

ME115 - Linguagem R

Atividade Prática 07

1º semestre de 2023

Introdução

Nessa atividade, exploraremos os seguintes tópicos:

1. Criação de *tibbles* usando o pacote `tibble`;
2. Formatação de banco de dados usando o pacote `tidyr`.

Antes de iniciar a atividade instale, se necessário, e carregue os pacotes `tibble`, `tidyr` e `reshape2`.

Atividade

1. Crie o conjunto de dados chamado `nba` que seja um *tibble* com as colunas a seguir. Esse conjunto de dados está no formato *tidy*? Se não estiver, explique.

```
player <- c("James", "Durant", "Curry", "Harden", "Paul", "Wade")
team <- c("CLEOH", "GSWCA", "GSWCA", "HOUTX", "HOUTX", "CLEOH")
day1points <- c(25, 23, 30, 41, 26, 20)
day2points <- c(24, 25, 33, 45, 26, 23)
```

2. Transforme o conjunto de dados `nba` para o formato *tidy*, usando cada uma das funções abaixo. Note que as colunas `player` e `teams` estão fixas.
 - `melt()` do pacote `reshape2` e armazene-o no objeto `nba_melt`.
 - `gather()` do pacote `tidyr` e armazene-o no objeto `nba_gather`.
 - `pivot_longer()` do pacote `tidyr` e armazene-o no objeto `nba_long`.
3. As funções `pivot_wider()` e `spread()` são equivalentes. Ambas pegam diferentes valores de uma variável e os espalha em diferentes colunas. Imagine isso como o reverso de `pivot_longer()` e `gather()`, respectivamente. Então, transforme os objetos `nba_long` e `nba_gather` de volta para o formato *wide*.
4. A função `separate()` pega valores dentro de uma coluna e os separa. Como você provavelmente já viu, a coluna `team` é um pouco estranha. Obviamente, há mais de uma variável lá e temos que consertá-la. O time e o estado dos EUA são nossas variáveis, então temos que criar duas colunas, uma para o time (`team`) e outra para o estado (`state`). O que o argumento `sep` representa?

```
## # A tibble: 6 x 5
##   player team  state day      points
##   <chr> <chr> <chr> <chr>    <dbl>
## 1 James  CLE   OH    day1points    25
## 2 James  CLE   OH    day2points    24
## 3 Durant GSW    CA    day1points    23
## 4 Durant GSW    CA    day2points    25
## 5 Curry  GSW    CA    day1points    30
## 6 Curry  GSW    CA    day2points    33
```

5. Crie o banco de dados **wide** como mostrado a seguir:

```
wide <- tibble(ID = 1:10,  
              Face.1 = c(411, 723, 325, 456, 579, 612, 709, 513, 527, 379),  
              Face.2 = c(123, 300, 400, 500, 600, 654, 789, 906, 413, 567),  
              Face.3 = c(1457, 1000, 569, 896, 956, 2345, 780, 599, 1023, 678))
```

Utizando esse conjunto de dados:

- Organize o banco de dados **wide** e guarde-o no objeto **long**, criando as colunas **Face** e **ResponseTime** usando a função **pivot_longer()**.
 - Usando a função **separate()**, separe o número da palavra “Face” na coluna **Face**, guardando a palavra “Face” na coluna **Target** e o número na coluna **Number**. Guarde o novo *data frame* no objeto **long_separate**.
 - Usando a função **unite()** no objeto **long_separate**, junte o conteúdo de **Target** e **Number** e guarde em **Face**. Guarde o novo *data frame* no objeto **long_unite**.
 - Utizando o objeto **long_unite** volte ao formato **wide** usando a função **pivot_wider()** e guarde o novo formato no objeto **back_to_wide**.
6. Os dados **iris** não estão no formato *tidy*. Por quê? No último slide da aula, ficou a tarefa de colocarmos os dados **iris** no formato *tidy*. Para isso:
- usar a função **pivot_longer()** para agrupar as colunas 1 a 4 em uma única coluna, com os valores em uma coluna chamada **Medida**;
 - usar a função **separate()** para separar a parte da planta e a dimensão em duas colunas.

Agradecimento

O material foi produzido pela Profa. Tatiana Benaglia para o curso de ME115.