# Atividade 05

Charles Santos

3/15/2021

## Contents

### Proposta da atividade

O objetivo é realizar a junção de bases de dados com informações de municípios e estados para analisar a relação entre as variáveis:

1. V1: Taxas de homicídio (em 100 mil habitantes) por município em 2010;
2. V2: População por estado em 2000;
3. V3: População por estado em 2010;
4. V4: Produto Interno Bruto (PIB) por município em 2005;
5. V5: Produto Interno Bruto (PIB) por município em 2010;
6. V6: Índice de Desenvolvimento Humano geral em 2010;
7. V7: Ranking do IDH em 2010;
8. V8: Estado;
9. V9: Macrorregião.

A partir da consolidação da base de dados que contenha todos os municípios brasileiros, crie as seguintes variáveis:

1. V10: Variação da população entre 2000 e 2010 por estado;
2. V11: Variação do PIB entre 2005 e 2010;
3. V12: Categorias do PIB em 2010: Muito baixo; Baixo; Médio; Alto. A divisão deve ser feita de tal forma que as 4 categorias tenham o mesmo número de municípios (ou próximo disso).
4. Para o arquivo final vocês vão calcular o coeficiente de correlação entre as variáveis V1, V6, V11.

---

**Atenção:** Clique aqui para baixar a versão em PDF ou aqui para ver a versão renderizada em HTML deste estudo.

---

**Pacote necessário para a solução** Para a solução dos problemas os pacotes htmltab — ferramenta excepcional para simplificar a importação e parse de páginas direto da Internet — e DataTables — ferramenta para gerar tabelas ativas com visual agradável — foram utilizados e são necessários para a geração desta documentação.

**Atenção:** Todas as bases de dados para a produção deste documento são carregadas dinamicamente da Internet.

```
if (!require('DT')) install.packages('DT')
```

```
## Loading required package: DT
```

```
if (!require('htmltab')) install.packages('htmltab')
```

```
## Loading required package: htmltab
```

```
library(DT)
library(htmltab)
```

---

### Base taxa de homicídios 2010

Os dados para plotagem das taxas de homicídios por município de 2010 foram baixados do site do IPEA no formato CSV e disponibilizado no meu Github pessoal para possibilitar a execução deste *script* em qualquer máquina.

```
# Leitura dos dados da taxa de homicídio dos municípios em 2010
TH2010 <- read.csv('https://raw.githubusercontent.com/thyarles/thyarles.github.io/master/est0001/taxa-h

# Configuração dos títulos das colunas, para ficarem mais adequados à proposta
names(TH2010) <- c('id_mun', 'municipio', 'periodo', 'taxaHom2010')

# Remoção da coluna período (desnecessária)
TH2010$periodo <- NULL

# Verficação dos dados até o momento
datatable(TH2010[order(TH2010$municipio), c(1:3)],
          colnames = c('ID', 'Município', 'Taxa de homicídio em 2010'),
          caption = 'Tabela 1: Taxa de homicídio nos municípios em 2010.')
```

---

### Base população por Estado 2000

Para a população por Estado do Brasil em 2000 coletou-se, em tempo de execução, dados do Wikipedia. Dado que os valores foram verificados e confirmados com outras bases de dados, e que a proposta deste trabalho é unir dados das fontes mais diversas possíveis, a Wikipedia pareceu adequada como parte da solução.

```
# Configuracao da URL e elemento da pagina
url <- 'https://pt.wikipedia.org/wiki/Censo_demogr%C3%A1fico_do_Brasil_de_2000'
xp  <- '//table[contains(@class,"wikitable")]'

# Coleta dos daados (PE = População por Estado)
PE <- htmltab(doc = url, which = xp, rm_whitespace = TRUE)
remove(url, xp)

# Apagar Brasil (não é Estado)
PE <- PE[-28,]

# Apagar Rank (não é necessário)
PE <- PE[, 2:3]

# Ajuste nos títulos
names(PE) <- c('nome_uf', 'pop2000')

# Ajuste dos números
pt_BR <- readr::locale(decimal_mark = ',')
PE[2] <- lapply(PE[2], function(i) readr::parse_number(i, locale = pt_BR))
remove(pt_BR)
```

```r
# Ajuste nas linhas
rownames(PE) <- c(1:27)

# Retirar espaços em brancos especial em torno dos nomes dos Estados
PE$nome_uf <- stringr::str_replace(PE$nome_uf, pattern = '\u00A0', '')

# Estado atual da tabela PE
datatable(PE, colnames = c('UF', 'População em 2000'),
          caption = 'Tabela 2: População por Estado em 2000.')
```

---

**Base população por Estado 2010**

Para a população por Estado do Brasil em 2010 coletou-se, em tempo de execução, dados do IBGE. Como a proposta deste trabalho é pegar dados das fontes mais diversas possíveis, o IBGE, renomada empresa que coleta dados há anos, pareceu-me adequada usá-la como parte da solução.

_Observação: Com os dados obtidos eu poderia usar para solucionar tanto o item V2 quanto o item V3. Optei por manter os dados da Wikipedia para ficar registrado a diversidade de possibilidades de se obter os dados.

```r
# Configuracao da URL e elemento da pagina
url <- 'https://censo2010.ibge.gov.br/sinopse/index.php?dados=4&uf=00'
xp  <- '//*[@id="div_tabela_dados"]/table'

# Coleta dos dados
aux <- htmltab(doc = url, which = xp, rm_whitespace = TRUE)
```

```
## Warning: Columns [Região] seem to have no data and are removed. Use
## rm_nodata_cols = F to suppress this behavior.
```

```r
remove(url, xp)

# Ajuste no título da UF
colnames(aux)[1] <- 'nome_uf'
colnames(aux)[13] <- 'pop2010'

# Ajuste nas linhas
rownames(aux) <- c(1:27)

# Substituição dos valores ... por NA
aux[aux == '...'] <- NA

# Ajuste dos números
pt_BR <- readr::locale(decimal_mark = ',')
aux[, 2:13] <- lapply(aux[, 2:13], function(i) readr::parse_number(i, locale = pt_BR))
remove(pt_BR)

# Exclusão das colunas que não serão utilizadas na junção
# Eu sei, poderia ter excluido de início, mas queria ver a transformação acontecer
aux <- aux[, c(1, 13)]

# Merge da aux com a PE (População por Estado)
PE <- merge(PE, aux)
```

```r
# Remoção da tabela auxiliar para liberar memória
remove(aux)

# Estado atual da tabela PE
datatable(PE[order(PE$nome_uf), c(1:3)],
          colnames = c('UF', 'População em 2000', 'População em 2010'),
          caption = 'Tabela 3: População por Estado em 2000 e 2010.')
```

---

**Base PIB Município 2005 e 2010**

Para a solução deste item usamos dados do IBGE, uma planilha de Excel que já forneceu dados do ano 2005 (solução desta questão) e 2010 (solução da próxima questão).

Como feito anteriormente, para garantir que esse *script* irá rodar em qualquer computador, a base de dados foi colocada no meu Github pessoal na forma descompactada sem qualquer alteração feita via Excel; todos os ajustes serão feitos via R.

```r
# Leitura da base do PIB por Município de 2005 na Internet
url <- 'https://github.com/thyarles/thyarles.github.io/blob/master/est0001/pib_municipios_2005_2010_xls
httr::GET(url, httr::write_disk(pib <- tempfile(fileext = '.xls')))
```

```
## Response [https://raw.githubusercontent.com/thyarles/thyarles.github.io/master/est0001/pib_municipios
##   Date: 2021-03-18 04:15
##   Status: 200
##   Content-Type: application/octet-stream
##   Size: 8.86 MB
## <ON DISK>  /tmp/RtmpLPcBOG/file1c2e9166c90d7.xls
```

```r
aux <- readxl::read_xls(pib)
remove(url, pib)

# Mantém-se apenas as colunas de interesse
aux <- aux[, c(1:5, 16)]

# Ajusta-se valores numéricos
pt_BR <- readr::locale(decimal_mark = ',')
aux[, c(1, 2, 4)] <- lapply(aux[, c(1, 2, 4)], function(i) readr::parse_number(i, locale = pt_BR))
remove(pt_BR)

# Configura-se o nome das colunas
colnames(aux) <- c('ano_pib', 'id_uf', 'nome_uf', 'id_mun', 'municipio', 'pib')

# Verifica-se os dados
str(aux)
```

```
## tibble [33,386 x 6] (S3: tbl_df/tbl/data.frame)
##  $ ano_pib  : num [1:33386] 2005 2005 2005 2005 2005 ...
##  $ id_uf    : num [1:33386] 11 11 11 11 11 11 11 11 11 11 ...
##  $ nome_uf  : chr [1:33386] "Rondônia" "Rondônia" "Rondônia" "Rondônia" ...
##  $ id_mun   : num [1:33386] 1100015 1100023 1100031 1100049 1100056 ...
##  $ municipio: chr [1:33386] "Alta Floresta D'Oeste" "Ariquemes" "Cabixi" "Cacoal" ...
##  $ pib      : num [1:33386] 186967 700915 77122 751181 146866 ...
```

```r
# Gera-se base para os PIBs de 2005
PM2005 <- aux[aux$ano_pib == 2005,]
```

```r
colnames(PM2005)[6] <- 'pib_2005'
PM2005$ano_pib <- NULL
datatable(PM2005, caption = 'Tabela 4: PIB dos municípios em 2005.')
```

```r
# Gera-se base para os PIBs de 2010
PM2010 <- aux[aux$ano_pib == 2010,]
colnames(PM2010)[6] <- 'pib_2010'
PM2010$ano_pib <- NULL
datatable(PM2010, caption = 'Tabela 5: PIB dos municípios em 2010.')
```

```r
# Libera auxiliar para liberar memória
remove(aux)
```

---

**Base Estados**

Para a solução deste item usamos dados do site Oobj, com os Estados. Para essa tarefa montaremos uma tabela que relacione UF, código da UF e nome da UF — será útil nas junções futuras.

```r
# Configuracao da URL e elemento da página
url <- 'https://www.oobj.com.br/bc/article/quais-os-c%C3%B3digos-de-cada-uf-no-brasil-465.html'
xp  <- '//*[@id="ARTICLECONTENT"]/article/table'

# Coleta dos dados
UF <- htmltab(doc = url, which = xp, rm_whitespace = TRUE)
remove(url, xp)

# Ajuste dos titulos
colnames(UF) <- c('id_uf', 'nome_uf', 'uf')

# Resultados
datatable(UF[order(UF$nome_uf), c(1:3)],
          colnames = c('ID', 'Nome', 'Sigla'),
          caption = 'Tabela 6: Estados com Código, Sigla e Nome.')
```

---

**Base ranking IDH 2010**

Os dados para a solução do Ranking do IDH de 2010 foram disponibilizados pelas Nações Unidas (UNDP).

```r
url <- 'https://www.br.undp.org/content/brazil/pt/home/idh0/rankings/idhm-municipios-2010.html'
xp  <- '//table[@class="tableizer-table"]'
IDH2010 <- htmltab(doc = url, which = xp, rm_whitespace = TRUE)
remove(url, xp)

# Comentário de mim para mim mesmo, ignore
# Para acelerar a geração deste documento, os dados do comando anterior foram salvos com o comando
# save(idh_2010, file = "idh_2010.RData")
# Quando for rodar na sua maquina, descomente a linha que lê o arquivo e comente a seguinte
# Leitura dos dados salvos
# load("idh_2010.RData")

# Renomeia-se as colunas
colnames(IDH2010) <- c('idh_rank', 'municipio', 'idh_geral', 'idh_renda', 'idh_longev', 'idh_educ')
```

```
# Converte-se a coluna Ranking IDHM 2010 para número inteiro
IDH2010$idh_rank <- readr::parse_number(IDH2010$idh_rank)

# Limpa todos as colunas IDHM para que não tenham lixo (ex. 'td>')
IDH2010[3:6] <- lapply(IDH2010[3:6], function(i) stringr::str_extract(i, '[\\d]+,[\\d]+'))

# Faz a conversão das colunas IDHM
pt_BR <- readr::locale(decimal_mark = ",")
IDH2010[3:6] <- lapply(IDH2010[3:6], function(i) readr::parse_double(i, locale = pt_BR))
remove(pt_BR)

# Transforma o Estado em parêntesis em nova coluna
IDH2010$uf <- stringr::str_extract(stringr::str_extract(IDH2010$municipio,
                                                        '\\ \\(.*\\)'), '[A-Z][A-Z]')

# Retira-se o Estado em parentesis da coluna Município
IDH2010$municipio <- stringr::str_remove(IDH2010$municipio, '\\ \\(.*\\)')

# Adiciona-se código da UF na tabela aux
IDH2010 <- merge(IDH2010, UF)

# Remove-se a coluna id_uf para evitar duplicação no resultado final
IDH2010$id_uf <- NULL

# Mostra estado final da tabela
datatable(IDH2010[order(IDH2010$idh_rank), c('nome_uf', 'idh_rank', 'municipio', 'idh_geral',
                                             'idh_renda', 'idh_longev', 'idh_educ')],
          colnames = c('ID', 'UF', 'Rank IDH', 'Município', "IDH Geral", "IDH Renda",
                       "IDH Longevidade", "IDH Educação"),
          caption = 'Tabela 7: IDH de 2010, com ranking, dos Municípios.')
```

---

---

### ==> Solução da V10

**Variação da população entre 2000 e 2010 por Estado** Para ver a variação da população vamos apenas criar uma nova coluna na base PE (População por Estado) criada anteriormente.

```
# Insere variação bruta (população de 2010 - população de 2000)
PE$variacao <- PE$pop2010 - PE$pop2000

# Insere variação percentual
PE$varPercentual <- (PE$pop2010 / PE$pop2000 - 1)

# Impressão dos resultados
datatable(PE, colnames = c('ID', 'UF', 'População em 2000', 'População em 2010',
                           'Variação', 'Variação percentual'),
          caption = 'Tabela 8: Variação da população entre 2000 e 2010 por Estado') %>%
  formatPercentage('varPercentual', 2) %>%
  formatCurrency(c('pop2000', 'pop2010', 'variacao'), currency = "", digits = 0, interval = 3, mark = "
```

### ==> Solução da V11

**Variação do PIB entre 2005 e 2010** Para ver a variação do PIB entre 2005 e 2010 deve-se criar tabelas temporárias, de forma a fazer o merge da PM2005 e PM2010. Deve-se observar que o número de municípios são distintos, então deve-se cuidar para perder o mínimo de dado possível.

```
# Verificação da quantidade de dados em cada tabela necessária
dim(PM2005)
```

```
## [1] 5564    5
```

```
dim(PM2010)
```

```
## [1] 5565    5
```

```
# Existe um município a mais no PM2010
# Como pode ser um município criado depois de 2005, vou excluir para simplificar
# Para encontrar a falha, usaremos um loop e excluiremos o município extra
temFalha <- TRUE
while (temFalha) {
  for (i in 1:nrow(PM2010)) {
    temFalha <- FALSE
    if (PM2010$id_mun[i] != PM2005$id_mun[i]) {
      cat('Achei, linha', i)
      # Remove-se o município
      cat(' ... excluindo município de ID', PM2010$id_mun[i])
      PM2010 <- PM2010[-c(i),]
      temFalha <- TRUE
      break
    }
    if (i == nrow(PM2010)) cat(', tudo verificado!')
  }
}
```

```
## Achei, linha 803 ... excluindo município de ID 2206720, tudo verificado!
```

```
remove(i, temFalha)

# Agora que as colunas têm exatamente o mesmo número de linhas e dados,
# procede-se com a junção dos PIBs de 2005 e 2010, criando a PM (PIB por Município)
PM <- cbind(PM2005, PM2010[5])

# Insere variação
PM$variacao <- PM$pib_2010 - PM$pib_2005

# Insere variação percentual
PM$varPercentual <- (PM$pib_2010 / PM$pib_2005 - 1)

# Impressão dos resultados
datatable(PM[, 2:8], colnames = c('UF', 'Código do Município', 'Município', 'PIB 2005',
                                  'PIB 2010', 'Variação', 'Variação percentual'),
          caption = 'Tabela 9: Variação do PIB entre 2005 e 2010 nos Municípios') %>%
  formatPercentage('varPercentual', 2) %>%
  formatCurrency(c('pib_2005', 'pib_2010', 'variacao'), currency = '', digits = 0,
                 interval = 3, mark = '.')
```

### ==> Solução da V12

**Categorias do PIB em 2010** De acordo com o comando da questão, os PIBs dos Municípios, no ano de 2010, devem ser classifcados como Muito baixo; Baixo; Médio; Alto e a divisão deve ser feita de tal forma que as 4 categorias tenham o mesmo número de municípios (ou próximo disso).

Para a solução do problema, utilizaremos somente a base de dados `PM2010` e separaremos os PIBs de 2010 por quartis.

```
# Obtém-se os quartis para a coluna pib_2010
quartis <- quantile(PM2010$pib_2010)

# Cria-se as categorias
PM2010$pib_cat[PM2010$pib_2010 <  quartis[2]] <- 'Muito baixo'
PM2010$pib_cat[PM2010$pib_2010 >= quartis[2] &
               PM2010$pib_2010 <  quartis[3]] <- 'Baixo'
PM2010$pib_cat[PM2010$pib_2010 >= quartis[3] &
               PM2010$pib_2010 <  quartis[4]] <- 'Médio'
PM2010$pib_cat[PM2010$pib_2010 >  quartis[4]] <- 'Alto'
remove(quartis)

# Verificação da separação
table(PM2010$pib_cat)
```

```
##
##        Alto       Baixo       Médio Muito baixo
##        1391        1391        1391        1391
```

```
# Impressão dos resultados
datatable(PM2010[, 2:6], colnames = c('UF', 'Código do Município', 'Município',
                                      'PIB 2010', 'Classificação do PIB'),
          caption = 'Tabela 11: Categorias do PIB em 2010.') %>%
  formatCurrency(c('pib_2010'), currency = '', digits = 0,
                 interval = 3, mark = '.')
```

### ==> Produção da Base Final

Como último passo ficou a correlação de variáveis obtidas em passos anteriores. Ao contrário do que foi pedido, eu não gerei uma única tabela com todos os dados porque **ela não era necessária para responder às questões**; se eu tivesse unido de início a tablea das taxas de homícios de 2010 `TH2010` com a de PIB nos Municípios `PM`, nós teríamos perdido os dados de 72 municípios nas soluções anteriores... da maneira que eu fiz, perdemos do dado de apenas um do PIB de 2010.

No entanto, para a solução desta questão, será necessário fazer as várias junções. Ou seja, nesse passo ter-se-á o arquivo com todos os dados, conforme esperado.

```
# Como ditro na introdução, iremos perder cerca de 72 municípios na junção
# Para contornar, iremos utilizar uma tabela auxiliar para as junções
dim(IDH2010)
```

```
## [1] 5565    8
```

```
dim(TH2010)
```

```
## [1] 5493    3
```

```
# Vamos tentar inserir o ID_mun na tabela aux por meio da tabela PM
aux <- merge(IDH2010, PM, by = c('nome_uf', 'municipio'))
```

```
# Vamos renomear dois campos, para não ficar confuso
colnames(aux)[13] <- 'pib_var'
colnames(aux)[14] <- 'pib_var_perc'

# Como a IDH2010 tem mais observações, vou usá-la como principal
# Faz-se o merge da taxa de homicídios 2010 com o IDH 2010 na BF (base final)
BF <- merge(aux, TH2010[, c(1, 3)], by = 'id_mun')
remove(aux)

# A tabela está completa, traz taxa de homicídio, idh e variação do PIB
# Vamos imprimir somente os campos de interesse para evitar poluicão
datatable(BF[, c(2, 3, 15, 6, 13)], colnames = c('UF', 'Município', 'Taxa Homicídio 2010',
                                                 'IDHM Geral 2010', 'Variacão do PIB 2005/2010'),
          caption = 'Tabela 12: Tabela final com todos os dados unidos.') %>%
  formatCurrency(c('taxaHom2010', 'idh_geral', 'pib_var'), currency = '', digits = 2,
                 interval = 3, mark = ',')
```

**==> Correlação**

Por fim, resta fazer a correlação entre as variáveis V1, V6 e V11

```
# Correlação usando Pearson
cor(BF[, c('taxaHom2010', 'idh_geral', 'pib_var')], method = 'pearson', use = 'complete.obs')
```

```
##             taxaHom2010   idh_geral    pib_var
## taxaHom2010  1.00000000 -0.05741233 0.04714794
## idh_geral   -0.05741233  1.00000000 0.13605628
## pib_var      0.04714794  0.13605628 1.00000000
```

```
# Correlação usando Kendall
cor(BF[, c('taxaHom2010', 'idh_geral', 'pib_var')], method = 'kendall', use = 'complete.obs')
```

```
##             taxaHom2010   idh_geral   pib_var
## taxaHom2010  1.00000000 -0.03792023 0.2152987
## idh_geral   -0.03792023  1.00000000 0.3077920
## pib_var      0.21529874  0.30779202 1.0000000
```

```
# Correlação usando Spearman
cor(BF[, c('taxaHom2010', 'idh_geral', 'pib_var')], method = 'spearman', use = 'complete.obs')
```

```
##             taxaHom2010  idh_geral   pib_var
## taxaHom2010   1.0000000 -0.0579115 0.3077241
## idh_geral    -0.0579115  1.0000000 0.4463227
## pib_var       0.3077241  0.4463227 1.0000000
```

A ideia da correlação foi legal, falta agora assistir a aula para entender exatamente o que cada resultado quer dizer!