

Workshop API CepespData - Folha de Respostas

Equipe CepespData/FGV

28 de junho de 2019

5. Exercícios

5.1. Quantos votos na legenda o PSL teve em 2018 para Deputado Federal? E em 2014? Houve aumento?

```
lista.colunas <- list("ANO_ELEICAO", "NUMERO_CANDIDATO", "QTDE_VOTOS")

votosPSL <- get_elections(year="2014,2018",
                          position="Deputado Federal",
                          regional_aggregation="Brasil",
                          candidate_number = "17",
                          columns_list = lista.colunas)

print(votosPSL)
```

```
##      ANO_ELEICAO NUMERO_CANDIDATO QTDE_VOTOS
## 1:          2014              17      36082
## 2:          2018              17     1027661
```

5.2. Quantas governadoras foram eleitas nas últimas quatro eleições?

```
lista.colunas <- list("ANO_ELEICAO", "SIGLA_UE", "NUMERO_CANDIDATO", "QTDE_VOTOS", "DESCRICAO_SEXO")

governadoras <- get_elections(year="2006,2010,2014,2018",
                              position="Governador",
                              regional_aggregation="Brasil",
                              only_elected = T,
                              columns_list = lista.colunas)

governadoras %>%
  group_by(DESCRICAO_SEXO) %>%
  summarise(n = n())
```

```
## # A tibble: 2 x 2
##   DESCRICAO_SEXO      n
##   <chr>          <int>
## 1 FEMININO         7
## 2 MASCULINO       102
```

5.3. Quantas mulheres negras (pretas ou pardas) concorreram ao cargo de prefeita em 2016 no Brasil?

```

lista.colunas <- list("ANO_ELEICAO", "DESCRICAO_COR_RACA", "QTDE_VOTOS", "DESCRICAO_SEXO")

prefeitasnegras <- get_elections(year="2016",
                                position="Prefeito",
                                columns_list = lista.colunas)

prefeitasnegras %>%
  filter(DESCRICAO_COR_RACA %in% c("PRETA", "PARDA")) %>%
  group_by(DESCRICAO_SEXO, DESCRICAO_COR_RACA) %>%
  summarise(n = n())

```

```

## # A tibble: 4 x 3
## # Groups:   DESCRICAO_SEXO [2]
##   DESCRICAO_SEXO DESCRICAO_COR_RACA      n
##   <chr>          <chr>          <int>
## 1 FEMININO      PARDA              601
## 2 FEMININO      PRETA              59
## 3 MASCULINO     PARDA             4169
## 4 MASCULINO     PRETA             449

```

5.4. Quantas pessoas que eram filiadas ao PCO no Estado de Alagoas se desfiliaram do partido ou tiveram sua filiação cancelada?

```

filiadosAL <- get_filiates(state='AL', party = 'PCO')

filiadosAL %>%
  group_by(SITUACAO_REGISTRO) %>%
  summarise(n = n())

```

```

## # A tibble: 3 x 2
##   SITUACAO_REGISTRO      n
##   <chr>          <int>
## 1 CANCELADO         4
## 2 DESFILIADO        3
## 3 REGULAR          66

```

5.5. Considerando os candidatos a vereador no Rio Grande do Sul em 2012, qual é o partido com o maior valor total de bens declarados?

```

lista.colunas <- list("NUMERO_PARTIDO", "SIGLA_PARTIDO", "VALOR_BEM")

bens_vereadores <- get_assets(year=2012,
                              state = "RS",
                              columns_list = lista.colunas)

bens_vereadores %>%
  mutate(valor = gsub("[.]", "", VALOR_BEM), #remove o ponto do numeral
         valor = as.numeric(sub(",", ".", valor))) %>% #transforma a vírgula em ponto
  group_by(SIGLA_PARTIDO) %>%

```

```
summarise(total = sum(valor)) %>%
top_n(1,total)
```

```
## # A tibble: 1 x 2
##   SIGLA_PARTIDO      total
##   <chr>             <dbl>
## 1 PT               311063311.
```

6. Trabalhando com outras bases de dados a partir do código do IBGE

6.1. Baixe os dados do Programa Bolsa Família (pbf_2016.csv) e abra no R. Dica: veja como utilizar a função `read.csv2()`. Inspecione quais são as variáveis no banco.

```
pbf <- read.csv2("pbf_2016.csv",stringsAsFactors = F)
glimpse(pbf)
```

```
## Observations: 5,570
## Variables: 3
## $ qtd_familias_beneficiarias_bolsa_familia <int> 1800, 4969, 351, 3404...
## $ valor_repassado_bolsa_familia           <int> 272480, 664551, 44622...
## $ cod_ibge                                <int> 110001, 110002, 11000...
```

6.2. Baixe os dados da população do IBGE (pop_2016.csv) e abra no R. Inspecione quais são as variáveis no banco.

```
pop <- read.csv2("pop_ibge_2016.csv",stringsAsFactors = F)
glimpse(pop)
```

```
## Observations: 5,570
## Variables: 4
## $ uf          <chr> "RO", "RO", "RO", "RO", "RO", "RO", "RO", "RO", "...
## $ nome_mun     <chr> "Alta Floresta D'Oeste", "Ariquemes", "Cabixi", "...
## $ pop_estimada <int> 25506, 105896, 6289, 87877, 17959, 18639, 8749, 1...
## $ cod_mun      <int> 1100015, 1100023, 1100031, 1100049, 1100056, 1100...
```

6.3. Utilizando o `cepespR`, construa um banco de dados que tenha a quantidade de votos por município por partido para o cargo de prefeito no ano de 2016

```
lista.colunas <- list("NUMERO_CANDIDATO","QTDE_VOTOS","COD_MUN_IBGE")

votos <- get_votes(year=2016,
                   position = "Prefeito",
                   columns_list = lista.colunas)
```

6.4. Crie uma variável no banco de votos que indique a porcentagem de votos recebida por partido por município.

```
#fazendo os totais de votos por município
votos_totais <- votos %>%
  group_by(COD_MUN_IBGE) %>%
  summarise(votosmun = sum(QTDE_VOTOS))

#juntando os totais e criando a variável com a porcentagem
votos <- votos %>%
  left_join(votos_totais, by="COD_MUN_IBGE") %>%
  mutate(pct_votos = 100*(QTDE_VOTOS/votosmun))

summary(votos$pct_votos)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
##  0.00724  14.64858  39.73689  34.51525  50.32925 100.00000
```

6.5. Crie um novo banco de dados que junte o banco de população com os dados do Bolsa Família utilizando o código do IBGE.

```
#vendo como é a variável do código do IBGE de cada um dos bancos
summary(pbf$cod_ibge)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 110001 251213 314628 325359 411919 530010
```

```
summary(pop$cod_mun)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 1100015 2512126 3146280 3253591 4119190 5300108
```

```
#ambas são numéricas, mas há diferença no número de dígitos
#o banco com os dados da população possui um dígito a mais.
```

```
#criando uma nova variável do código com 6 dígitos, para ficar igual ao do PBF:
pop <- pop %>%
  mutate(cod_mun2 = as.numeric(substr(cod_mun, start = 1, stop = 6)))

#fazendo o join
bd_join <- pop %>%
  left_join(pbf, by = c("cod_mun2" = "cod_ibge"))
```

6.6. Verifique se o join foi feito corretamente - o novo banco deve ter o mesmo número de linhas que os bancos originais. Também explore o novo banco com o comando `summary()`. Não podemos ter NA's!

```
#testando se não perdemos ou ganhamos linhas no join
nrow(bd_join) == nrow(pbf)
```

```
## [1] TRUE
```

```
nrow(bd_join) == nrow(pop)
```

```
## [1] TRUE
```

```
#vendo se todas as variáveis estão completas
summary(bd_join)
```

```
##      uf      nome_mun      pop_estimada
## Length:5570      Length:5570      Min.   :    815
## Class :character      Class :character      1st Qu.:   5483
## Mode  :character      Mode  :character      Median :   11578
##                                     Mean    :   36998
##                                     3rd Qu.:   25085
##                                     Max.    :12038175
##      cod_mun      cod_mun2
## Min.   :1100015      Min.   :110001
## 1st Qu.:2512126      1st Qu.:251213
## Median :3146280      Median :314628
## Mean    :3253591      Mean    :325359
## 3rd Qu.:4119190      3rd Qu.:411919
## Max.    :5300108      Max.    :530010
## qtd_familias_beneficiarias_bolsa_familia      valor_repassado_bolsa_familia
## Min.   :    4.0      Min.   :    415
## 1st Qu.:   384.2      1st Qu.:   56158
## Median :  1004.5      Median :  161696
## Mean    :  2508.0      Mean    :  404424
## 3rd Qu.:  2510.2      3rd Qu.:  418757
## Max.    :452357.0      Max.    :61204681
```

6.7. Nesse novo banco, crie uma variável que indique o número de famílias beneficiárias a cada 1000 habitantes.

```
bd_join <- bd_join %>%
  mutate(familias_1000hab = (qtd_familias_beneficiarias_bolsa_familia/pop_estimada)*1000)
summary(bd_join$familias_1000hab)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  1.504  43.100  84.391  95.607 149.756 326.983
```

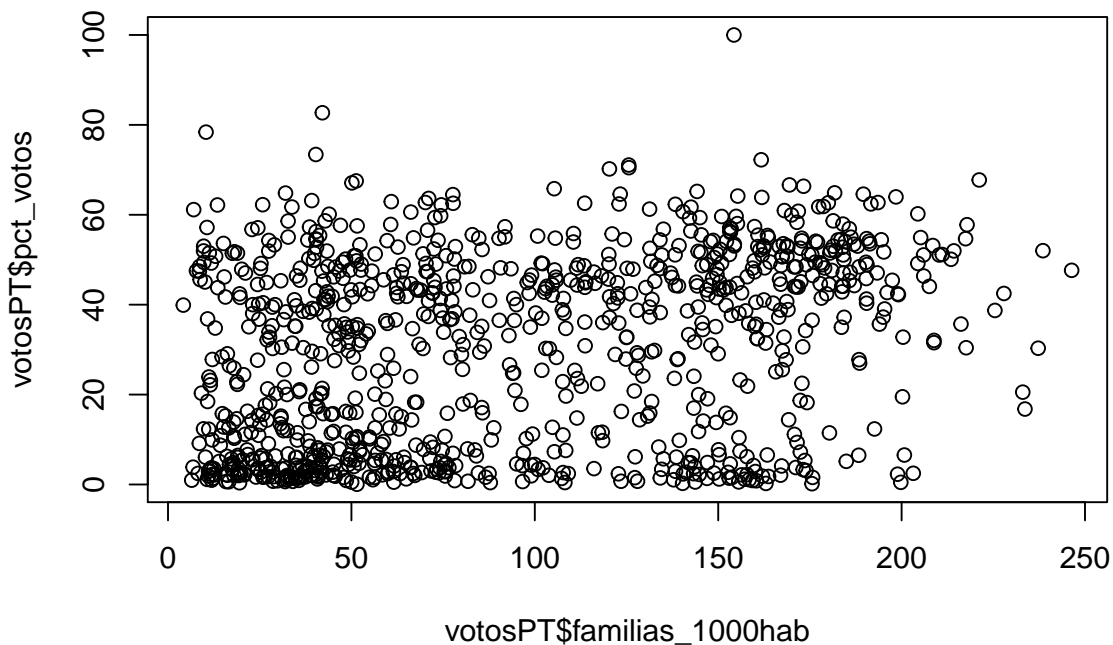
6.8. Agora, crie um novo banco de dados que junte o banco com a variável do item 7 ao banco com as porcentagens de voto. Confirme que o novo banco tem o mesmo número de observações que o original.

```
votos_completo <- votos %>%  
  mutate(cod_mun = COD_MUN_IBGE) %>%  
  left_join(bd_join, by="cod_mun")  
  
nrow(votos_completo) == nrow(votos)
```

```
## [1] TRUE
```

6.9. Filtre o banco com todas as variáveis para que tenha somente candidatos ao PT, e faça um gráfico de dispersão simples (*plot()*) entre a taxa de famílias beneficiárias e a porcentagem de votos para candidatos. Parece haver correlação?

```
votosPT <- votos_completo %>%  
  filter(NUMERO_CANDIDATO == 13)  
  
plot(votosPT$familias_1000hab, votosPT$pct_votos)
```





Rua Itapeva, 286 - 10º andar - São Paulo/SP - CEP: 01332-000

Telefone:(11) 3799 - 3228

E-mail: cepesp@fgv.br / midia.cepesp@fgv.br

[Twitter](#) / [Facebook](#) / [Instagram](#) / [GitHub](#)