

R para Ciência de Dados I



Sobre a Curso-R

A empresa



Nossos cursos

Programação em R

Introdução à Programação em R

R para Ciência de Dados I

R para Ciência de Dados II

Pacotes

Introdução ao R com C++

Visualização de dados

Relatórios e Visualização de Dados

Dashboards

Deploy

Modelagem

Regressão linear

Machine Learning

XGBoost

Deep Learning

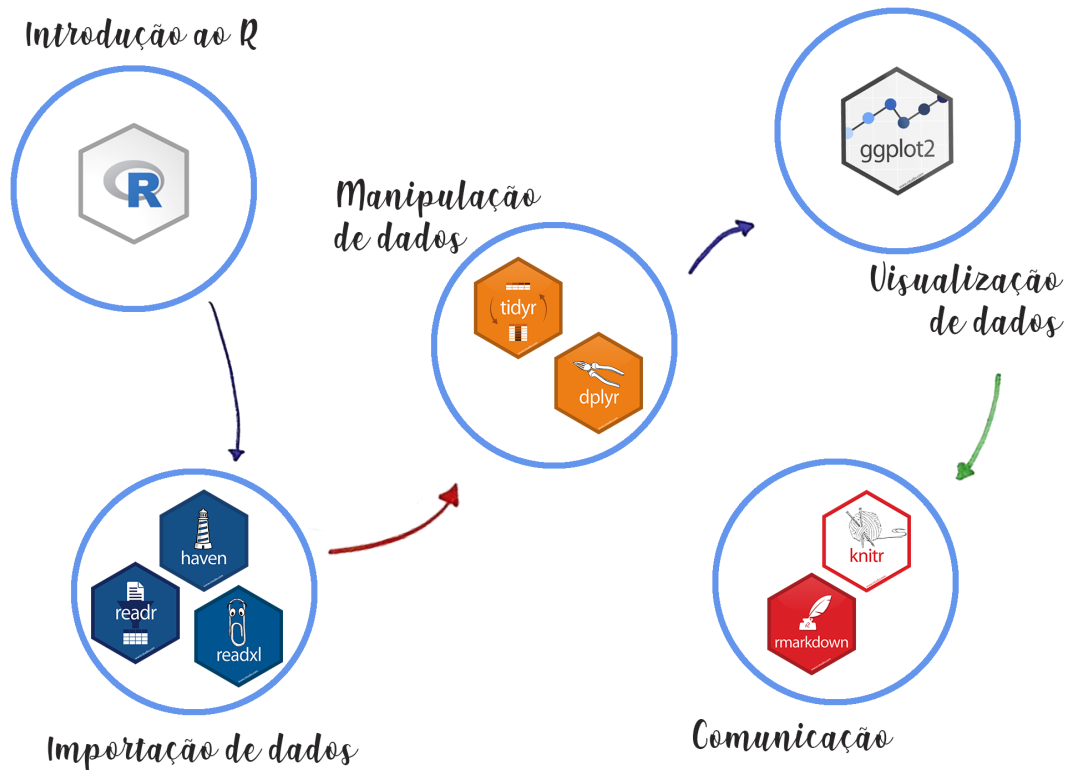
Extração de dados

Faxina de Dados

Web scraping

Sobre o curso

Conteúdo



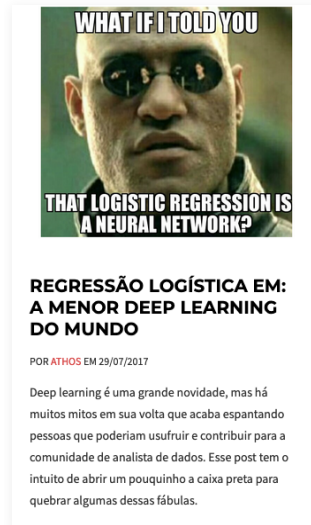
Material

Nosso livro Ciência de Dados em R:

<https://livro.curso-r.com/>

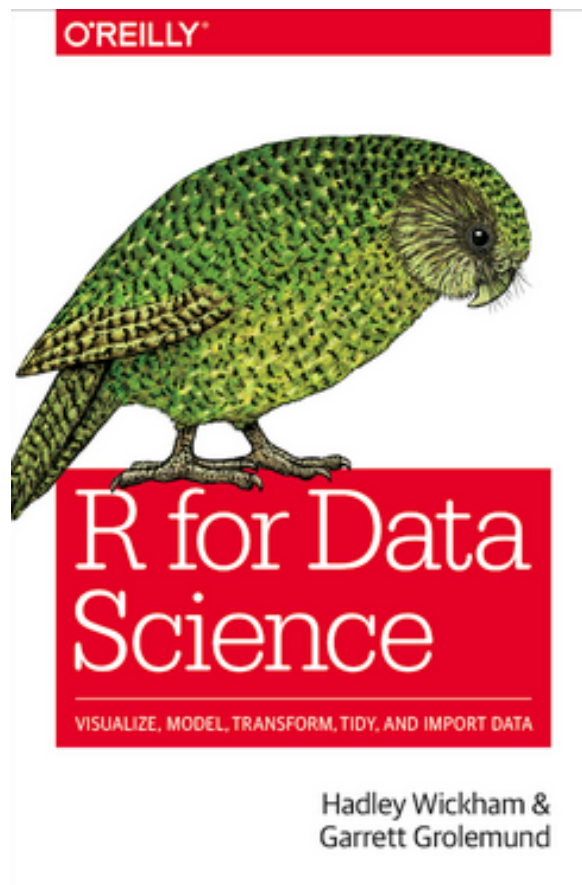
Também temos um blog:

<http://curso-r.com/blog/>



Referência

Disponível online e gratuitamente em [Inglês](#) e [Espanhol](#).



Como estudar?

Pratique

- Não se preocupe com estilos de estudo. Use o que te deixar mais confortável.
- Estude um pouco todo dia. Se você tem apenas uma hora para estudar na semana, faça 3 sessões de 20 minutos em dias diferentes.
- Se teste! Coloque em prática o que você aprendeu resolvendo exercícios ou inserindo o R no seu dia-a-dia.

Referência

Nahurhodo 205 - Powerpoint é útil para a aprendizagem?

Tirando dúvidas

- Não existe dúvida idiota.
- Nem sempre é trivial fazer a pergunta certa para que outra pessoa esclareça a sua dúvida. Neste curso, **vamos mostrar melhores práticas na hora de fazer perguntas sobre programação.**
- Fora do horário de aula ou monitoria:
 - perguntas gerais sobre o curso deverão ser feitas no Classroom.
 - perguntas sobre R, principalmente as que envolverem código, deverão ser enviadas no [nosso discourse](#).
- [Veja aqui dicas de como fazer uma boa pergunta.](#)

Por que usar o discourse?

- Muito melhor para escrever textos que possuem códigos. Com ele, podemos usar o pacote `{reprex}`!
- Saber pesquisar sobre erros e fazer a pergunta certa é essencial para aprender e resolver problemas de programação.
- No discourse, teremos mais pessoas acompanhando e respondendo as dúvidas.
- Em um ambiente aberto, as suas dúvidas vão contribuir com a comunidade.

<https://discourse.curso-r.com/>

Regras, boas práticas e estilo

Na grande maioria das vezes, você poderá resolver uma tarefa de diversas maneiras. Isso porque, apesar de existirem várias **regras**, linguagens de programação também possuem **boas práticas** e **estilos**.

- Regras: devem ser seguidas para o código funcionar (sintaxe, vocabulário)
- Boas práticas: recomenda-se seguir para a criação de códigos legíveis (espaçamento, nomes, organização)
- Estilo: cada um pode escolher aquele com o qual se sentir mais à vontade (tipos de indentação, formatação)

Introdução

Por que analisar dados usando linguagens de programação

Vantagens evidentes

- Linguagens de programação são gratuitas



- As melhores ferramentas de extração e manipulação de dados, modelagem e visualização são desenvolvidas primeiro dentro das principais linguagens de programação

Mas são dois os motivos pelos quais a gente realmente deve aprender a programar

Reprodutibilidade e Comunidade

Reprodutibilidade: o código é uma linguagem

Podemos documentar a nossa análise

```
image_read("https://jeroen.github.io/images/frink.png") %>%  
  image_rotate(270) %>%  
  image_background("blue", flatten = TRUE) %>%  
  image_border("red", "10x10") %>%  
  image_annotate("Linguagens de programação ainda\nsão linguagens!",  
                 color = "white", size = 30)
```

Entrada



Saída



Análises reprodutíveis...

- são essenciais na Ciência.
- são transparentes, algo muito importante no mercado e no setor público para evitar fraudes e má-fé.

poupam o nosso tempo.

Comunidade: todo mundo junto!

- As pessoas se ajudam tirando dúvidas e disponibilizando exemplos e análises gratuitamente.
- A linguagem cresce com a comunidade. Ao aprender a programar, você será uma desenvolvedora ou um desenvolvedor em potencial.
- Espaços seguros e de apoio a grupos sub-representados.



É fácil pedir ajuda: o código é texto!

Podemos copiar e colar



Por que o R?

O ciclo da ciência de dados

Importar



Arrumar

(Armazenar os dados
consistentemente)



Transformar

(Criar novas variáveis e
agregações)

Visualizar

(Surpreende, mas não é
escalável)

Modelar

(É escalável, mas não
surpreende)

Comunicar

Automatizar

O ciclo da ciência de dados com o R

Importar

readr
xml2
haven
readxl

Arrumar

tidyr
janitor
tibble

%>%
magrittr

Transformar

dplyr
stringr
forcats

lubridate
data.table

Visualizar

ggplot2

Modelar

caret
mlr
lme4
keras

Comunicar

rmarkdown shiny

Automatizar

plumber opencpu