Faxina de dados

curso-r

Sobre a Curso-R

A empresa

curso-r

R6 consultoria e treinamentos

www.curso-r.com

Ministrantes

Julio Trecenti

Doutorando em Estatística pelo IME-USP. Diretor da Associação Brasileira de Jurimetria (ABJ). Professor auxiliar no Insper.

Trabalha com web scraping, arrumação de dados, modelos preditivos, APIs, pacotes em R e dashboards em Shiny.

Fernando Corrêa

Bacharel e mestrando em Estatística pelo IME-USP. Ex-Diretor da Associação Brasileira de Jurimetria (ABJ).

Usa R para tudo, mas tem interesse especial em web scraping, visualização de dados e modelagem bayesiana.

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++

Modelagem

Regressão linear

Machine Learning

XGBoost

Deep Learning

Visualização de dados

Relatórios e Visualização de Dados

Dashboards

Deploy

Extração de dados

Faxina de Dados

Web scraping

curso-r

Sobre o curso

Sobre o curso

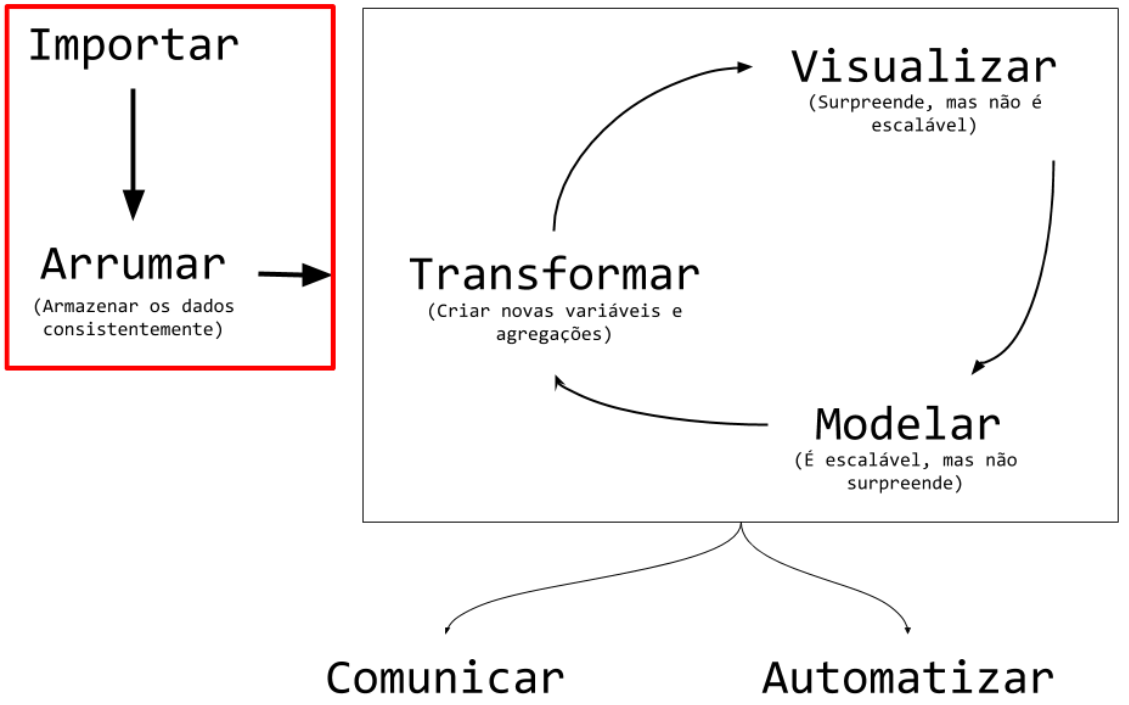
A pessoa que trabalha com ciência de dados passa **60-80%** do tempo na parte de tratamento de dados.

Por isso, trabalhar com os dados sujos da melhor forma possível tem **grande impacto** na realização dos projetos na prática.

Faxina de dados **não é fácil**. Em parte das situações, o problema não é exatamente técnico, e sim um **problema de negócio**, na fonte geradora dos dados.

O que vamos trabalhar aqui é com **ferramentas úteis** para leitura e arrumação de dados arrumados. Faremos isso através de vários exemplos.

Faxina no ciclo da ciência de dados



A faxina de dados também é conhecida como **ETL** (Extract, Transform, Load). As definições podem variar dependendo do contexto.

Requisitos

Este curso assume que você já possui os conhecimentos do curso de R para Ciência de Dados II.

Os requisitos mais importantes são:

- `{purrr}`: todos os tipos de `map()`.
- `{tidyr}` e `{dplyr}`: funções básicas, integração e pivotagem.
- `{stringr}`: funções básicas e expressões regulares.

> Se sentir dificuldade em algum ponto que a gente passar rápido, pergunte!

Aulas

Dia 01

Teoria

- Organização de projetos
- Conceito de *tidy data*

Exemplos

- Faxina de dados da SSP
  - Problemas de encoding
- Faxina de dados em um projeto de consultoria

Dia 02

Teoria

- Pacote `{janitor}`
- Funções menos conhecidas do `{dplyr}` e do `{tidyr}`
- Funções de leitura de dados

Exemplos

- Faxina de dados em um projeto de consultoria (continuação)
- Leitura de dados em outros formatos
  - PDF e OCR
  - Json e HTML/XML
- Leitura de dados grandes
  - Dados da RFB (brasil.io)

Dia 03

Teoria

- Integração de dados
- Detecção de inconsistências

Exemplos

- Faxina de dados de reclamações do Sindec (Procon)
- Fazendo um projeto completo do zero: RFB + Sindec
- Atividades que não deu tempo de fazer

Resultados

No final, você terá ...

- Conceitos básicos sobre arrumação de dados
- Conhecimento de melhores práticas
- Mais ferramentas para aplicar
- Mais tranquilidade e previsibilidade ao trabalhar com bases desarrumadas

Dinâmica

- Vários exemplos práticos por aula
    - **Foco**: bases públicas, consultoria
    - Sugira problemas do seu trabalho/pesquisa!
- Exercícios para casa, com entrega facultativa.
- Estaremos online 30 minutos antes das aulas para tirar dúvidas.
- Trabalho final, com entrega obrigatória
    - As 3 pessoas que fizerem os melhores trabalhos receberão uma **bolsa** para fazer qualquer curso da Curso-R
    - O trabalho final será definido na aula 2

Tirando dúvidas

- **Não existe dúvida idiota / básica demais**.
- Nem sempre é trivial fazer a pergunta certa para que outra pessoa esclareça a sua dúvida.
- Fora do horário de aula ou monitoria:
    - Perguntas gerais sobre o curso devem ser feitas no Classroom.
    - Perguntas sobre R, principalmente as que envolverem código, devem ser enviadas no nosso discourse.
    - Como os códigos de faxina de dados costumam ser extensos e envolvem dados grandes, tente criar um exemplo reprodutível pequeno do seu problema.
- 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/