n")
4 5
```

Recortes com filtros

```
library(tibble)
df <- tibble(nome=c("Fulano", "Ciclano", "Beltrano"), idade=c(17, 25, 19))
cat("Idade do Fulano:", unlist(df[df$nome == "Fulano","idade"]), "\n")
```

Idade do Fulano: 17

```
library(tibble)
df <- tibble(nome=c("Fulano", "Ciclano", "Beltrano"), idade=c(17, 25, 19))
cat("Maiores de idade:", unlist(df[df$idade>=18,]), "\n")
maiores <- df[df$idade>=18,]
cat("Maiores de idade:", paste(maiores$nome, ": ", maiores$idade, sep=""), sep="\n")
```

Maiores de idade: Ciclano Beltrano 25 19
Maiores de idade:
Ciclano: 25
Beltrano: 19

Outras coisas sobre R e dataframes

- **Para salvar um dataframe em um arquivo:**
  - `write_delim(df, file="blah.csv", delim=",")`
- **Para salvar o último gráfico exibido com ggplot2 em um arquivo:**
  - `ggsave("arquivo.pdf", width=6, height=4)` (polegadas)
- **Para ler diretamente arquivos do excel ou libreoffice:**
  - biliotecas `readxl` e `readODS` (*mas prefira usar CSV!*)
- **Bibliotecas muito úteis e populares:**
  - `purrr` (operações vetoriais com lists/vectors)
  - `dplyr` (operações vetoriais com dataframes/tibbles)
  - `stringr` (operações com textos)
    - » *carregue todas (além de `ggplot2`, `tibble`...) com `library(tidyverse)`*
- **Para acessar a documentação**
  - `?nome-da-função` (sobre a função)
  - `vignette("package")` (sobre a package)
  - `??alguma-coisa` (busca "alguma-coisa" em toda a documentação)

And now for more of the same

Exoplanetas

- **Dados sobre exoplanetas science.nasa.gov/exoplanets/exoplanet-catalog/**
- **Em formato tabular "mastigado"**

(Adaptado de github.com/mwaskom/seaborn-data/blob/master/planets.csv)

number	period	mass	distance
1	269.3	7.1	77.4
1	874.8	2.2	56.95
1	763.0	2.6	19.84
…			

-
```
   number  Number of planets in the system
   period  Orbital period in Earth days
     mass  Mass of the planet in Jupiter masses
 distance  Distance from Earth in light-years
```

Exoplanetas

- www.ime.usp.br/~lago/dadinhos/planets.csv
- Qual o maior número de planetas em um sistema?
- Qual a distância do planeta mais próximo?
- Quais são a massa e o período orbital médios?

```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    cat("O maior número de planetas em um sistema é", max(planetas$number), "\n")
    cat("A distância até o planeta mais próximo é", min(planetas$distance), "anos-luz\n")
    cat("A massa média dos planetas é", mean(planetas$mass), "\n")
    cat("O período orbital médio dos planetas é", mean(planetas$period), "\n")
}
main()
```

Intermezzo

ggplot2

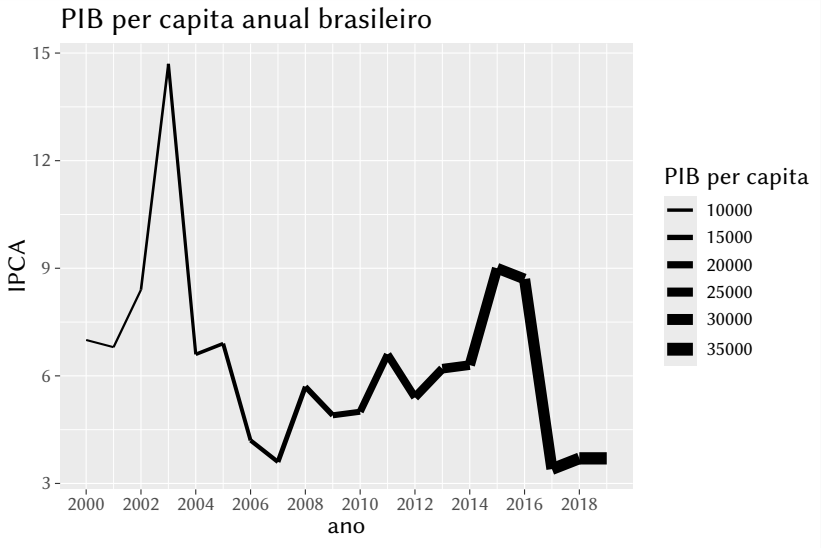
- **Em um gráfico, representamos dados com formas geométricas (pontos, linhas, caixas, barras...) e seus atributos estéticos (posição, forma, tamanho, cor...)**
  - Em um gráfico de linha, a forma é a linha e o atributo estético usado geralmente é a sua posição no plano cartesiano (coordenadas x, y)
- **Gráficos com `ggplot2` são construídos em camadas**
  - Adicionamos camadas com "+"
  - A primeira "camada" é a área (vazia) do gráfico: `ggplot(df)`
    - » *(um "atalho" comum é acrescentar o dataframe com os dados aqui também)*
  - As próximas camadas são as formas geométricas: `geom_point()`, `geom_line()` etc.
    - » *E, para cada uma, incluímos as definições dos atributos estéticos:* `aes()`
  - As próximas camadas definem os demais aspectos visuais, como o sistema de coordenadas, as escalas dos eixos, títulos, legendas etc.

ggplot2

- **Para cada forma geométrica, é preciso definir como o dado é representado por aquela forma**
  - No caso de linhas, pontos etc., geralmente queremos representar os dados pela posição das linhas/pontos no sistema de coordenadas:
    `geom_line(aes(x=blah, y=bleh))`
  - Mas a espessura e a cor da linha também podem ser usadas para representar algum dado:
    `geom_line(aes(x=blah, y=bleh, color=blih, linewidth=bloh))`

ggplot2

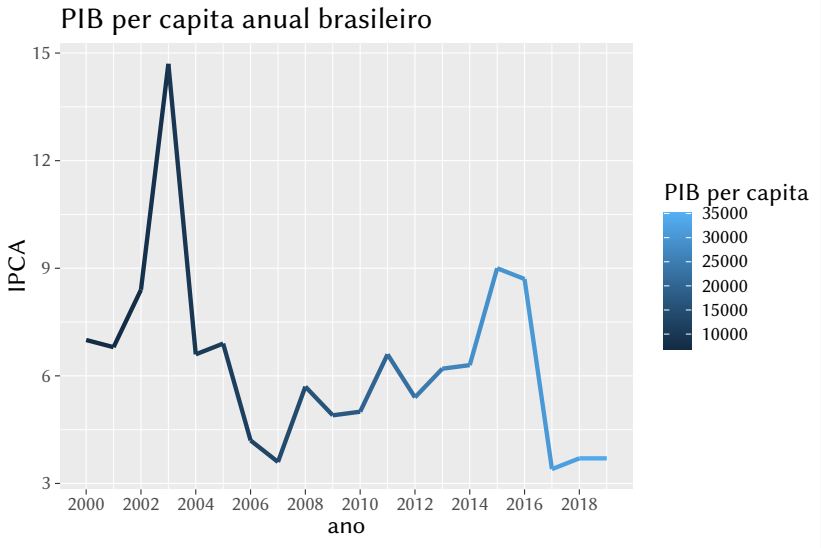
```
library(tidyverse)
endereço <- 'http://www.ime.usp.br/~lago/dadinhos/renda_vs_inflacao.csv'
df <- read_delim(endereço, delim=",", col_types = "idd")
names(df)[2] <- "PIBpc"
p <- ggplot(df) +
  geom_line(aes(x=ano, y=IPCA, linewidth=PIBpc)) +
  labs(title="PIB per capita anual brasileiro",
       linewidth="PIB per capita") +
  scale_x_continuous(breaks=seq(2000, 2019, 2))
print(p)
```


PIB per capita anual brasileiro
IPCA
15
12
9
6
3
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
ano
PIB per capita
10000
15000
20000
25000
30000
35000

ggplot2

```
library(tidyverse)
endereço <- 'http://www.ime.usp.br/~lago/dadinhos/renda_vs_inflacao.csv'
df <- read_delim(endereço, delim=",", col_types = "idd")
names(df)[2] <- "PIBpc"
p <- ggplot(df) +
  geom_line(aes(x=ano, y=IPCA, color=PIBpc), linewidth=2) +
  labs(title="PIB per capita anual brasileiro",
       color="PIB per capita") +
  scale_x_continuous(breaks=seq(2000, 2019, 2))
print(p)
```

ggplot2


PIB per capita anual brasileiro
IPCA
15
12
9
6
3
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
ano
PIB per capita
35000
30000
25000
20000
15000
10000

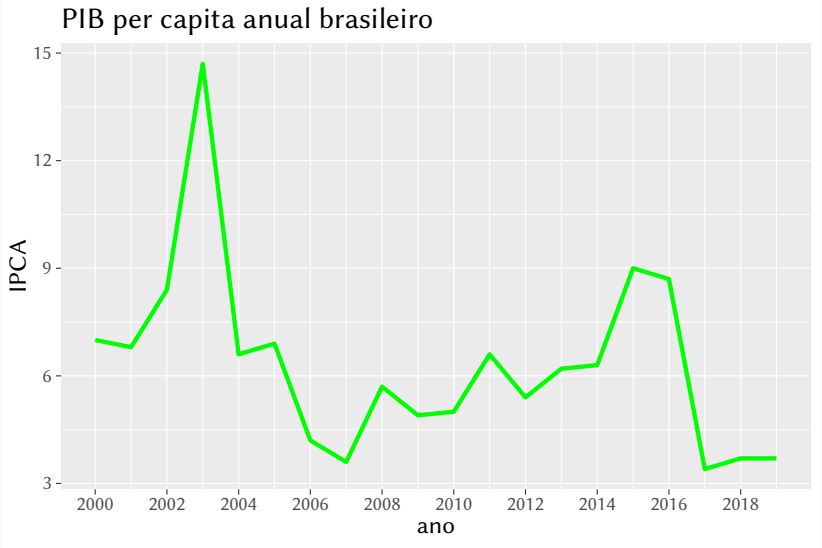
ggplot2

- **Para cada forma geométrica, é preciso definir como o dado é representado por aquela forma**
  - No caso de linhas, pontos etc., geralmente queremos representar os dados pela posição das linhas/pontos no sistema de coordenadas:
    `geom_line(aes(x=blah, y=bleh))`
  - Mas a espessura e a cor da linha também podem ser usadas para representar algum dado:
    `geom_line(aes(x=blah, y=bleh, color=blih, linewidth=bloh))`
  - Isso é totalmente diferente de
    `geom_line(aes(x=blah, y=bleh), color="green", linewidth=2))`

ggplot2

```
library(tidyverse)
endereço <- 'http://www.ime.usp.br/~lago/dadinhos/renda_vs_inflacao.csv'
df <- read_delim(endereço, delim=",", col_types = "idd")
names(df)[2] <- "PIBpc"
p <- ggplot(df) +
  geom_line(aes(x=ano, y=IPCA), color="green", linewidth=2) +
  labs(title="PIB per capita anual brasileiro") +
  scale_x_continuous(breaks=seq(2000, 2019, 2))
print(p)
```

ggplot2


PIB per capita anual brasileiro
IPCA
15
12
9
6
3
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
ano

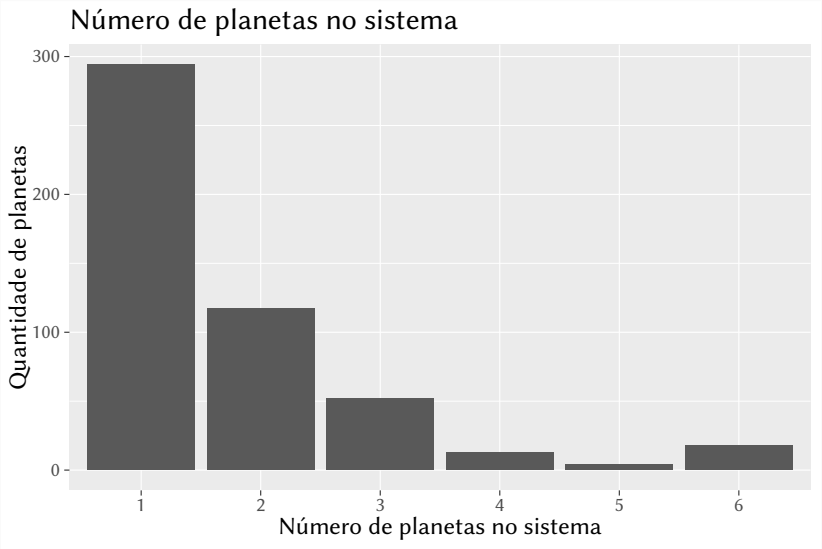
ggplot2

- **Para cada forma geométrica, é preciso definir como o dado é representado por aquela forma**
  - No caso de linhas, pontos etc., geralmente queremos representar os dados pela posição das linhas/pontos no sistema de coordenadas:
    `geom_line(aes(x=blah, y=bleh))`
  - Mas a espessura e a cor da linha também podem ser usadas para representar algum dado:
    `geom_line(aes(x=blah, y=bleh, color=blih, linewidth=bloh))`
  - Isso é totalmente diferente de
    `geom_line(aes(x=blah, y=bleh), color="green", linewidth=2)`
    - » *Veja todos os atributos possíveis com* `vignette('ggplot2-specs')`

Exoplanetas

- **Quantos planetas há no sistema de cada planeta? (barras)**

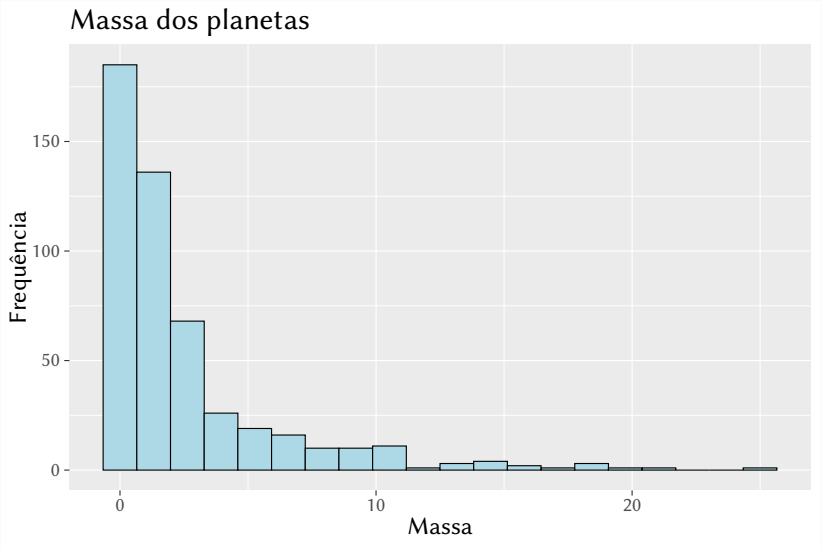
```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    quantidades <- unique(planetas$number)
    vizinhança <- integer(length(quantidades))
    for (i in seq_len(length(quantidades))) {
        q <- quantidades[i]
        vizinhança[i] <- nrow(planetas[planetas$number == q,])
    }
    p <- ggplot(NULL, aes(x=quantidades, y=vizinhança)) +
         geom_col() +
         labs(title="Número de planetas no sistema",
              x="Número de planetas no sistema",
              y="Quantidade de planetas") +
         scale_x_discrete(limits=as.character(1:6))
    print(p)
}
main()
```


Número de planetas no sistema
Quantidade de planetas
300
200
100
0
1
2
3
4
5
6
Número de planetas no sistema

Exoplanetas

- **Qual é a distribuição da massa dos planetas? (histograma)**

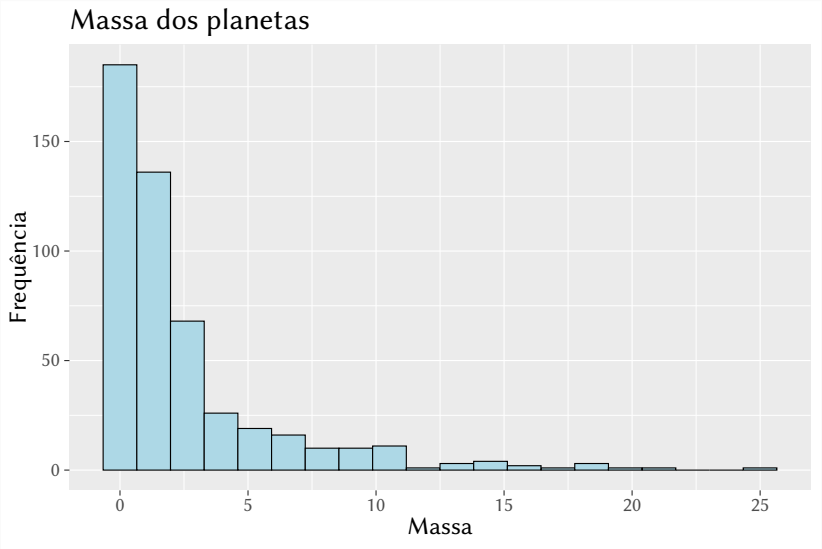
```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    p <- ggplot(planetas) +
         geom_histogram(aes(  mass), bins=20, color='black', fill='lightblue') +
         labs(title="Massa dos planetas", x="Massa", y="Frequência")
    print(p)
}
main()
```


Massa dos planetas
Frequência
150
100
50
0
0
10
20
Massa

Exoplanetas

- **Qual é a distribuição da massa dos planetas? (histograma)**

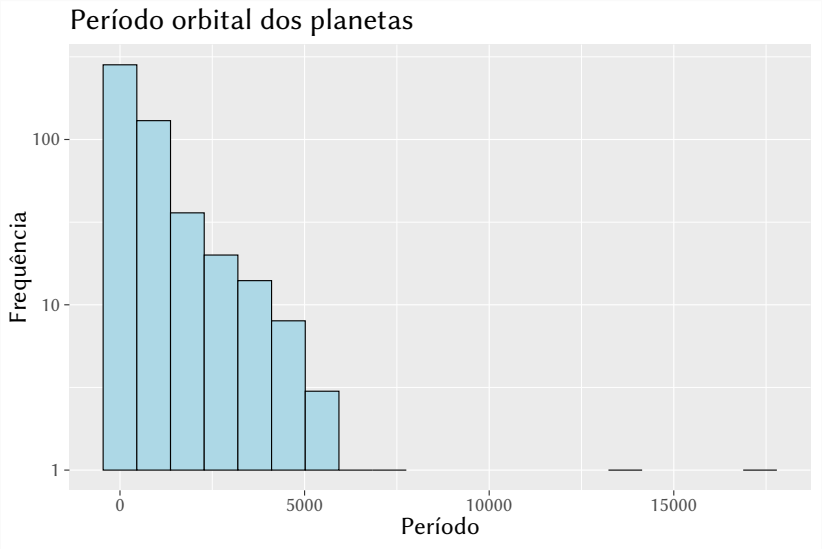
```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    p <- ggplot(planetas) +
         geom_histogram(aes(mass), bins=20, color='black', fill='lightblue') +
         labs(title="Massa dos planetas", x="Massa", y="Frequência") +
         scale_x_continuous(breaks=seq(0,50,5))
    print(p)
}
main()
```


Massa dos planetas
Frequência
150
100
50
0
0
5
10
15
20
25
Massa

Exoplanetas

- **Qual é a distribuição do período orbital dos planetas? (histograma)**

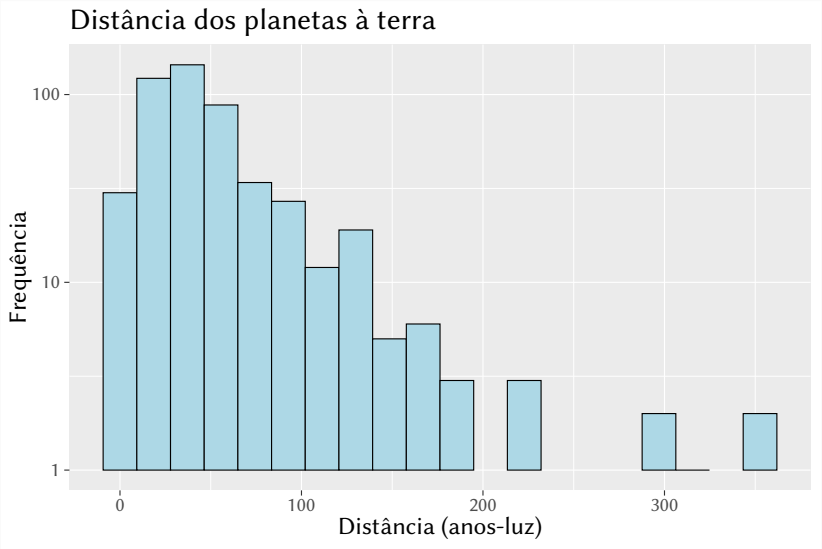
```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    p <- ggplot(planetas) +
         geom_histogram(aes(period), bins=20, color='black', fill='lightblue') +
         labs(title="Período orbital dos planetas", x="Período", y="Frequência") +
         scale_y_continuous(transform = 'log10')
    print(p)
}
main()
```


Período orbital dos planetas
Frequência
100
10
1
0
5000
10000
15000
Período

Exoplanetas

- **Qual é a distribuição da distância dos planetas? (histograma)**

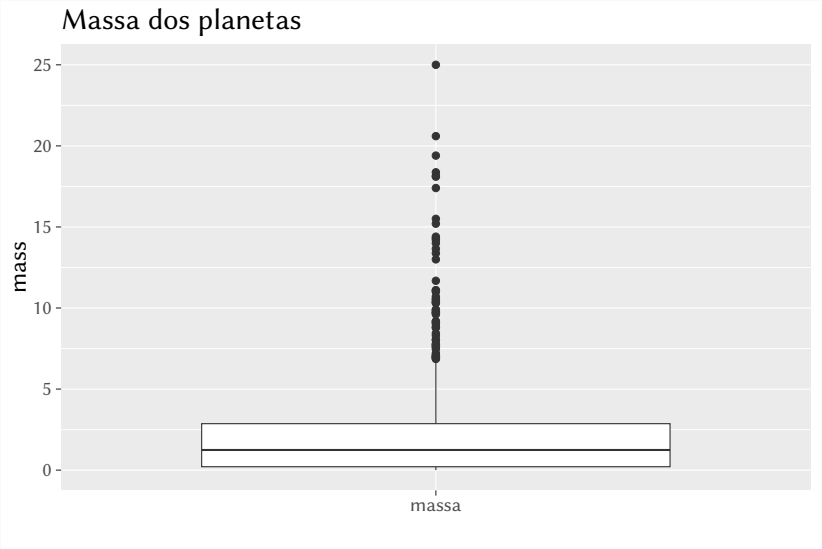
```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    p <- ggplot(planetas) +
         geom_histogram(aes(distance), bins=20, color='black', fill='lightblue') +
         labs(title="Distância dos planetas à terra",
              x="Distância (anos-luz)", y="Frequência") +
         scale_y_continuous(transform = 'log10')
    print(p)
}
main()
```


Distância dos planetas à terra
Frequência
100
10
1
0
100
200
300
Distância (anos-luz)

Exoplanetas

- **Qual é a distribuição da massa dos planetas? (boxplot)**

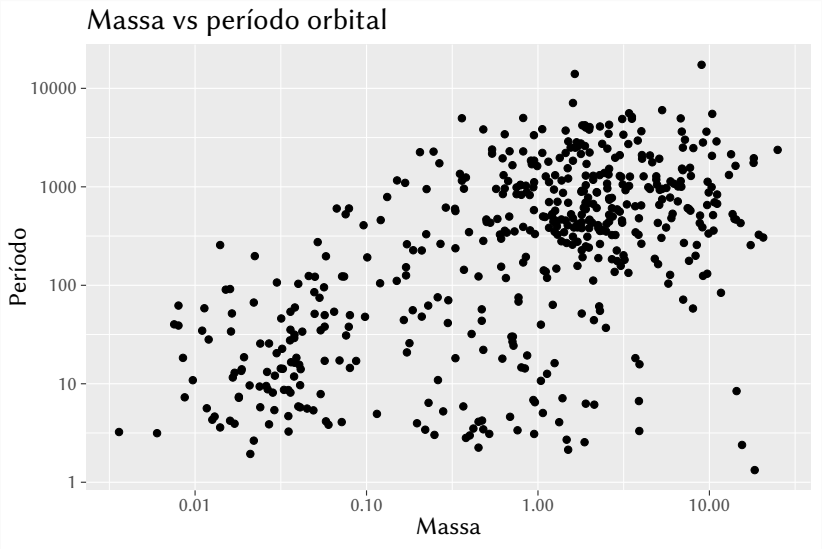
```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    p <- ggplot(planetas) +
         geom_boxplot(aes(x="massa", y=mass)) +
         labs(title="Massa dos planetas", x="")
    print(p)
}
main()
```


Massa dos planetas
mass
25
20
15
10
5
0
massa

Exoplanetas

- **Existe relação entre a massa dos planetas e seu período orbital? (dispersão ou *scatter*)**

```
library(tidyverse)
main <- function() {
    arq <- 'http://www.ime.usp.br/~lago/dadinhos/planets.csv'
    planetas <- read_delim(arq, delim=",", show_col_types = FALSE)
    p <- ggplot(planetas) +
         geom_point(aes(x=mass, y=period)) +
         labs(title="Massa vs período orbital", x="Massa", y="Período") +
         scale_x_continuous(transform = 'log10') +
         scale_y_continuous(transform = 'log10')
    print(p)
}
main()
```


Massa vs período orbital
Período
10000
1000
100
10
1
0.01
0.10
1.00
10.00
Massa