ME115 - Linguagem R

Atividade Prática 08 - Gabarito

1º semestre de 2023

Introdução

Nessa atividade, exploraremos os seguintes tópicos:

1. Aplicaremos a ideia: dado original —> seleção —> filtro (*pipe* ou `%>%`);
2. Principais verbos do pacote `dplyr`: `select()`, `filter()`, `arrange()`, `mutate()`, `summarize()`, e `group_by()`.

Antes de iniciar a atividade instale, se necessário, e carregue os pacotes `tidyverse` e `dslabs`. Note que ao carregar o `tidyverse`, vários pacotes são carregados, incluindo o `dplyr`. Veja quais são os demais.

```
library(tidyverse)
library(dslabs)
```

Nessa atividade, iremos trabalhar com o conjunto de dados `murders` do pacote `dslabs`. Carregue o conjunto de dados e use a função `glimpse()` do pacote `dplyr` para olhar sua estrutura. Compare com a função `str()` da base do R.

Solução:

```
data(murders)
glimpse(murders)
```

```
## Rows: 51
## Columns: 5
## $ state      <chr> "Alabama", "Alaska", "Arizona", "Arkansas", "California", "~
## $ abb        <chr> "AL", "AK", "AZ", "AR", "CA", "CO", "CT", "DE", "DC", "FL",~
## $ region     <fct> South, West, West, South, West, West, Northeast, South, Sou~
## $ population <dbl> 4779736, 710231, 6392017, 2915918, 37253956, 5029196, 35740~
## $ total      <dbl> 135, 19, 232, 93, 1257, 65, 97, 38, 99, 669, 376, 7, 12, 36~
```

```
str(murders)
```

```
## 'data.frame':    51 obs. of  5 variables:
##  $ state     : chr  "Alabama" "Alaska" "Arizona" "Arkansas" ...
##  $ abb       : chr  "AL" "AK" "AZ" "AR" ...
##  $ region    : Factor w/ 4 levels "Northeast","South",..: 2 4 4 2 4 4 1 2 2 2 ...
##  $ population: num  4779736 710231 6392017 2915918 37253956 ...
##  $ total     : num  135 19 232 93 1257 ...
```

Atividade

1. Usando a função `mutate()`, adicione uma nova coluna chamada `rate` aos dados `murders` do pacote `dslabs`, dada por `rate = total/ population * 100000`.

   Solução:

```
murders <- murders %>% mutate(rate = total/ population * 100000)
# murders %<>% mutate(rate = total/ population * 100000)  # operador pipe + '<-'
glimpse(murders)
```

```
## Rows: 51
## Columns: 6
## $ state      <chr> "Alabama", "Alaska", "Arizona", "Arkansas", "California", "~
## $ abb        <chr> "AL", "AK", "AZ", "AR", "CA", "CO", "CT", "DE", "DC", "FL",~
## $ region     <fct> South, West, West, South, West, West, Northeast, South, Sou~
## $ population <dbl> 4779736, 710231, 6392017, 2915918, 37253956, 5029196, 35740~
## $ total      <dbl> 135, 19, 232, 93, 1257, 65, 97, 38, 99, 669, 376, 7, 12, 36~
## $ rate       <dbl> 2.8244238, 2.6751860, 3.6295273, 3.1893901, 3.3741383, 1.29~
```

2. A partir do conjunto de dados `murders` aumentado em (1), selecione as variáveis `state, region, rate` e os registros onde a taxa (`rate`) é maior que 0.6 usando as funções `select()` e `filter()`. Escreva seu código do modo tradicional, ou seja, sem usar o operador `%>%` (pipe).

   Solução:

```
new_table <- select(murders, state, region, rate)
high_rates <- filter(new_table, rate > 0.6)
slice_min(high_rates, rate, n = 5)
```

```
##     state        region      rate
## 1    Iowa North Central 0.6893484
## 2   Idaho          West 0.7655102
## 3    Utah          West 0.7959810
## 4   Maine     Northeast 0.8280881
## 5 Wyoming          West 0.8871131
```

   Outras soluções apresentadas em aula:

```
sol.1 <- filter(select(murders, state, region, rate), rate > 0.6)
slice_min(sol.1, rate, n = 5)
```

```
##     state        region      rate
## 1    Iowa North Central 0.6893484
## 2   Idaho          West 0.7655102
## 3    Utah          West 0.7959810
## 4   Maine     Northeast 0.8280881
## 5 Wyoming          West 0.8871131
```

```
sol.2 <- subset(murders[, c("state", "region", "rate")], rate > 0.6)
slice_min(sol.2, rate, n = 5)
```

```
##     state        region      rate
## 1    Iowa North Central 0.6893484
## 2   Idaho          West 0.7655102
## 3    Utah          West 0.7959810
## 4   Maine     Northeast 0.8280881
## 5 Wyoming          West 0.8871131
```

3. A partir do conjunto de dados `murders` aumentado em (1), selecione as variáveis `state, region, rate` e os registros onde a taxa (`rate`) é maior que 0.6 usando as funções `select()` e `filter()`, agora com o operador `%>%`. Observe as diferenças entre os dois códigos.

   Solução:

```
murders %>%
  select(state, region, rate) %>%
  filter(rate > 0.6) %>%
  slice_min(rate, n = 5)
```

```
##     state        region      rate
## 1    Iowa North Central 0.6893484
## 2   Idaho          West 0.7655102
## 3    Utah          West 0.7959810
## 4   Maine     Northeast 0.8280881
## 5 Wyoming          West 0.8871131
```

4. Crie uma coluna chamada `rank` nos dados `murders` contendo o posto em ordem descrescente do estado de acordo com a taxa de assassinatos. Dica: `rank()`.

Solução:

```
murders <- murders %>% mutate(rank = rank(-rate))
murders %>% arrange(rank)
```

```
##                  state abb        region population total       rate rank
## 1  District of Columbia  DC         South     601723    99 16.4527532    1
## 2             Louisiana  LA         South    4533372   351  7.7425810    2
## 3              Missouri  MO North Central    5988927   321  5.3598917    3
## 4              Maryland  MD         South    5773552   293  5.0748655    4
## 5        South Carolina  SC         South    4625364   207  4.4753235    5
## 6              Delaware  DE         South     897934    38  4.2319369    6
## 7              Michigan  MI North Central    9883640   413  4.1786225    7
## 8           Mississippi  MS         South    2967297   120  4.0440846    8
## 9               Georgia  GA         South    9920000   376  3.7903226    9
## 10              Arizona  AZ          West    6392017   232  3.6295273   10
## 11         Pennsylvania  PA     Northeast   12702379   457  3.5977513   11
## 12            Tennessee  TN         South    6346105   219  3.4509357   12
## 13              Florida  FL         South   19687653   669  3.3980688   13
## 14           California  CA          West   37253956  1257  3.3741383   14
## 15           New Mexico  NM          West    2059179    67  3.2537239   15
## 16                Texas  TX         South   25145561   805  3.2013603   16
## 17             Arkansas  AR         South    2915918    93  3.1893901   17
## 18             Virginia  VA         South    8001024   250  3.1246001   18
## 19               Nevada  NV          West    2700551    84  3.1104763   19
## 20       North Carolina  NC         South    9535483   286  2.9993237   20
## 21             Oklahoma  OK         South    3751351   111  2.9589340   21
## 22             Illinois  IL North Central   12830632   364  2.8369608   22
## 23              Alabama  AL         South    4779736   135  2.8244238   23
## 24           New Jersey  NJ     Northeast    8791894   246  2.7980319   24
## 25          Connecticut  CT     Northeast    3574097    97  2.7139722   25
## 26                 Ohio  OH North Central   11536504   310  2.6871225   26
## 27               Alaska  AK          West     710231    19  2.6751860   27
## 28             Kentucky  KY         South    4339367   116  2.6732010   28
## 29             New York  NY     Northeast   19378102   517  2.6679599   29
## 30               Kansas  KS North Central    2853118    63  2.2081106   30
## 31              Indiana  IN North Central    6483802   142  2.1900730   31
## 32        Massachusetts  MA     Northeast    6547629   118  1.8021791   32
## 33             Nebraska  NE North Central    1826341    32  1.7521372   33
## 34            Wisconsin  WI North Central    5686986    97  1.7056487   34
```

```
## 35         Rhode Island  RI     Northeast    1052567    16  1.5200933   35
## 36        West Virginia  WV         South    1852994    27  1.4571013   36
## 37           Washington  WA          West    6724540    93  1.3829942   37
## 38             Colorado  CO          West    5029196    65  1.2924531   38
## 39              Montana  MT          West     989415    12  1.2128379   39
## 40            Minnesota  MN North Central    5303925    53  0.9992600   40
## 41         South Dakota  SD North Central     814180     8  0.9825837   41
## 42               Oregon  OR          West    3831074    36  0.9396843   42
## 43              Wyoming  WY          West     563626     5  0.8871131   43
## 44                Maine  ME     Northeast    1328361    11  0.8280881   44
## 45                 Utah  UT          West    2763885    22  0.7959810   45
## 46                Idaho  ID          West    1567582    12  0.7655102   46
## 47                 Iowa  IA North Central    3046355    21  0.6893484   47
## 48         North Dakota  ND North Central     672591     4  0.5947151   48
## 49               Hawaii  HI          West    1360301     7  0.5145920   49
## 50        New Hampshire  NH     Northeast    1316470     5  0.3798036   50
## 51              Vermont  VT     Northeast     625741     2  0.3196211   51
```

5. Calcule a média e o desvio padrão da taxa de assassinatos segundo a região e guarde o resultado no objeto `murder.by.region`. Qual a região mais segura? Dica: `group_by()` e `summarize()`.

   Solução:

```r
murder.by.region <- murders %>%
  group_by(region) %>%
  summarize(media = mean(rate), desvio_padrao = sd(rate))

murder.by.region
```

```
## # A tibble: 4 x 3
##   region        media desvio_padrao
##   <fct>         <dbl>         <dbl>
## 1 Northeast      1.85          1.17
## 2 South          4.42          3.37
## 3 North Central  2.18          1.44
## 4 West           1.83          1.17
```

   A região mais segura é aquela com menor taxa média de assassinatos, no caso, a região a West.

```r
murder.by.region %>% filter(media == min(media))
```

```
## # A tibble: 1 x 3
##   region media desvio_padrao
##   <fct>  <dbl>         <dbl>
## 1 West    1.83          1.17
```

```r
murder.by.region %>% slice_min(media)
```

```
## # A tibble: 1 x 3
##   region media desvio_padrao
##   <fct>  <dbl>         <dbl>
## 1 West    1.83          1.17
```

6. Ordene o objeto `murder.by.region` em ordem decrescente de taxa de assassinato média.

   Solução:

```r
murder.by.region %>% arrange(desc(media))
```

```
## # A tibble: 4 x 3
##   region        media desvio_padrao
##   <fct>         <dbl>         <dbl>
## 1 South          4.42          3.37
## 2 North Central  2.18          1.44
## 3 Northeast      1.85          1.17
## 4 West           1.83          1.17
```

7. Calcule quantidade de estados da região Sul (*South*) com taxa de assassinatos menor do que a média de assassinatos da mesma região.

   Solução:

```
murders %>%
  filter(region == 'South') %>%
  filter(rate < mean(rate)) %>%
  summarise(count = n())
```

```
##   count
## 1    13
```

   Ou,

```
murders %>%
  filter(region == "South") %>%
  filter(rate < mean(rate)) %>%
  count()
```

```
##    n
## 1 13
```

   Ou,

```
murders %>%
  filter(region == 'South') %>%
  filter(rate < mean(rate)) %>%
  nrow()
```

```
## [1] 13
```

8. Calcule a proporção de estados para cada região com taxa de assassinatos menor do que a média de assassinatos da respectiva região. Qual a região mais segura?

   Solução:

```
murders %>%
  select(state, region, rate) %>%
  group_by(region) %>%
  mutate(mean_rate = mean(rate), rate_lower = rate < mean_rate) %>%
  summarise(low_rate = sum(rate_lower),
            n = n(),
            prop_low_rate = mean(rate_lower)) %>%
  arrange(prop_low_rate)
```

```
## # A tibble: 4 x 4
##   region        low_rate     n prop_low_rate
##   <fct>            <int> <int>         <dbl>
## 1 North Central        6    12         0.5
## 2 Northeast            5     9         0.556
## 3 West                 8    13         0.615
```

```
## 4 South               13    17         0.765
```

```
# Fazendo direto no summarise
murders %>%
  group_by(region) %>%
  summarise(low_rate = sum(rate < mean(rate, na.rm = TRUE)),
            n = n(),
            prop = low_rate/n) %>%
  arrange(prop)
```

```
## # A tibble: 4 x 4
##   region        low_rate     n  prop
##   <fct>            <int> <int> <dbl>
## 1 North Central        6    12 0.5
## 2 Northeast            5     9 0.556
## 3 West                 8    13 0.615
## 4 South               13    17 0.765
```

9. Crie uma nova coluna chamada `rank10` em `murders` usando `mutate()` tal, baseado na coluna `rank` criada em (4), ela seja 1 se o estado foi rankeado abaixo de 10 e 0 caso contrário. A seguir faça uma tabela classificando os estados abaixo da décima posição, por região. Qual região é mais segura?

   Solução:

```
murders <- murders %>% mutate(rank10 = ifelse(rank < 10, 1, 0))
murders %>%
  group_by(region, rank10) %>%
  summarise(n = n()) %>%
  pivot_wider(names_from = rank10, values_from = n, values_fill = 0)
```

```
## # A tibble: 4 x 3
## # Groups:   region [4]
##   region          `0`   `1`
##   <fct>         <int> <int>
## 1 Northeast         9     0
## 2 South            10     7
## 3 North Central    10     2
## 4 West             13     0
```

```
murders %>%
  group_by(region) %>%
  summarize(Total = n(),
            EstadosViolentos = sum(rank10),
            Proporcao = EstadosViolentos / Total) %>%
  arrange(desc(Proporcao))
```

```
## # A tibble: 4 x 4
##   region        Total EstadosViolentos Proporcao
##   <fct>         <int>            <dbl>     <dbl>
## 1 South            17                7     0.412
## 2 North Central    12                2     0.167
## 3 Northeast         9                0     0
## 4 West             13                0     0
```

Agradecimento

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