Introdução à Programação e Ciência de Dados para a Gestão Pública

Rebeca de Jesus Carvalho

FGV CEPESP

rebeca.jesus.carvalho@gmail.com

Tópicos da aula

Conteúdo da aula de hoje

- Realizaremos uma breve recapitulação dos assuntos abordados até aqui.
- Em seguida, conheceremos as diversas formas de se importar dados em R.
- E, por fim, ampliaremos nossos conhecimentos e veremos novas funções da gramática *dplyr*.

O que aprendemos até agora

- Conhecemos a gramática do *dplyr* e suas facilidades na manipulação de dados, especialmente com o emprego do operador **pipe** (`%>%`).
- Aprendemos a filtrar, selecionar e renomear colunas: `filter()`, `select()` e `rename()`.
- Também vimos como ordenar, criar novas colunas, agrupar e sumarizar nossos dados: `arrange()`, `mutate()`, `group_by()` e `summarise()`.

O que aprendemos até agora

- Além disso, também conhecemos algumas das funções básicas disponíveis para explorar um *data frame*: `head()`, `glimpse()`, `names()`, etc.
- Exploramos os operadores aritméticos, relacionais e lógicos e seus diferentes usos.
- Discutimos as diferentes cláusulas condicionais `if`, `else` e `else if` e suas aplicações no R.
- E, finalmente, descobrimos os benefícios das estruturas de repetição: `while` e `for`.

Diretório de trabalho

- Para importar dados no R é preciso informar onde o arquivo está em seu computador (ou informar o endereço na web). O R sempre começa a procurar pelos arquivos no 'diretório de trabalho', ou *wd*.
- Para saber qual o diretório de trabalho atual, usamos a função `getwd()` .
- Para alterar o endereço do diretório de trabalho, usamos a função `setwd()`.

O pacote *readr*

- O pacote *readr* contém funções para abertura de dados 'retangulares' (linhas x colunas) em formato de texto (.csv, .tsv, .txt).

- Um mesmo arquivo pode ser salvo em diferentes estruturas:

  - Diferentes separadores (vírgula, ponto e vírgula, *tab*, etc);
  - Com ou sem cabeçalho, isto é, nomes das colunas; e
  - Diferentes *encodings*, isto é, padrão em que os caracteres foram salvos.

A função `fread()`

- A função `fread()` é extremamente rápida e reconhece automaticamente boa parte dos parâmetros necessários para abrir os dados.

- Ela consegue abrir, de forma ágil, bases muito grandes na memória RAM. Característica que as funções do *readr* não possuem ou demoram muito para fazer.

Importando planilhas

- Há dois bons pacotes com funções para dados em editores de planilha: *readxl* e *gdata*. Vamos trabalhar apenas com o primeiro.

- Embora o *readxl* também integre o *tidyverse*, temos que abrí-lo de forma independente, pois não é carregado automaticamente ao carregarmos o pacote *tidyverse*.

- É possível trabalhar com arquivos com múltiplas planilhas. Podemos usar tanto o nome quanto a posição para indicar qual planilha utilizaremos no argumento *sheet*.

Dados de SPSS, Stata e SAS

- R é bastante flexível quanto à importação de dados de outros softwares estatísticos. Para este fim, temos o pacote *haven*, que também é parte do *tidyverse*.

- Existem cinco funções de importação de dados em *haven*:

  - Para dados do SPSS: `read_sav()` e `read_por()`.
  - Para dados em formato *.dta* gerados em Stata: `read_stata()` e `read_dta()`.
  - Para dados em SAS: `read_sas()`.

Agrupamentos, tabelas e reestruturação dos dados

- A função `count()` serve para contar as linhas dentro de cada 'grupo' de uma variável.
- A função `str_sub()` ('str' de *string* e 'sub' de *subset*), permite a extração de uma parcela dos elementos de uma *string* com base na posição de início e fim.
- A função `recode()` possibilita a recodificação dos valores de uma variável para outros de nossa escolha.
- A função `replace()` trabalha de forma similar a função `recode()` e também permite a 'troca' de um padrão textual encontrado em uma variável por outro.

Long e *wide*

- Uma base no formato *long* (ou seja, "comprido") possui mais linhas e pode ter menos colunas, enquanto no formato *wide* ("largo") possui menos linhas e pode ter mais colunas.
- Na gramática do *dplyr* trabalharemos sempre com o formato *long*.
- Existem dois verbos no *tidyverse* que são utilizados para redesenhar *data frames* de *long* em *wide* e vice-versa: `pivot_wider()` e `pivot_longer()`.

Laboratório

Agora é o momento de partir para a ação! Temos dois tutoriais agendados para hoje, prontos para serem explorados. Eles estão disponibilizados no GitHub, e você pode acessá-los clicando neste **link**.

Se deixou algum tutorial inacabado dos encontros anteriores, comece por eles. Caso contrário, prossiga.

Dúvidas?