

# R Finals

[Code ▾](#)

Shai Vaz, Maisa Fraiz

## Introdução

O seguinte projeto é o trabalho final da disciplina de *Introdução ao R Aplicado em Ciência de Dados*, da EPGE.

Utilizaremos dados estatísticos da Secretaria de Segurança Pública do Estado de São Paulo, disponíveis para download em <https://basedosdados.org/dataset/br-sp-gov-ssp> (<https://basedosdados.org/dataset/br-sp-gov-ssp>), e a planilha de informações dos municípios do estado de São Paulo, elaborada pelo IBGE e disponível em <https://www.ibge.gov.br/cidades-e-estados/sp/sao-paulo.html> (<https://www.ibge.gov.br/cidades-e-estados/sp/sao-paulo.html>). Ambos com acessos em 07/05/2022. Os dois arquivos .csv baixados já estão no repositório GitHub desse projeto, e portanto, tendo-o clonado na sua máquina, não é necessário realizar o download nas fontes.

## Preparação

### Importando os pacotes

[Hide](#)

```
# Descomente se você não tiver os pacotes:
# install.packages("tidyverse")
# install.packages("psych")
# devtools::install_github("teunbrand/ggh4x")

library(tidyverse)
library(readxl) #apesar de ser baixado no pacote "tidyverse", precisamos carregar separadamente
library(psych) #usamos para a função describe
library(lubridate) # tambem parte do tidyverse
library(ggh4x)
```

### Importando a base de dados

Lendo duas linhas do arquivo para verificar o formato do CSV (evitando assim erros com o delimitador no momento de salvar os dados, por exemplo)

[Hide](#)

```
read_lines("ocorrencias_registradas.csv", n_max = 2)
```

```
[1] "ano,mes,id_municipio,regiao_ssp,homicidio_doloso,numero_de_vitimas_em_homicidio_doloso,homicidio_doloso_por_acidente_de_transito,numero_de_vitimas_em_homicidio_doloso_por_acidente_de_transito,homicidio_culposo_por_acidente_de_transito,homicidio_culposo_outros,tentativa_de_homicidio,lesao_corporal_seguida_de_morte,lesao_corporal_dolosa,lesao_corporal_culposa_por_acidente_de_transito,lesao_corporal_culposa_outras,latrocinio,numero_de_vitimas_em_latrocinio,total_de_estupro,estupro,estupro_de_vulneravel,total_de_roubo_outros,roubo_outros,roubo_de_veiculo,roubo_a_banco,roubo_de_carga,furto_outros,furto_de_veiculo"
[2] "2002,1,3500105,Presidente Prudente,0,0,0,0,0,0,0,24,13,2,0,0,0,,,0,0,0,21,0"
```

Já podemos reparar que teremos dados NA na nossa base, porém resolveremos isso mais a diante. Como o separador é vírgula, podemos utilizar o leitor de CSV padrão.

[Hide](#)

```
ssp <- read_csv("ocorrencias_registradas.csv")
```

```
Rows: 153876 Columns: 27
-- Column specification -----
Delimiter: ","
chr  (1): regiao_ssp
dbl (26): ano, mes, id_municipio, homicidio_doloso, numero_de_vitimas_em_hom...

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Também utilizaremos a planilha do IBGE com os municípios do estado de São Paulo. Analisando os dados, percebemos que as duas primeiras linhas não contém dados, e após a linha 645, temos metadados e fontes da base. Também reparamos que o símbolo "-" é utilizado para representar NA. Todos esses detalhes foram considerados ao ler o arquivo no projeto.

[Hide](#)

```
mun <- read_excel("sp-municipios.xlsx", na="-", skip = 2, n_max = 645)
```

### Primeiras olhadas nos DFs

Vamos verificar se estamos utilizando dataframes do tipo Tibble.

[Hide](#)

```
class(ssp)
```

```
[1] "spec_tbl_df" "tbl_df"      "tbl"        "data.frame"
```

Hide

```
class(mun)
```

```
[1] "tbl_df"      "tbl"        "data.frame"
```

## Dataframe principal (SSP)

Vamos qverificar os nomes das colunas e o tipo do dado de cada uma.

Hide

```
spec(ssp)
```

```
cols(
  ano = col_double(),
  mes = col_double(),
  id_municipio = col_double(),
  regioao_ssp = col_character(),
  homicidio_doloso = col_double(),
  numero_de_vitimas_em_homicidio_doloso = col_double(),
  homicidio_doloso_por_acidente_de_transito = col_double(),
  numero_de_vitimas_em_homicidio_doloso_por_acidente_de_transito = col_double(),
  homicidio_culposo_por_acidente_de_transito = col_double(),
  homicidio_culposo_outros = col_double(),
  tentativa_de_homicidio = col_double(),
  lesao_corporal_seguida_de_morte = col_double(),
  lesao_corporal_dolosa = col_double(),
  lesao_corporal_culposa_por_acidente_de_transito = col_double(),
  lesao_corporal_culposa_outras = col_double(),
  latrocinio = col_double(),
  numero_de_vitimas_em_latrocinio = col_double(),
  total_de_estupro = col_double(),
  estupro = col_double(),
  estupro_de_vulneravel = col_double(),
  total_de_roubo_outros = col_double(),
  roubo_outros = col_double(),
  roubo_de_veiculo = col_double(),
  roubo_a_banco = col_double(),
  roubo_de_carga = col_double(),
  furto_outros = col_double(),
  furto_de_veiculo = col_double()
)
```

Vamos imprimir os nomes das colunas do dataframe.

Hide

```
names(ssp)
```

```
[1] "ano"
[2] "mes"
[3] "id_municipio"
[4] "regiao_ssp"
[5] "homicidio_doloso"
[6] "numero_de_vitimas_em_homicidio_doloso"
[7] "homicidio_doloso_por_acidente_de_transito"
[8] "numero_de_vitimas_em_homicidio_doloso_por_acidente_de_transito"
[9] "homicidio_culposo_por_acidente_de_transito"
[10] "homicidio_culposo_outros"
[11] "tentativa_de_homicidio"
[12] "lesao_corporal_seguida_de_morte"
[13] "lesao_corporal_dolosa"
[14] "lesao_corporal_culposa_por_acidente_de_transito"
[15] "lesao_corporal_culposa_outras"
[16] "latrocinio"
[17] "numero_de_vitimas_em_latrocinio"
[18] "total_de_estupro"
[19] "estupro"
[20] "estupro_de_vulneravel"
[21] "total_de_roubo_outros"
[22] "roubo_outros"
[23] "roubo_de_veiculo"
[24] "roubo_a_banco"
[25] "roubo_de_carga"
[26] "furto_outros"
[27] "furto_de_veiculo"
```

E olharemos os primeiros 6 registros de nossa tabela.

Hide

head(ssp)

ano	...	id_municipio	regiao_ssp	homicidio_doloso	numero_de_vitimas_em_homicidio_doloso
<dbl>	<dbl>	<dbl>	<chr>	<dbl>	<dbl>
2002	1	3500105	Presidente Prudente	0	0
2002	2	3500105	Presidente Prudente	0	0
2002	3	3500105	Presidente Prudente	0	0
2002	4	3500105	Presidente Prudente	0	0
2002	5	3500105	Presidente Prudente	0	0
2002	6	3500105	Presidente Prudente	0	0

6 rows | 1-6 of 27 columns

Por fim, daremos uma olhada em dados estatísticos (desvio padrão, média, mediana, mínimo, máximo e outros) de nossas colunas.

Hide

describe(ssp)

	vars	n	mean	sd
	<dbl>	<dbl>	<dbl>	<dbl>
ano	1	153876	2011.50	5.77
mes	2	153876	6.50	3.45
id_municipio	3	153876	3528670.39	16732.51
regiao_ssp*	4	153876	7.08	3.50
homicidio_doloso	5	151296	0.67	6.78
numero_de_vitimas_em_homicidio_doloso	6	151296	0.71	7.21
homicidio_doloso_por_acidente_de_transito	7	143700	0.00	0.07
numero_de_vitimas_em_homicidio_doloso_por_acidente_de_transito	8	143700	0.00	0.09
homicidio_culposo_por_acidente_de_transito	9	151296	0.54	2.36
homicidio_culposo_outros	10	151296	0.03	0.32

1-10 of 27 rows | 1-5 of 13 columns

Previous 1 2 3 Next

A análise desses dados estatísticos não faz sentido para todas as variáveis (por exemplo, não faz sentido calcular a média, desvio padrão, etc, de ano e mês) e, como há muitos dados, para outras faria mais sentido fazer uma investigação anual (é muito mais relevante comparar a média de crimes por ano e se ela está crescendo ou diminuindo do que apenas ver a média do dataset completo). Análises mais complexas serão feitas na seção de Visualização de Dados do projeto.

## Dataframe secundário (MUN)

Seguiremos o mesmo processo para esse dataframe.

Hide

names(mun)

```
[1] "Município [-]"
[2] "Código [-]"
[3] "Gentílico [-]"
[4] "Prefeito [2021]"
[5] "Área Territorial - km² [2021]"
[6] "População estimada - pessoas [2021]"
[7] "Densidade demográfica - hab/km² [2010]"
[8] "Escarlarização <span>6 a 14 anos</span> - % [2010]"
[9] "IDHM <span>Índice de desenvolvimento humano municipal</span> [2010]"
[10] "Mortalidade infantil - óbitos por mil nascidos vivos [2020]"
[11] "Receitas realizadas - R$ (×1000) [2017]"
[12] "Despesas empenhadas - R$ (×1000) [2017]"
[13] "PIB per capita - R$ [2019]"
```

Hide

head(mun)

Município [-]	Código [-]	Gentílico [-]	Prefeito [2021]
<chr>	<dbl>	<gl>	<chr>
Adamantina	3500105	NA	MARCIO CARDIM
Adolfo	3500204	NA	IZAEL ANTONIO FERNANDES
Aguaí	3500303	NA	JOSÉ ALEXANDRE PEREIRA DE ARAÚJO
Águas da Prata	3500402	NA	REGINA HELENA JANIZELO MORAES

Município [-] <chr>	Código [-] <dbl>	Gentílico [-] <lg>	Prefeito [2021] <chr>	
Águas de Lindóia	3500501	NA	GILBERTO ABDOU HELOU	
Águas de Santa Bárbara	3500550	NA	AROLDJO JOSE CAETANO	
6 rows   1-4 of 13 columns				

Hide

describe(mun)

Warning in FUN(newX[, i], ...) :  
no non-missing arguments to min; returning Inf  
Warning in FUN(newX[, i], ...) :  
no non-missing arguments to max; returning -Inf

	vars <dbl>	n <dbl>	mean <dbl>	
Município [-]*	1	645	323.00	
Código [-]	2	645	3528697.88	
Gentílico [-]	3	0	NaN	
Prefeito [2021]*	4	645	323.00	
Área Territorial - km² [2021]	5	645	384.84	
População estimada - pessoas [2021]	6	645	72324.24	
Densidade demográfica - hab/km² [2010]	7	645	301.98	
Escolarização <span>6 a 14 anos</span> - % [2010]	8	645	97.94	
IDHM <span>Índice de desenvolvimento humano municipal</span> [2010]	9	645	0.74	
Mortalidade infantil - óbitos por mil nascidos vivos [2020]	10	455	13.52	
1-10 of 13 rows   1-4 of 13 columns				
		Previous	1	2
				Next

## Limpendo os dados

Primeiro, retiramos algumas colunas que consideramos desnecessárias para a nossa análise na tabela mun (Gentílico, Prefeito, Receitas e Despesas)

Hide

```
mun2 <- mun %>%  
  select(!c(3,4,11,12))  
  
names(mun2)
```

```
[1] "Município [-]"  
[2] "Código [-]"  
[3] "Área Territorial - km² [2021]"  
[4] "População estimada - pessoas [2021]"  
[5] "Densidade demográfica - hab/km² [2010]"  
[6] "Escolarização <span>6 a 14 anos</span> - % [2010]"  
[7] "IDHM <span>Índice de desenvolvimento humano municipal</span> [2010]"  
[8] "Mortalidade infantil - óbitos por mil nascidos vivos [2020]"  
[9] "PIB per capita - R$ [2019]"
```

Depois, renomeamos as colunas por questão de simplificação, abreviando os nomes originais. Além disso, a coluna código é alterada para id\_município de forma a manter o mesmo nome nos dois DFs, para que seja possível um join.

Hide

```
colnames(mun2) <- c("nome", "id_municipio", "area", "pop", "den", "escol", "idh", "mort_inf", "pib_percap")  
names(mun2)
```

```
[1] "nome"      "id_municipio" "area"      "pop"      "den"  
[6] "escol"     "idh"          "mort_inf"  "pib_percap"
```

Para o dataset principal, podemos trocar o ID do município pelo seu nome. Mas primeiro, será que temos os mesmos municípios nos dois datasets?

Hide

```
all((unique(ssp$id_municipio)), (mun2$id_municipio))
```

```
Warning in all((unique(ssp$id_municipio)), (mun2$id_municipio)) :  
coercing argument of type 'double' to logical  
Warning in all((unique(ssp$id_municipio)), (mun2$id_municipio)) :  
coercing argument of type 'double' to logical
```

[1] TRUE

Sim! Podemos prosseguir com a limpeza do dataframe. Utilizamos a função de inner join para substituir o id do município em SSP pelo seu nome, que está presente apenas no dataset MUN2. Colocamos uma coluna nova, utilizando as colunas de ano e mes, criando uma com a data completa no formato lubridate, deixando 1 como o dia padrão. Também percebemos que a partir da data 09/2021, todas as entradas são NA, então retiramos esses registros.

Hide

```
ssp2 <- ssp %>%
  inner_join(mun2[,c("nome", "id_municipio")], by = "id_municipio") %>% # faz o inner join
  relocate("nome", .before = "id_municipio") %>% # reposiciona a coluna nome
  select(!id_municipio) %>% # remove a coluna id
  mutate(data = make_date(ano, mes, 1), .before = ano) %>% # transforma as colunas mes e ano em uma de data
  filter(data < ymd("2021-09-01")) # remove os registros após 09/2021

head(ssp2)
```

data <date>	ano <dbl>	mes <dbl>	nome <chr>	regiao_ssp <chr>	homicidio_doloso <dbl>
2002-01-01	2002	1	Adamantina	Presidente Prudente	0
2002-02-01	2002	2	Adamantina	Presidente Prudente	0
2002-03-01	2002	3	Adamantina	Presidente Prudente	0
2002-04-01	2002	4	Adamantina	Presidente Prudente	0
2002-05-01	2002	5	Adamantina	Presidente Prudente	0
2002-06-01	2002	6	Adamantina	Presidente Prudente	0

6 rows | 1-6 of 28 columns

## Glossário de Variáveis

Agora que já limpamos os dados, podemos fazer um glossário de nossas variáveis.

### Dataframe SSP2

Variável	Descrição
data	Ano e mês no formato lubridate, usando 1 como dia padrão
nome	Nome do município
regiao_ssp	Região definida pela Secretaria de Segurança Pública
homicidio_doloso	Homicídio quando se tem a intenção de matar (inclui acidente de trânsito)
numero_de_vitimas_em_homicidio_doloso	Vítimas de homicídio doloso (inclui acidente de trânsito)
homicidio_doloso_por_acidente_de_transito	Homicídio por acidentes de trânsito com intenção de matar
numero_de_vitimas_em_homicidio_doloso_por_acidente_de_transito	Vítimas de homicídio doloso em acidentes de trânsito
homicidio_culposo_por_acidente_de_transito	Homicídio por acidentes de trânsito sem intenção de matar
homicidio_culposo_outros	Homicídio culposo (exclui acidente de trânsito)
tentativa_de_homicidio	Tentativa de homicídio
lesao_corporal_seguida_de_morte	Lesão corporal dolosa seguida de homicídio culposo
lesao_corporal_dolosa	Lesão corporal dolosa sem resultado em morte
lesao_corporal_culposa_por_acidente_de_transito	Lesão corporal sem intenção de ferir ocorrida por acidente de trânsito
lesao_corporal_culposa_outras	Lesão corporal sem intenção de ferir (exclui acidente de trânsito)
latrocinio	Roubo seguido de morte
numero_de_vitimas_em_latrocinio	Vítimas de roubo seguido de morte
total_de_estupro	Soma das variáveis de estupro e estupro de vulnerável
estupro	Estupro
estupro_de_vulneravel	Estupro de vítima vulnerável
total_de_roubo_outros	Soma das variáveis roubo_outros, roubo_a_banco e roubo_de_carga
roubo_outros	Subtração patrimonial com ameaça ou violência (não inclui veículo, banco ou cargas)
roubo_de_veiculo	Roubo de veículos
roubo_a_banco	Roubo a banco
roubo_de_carga	Roubo de carga
furto_outros	Subtração patrimonial sem ameaça ou violência (não inclui veículo)

Variável	Descrição
furto_de_veiculo	Furto de veículo

## Dataframe MUN2

Variável	Descrição
nome	Nome do município do estado de São Paulo
id_municipio	ID do município no banco de dados do IBGE
area	Área territorial em km² em 2021
pop	População estimada em 2021
den	Densidade demográfica em habitantes por km² em 2010
escol	Porcentagem de escolarização em pessoas entre 6 e 14 anos em 2010
idh	IDHM - Índice de desenvolvimento humano municipal em 2010
mort_inf	Mortalidade infantil em óbitos por mil nascidos vivos em 2020
PIB_percap	PIB per capita em reais em 2019

## Análise e Visualização de dados

Hide

```
unique(ssp2$regiao_ssp)
```

```
[1] "Presidente Prudente"      "Ribeirão Preto"
[3] "Sorocaba"                  "Araçatuba"
[5] "Grande São Paulo (exclui a Capital)" "São José do Rio Preto"
[7] "Bauru"                     "Campinas"
[9] "Piracicaba"                "São José dos Campos"
[11] "Santos"                    "Capital"
```

Hide

```
maisPop4 <- mun2 %>%
  arrange(desc(pop)) %>%
  head(4) %>%
  pull(nome)

maisPop4
```

```
[1] "São Paulo"      "Guarulhos"      "Campinas"
[4] "São Bernardo do Campo"
```

## Séries Históricas

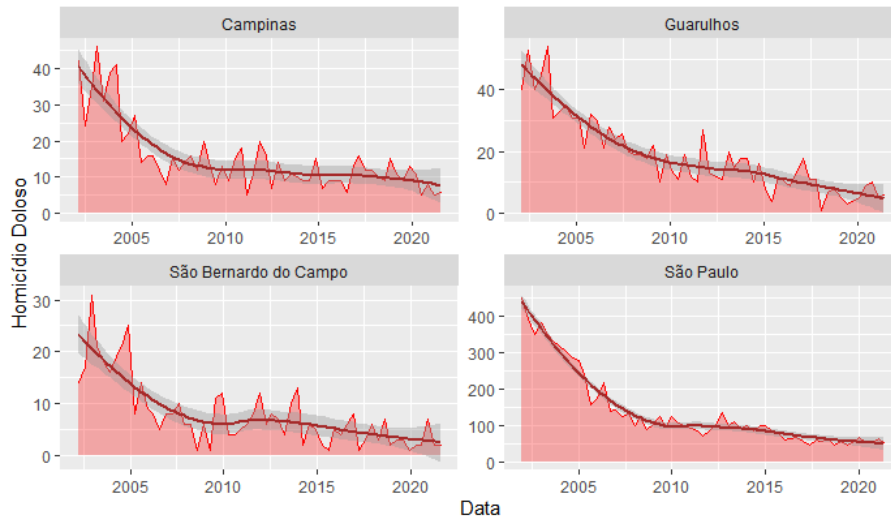
Gráfico da taxa de homicídio doloso nas 4 maiores cidades, facetado.

Hide

```
ssp2 %>%
  filter(nome == maisPop4) %>%
  ggplot(aes(x=data,y=homicidio_doloso)) +
  geom_area(color="red", fill="red", alpha=0.3) +
  geom_smooth(color="brown")+
  facet_wrap(vars(nome), scales="free") +
  labs(x = "Data", y = "Homicídio Doloso",
       title = "Homocídio doloso nas quatro maiores cidades de SP")
```

```
`geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

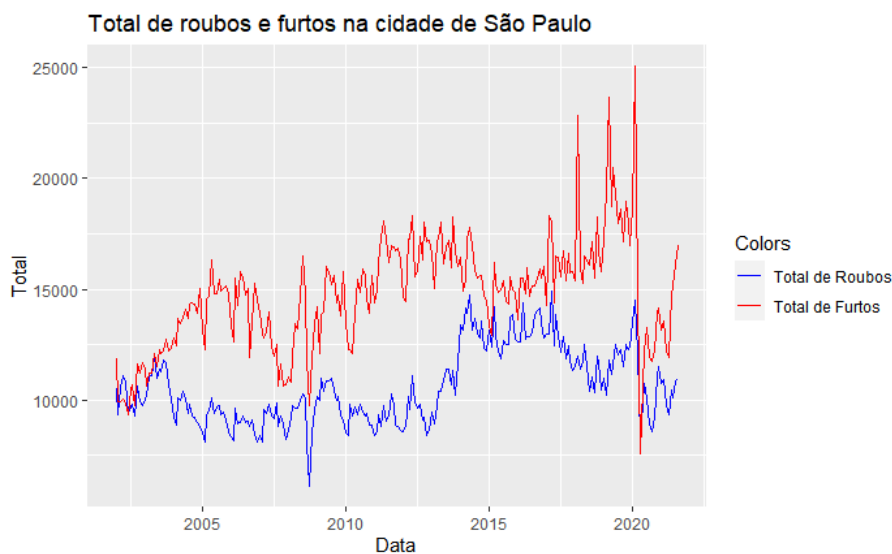
## Homocídio doloso nas quatro maiores cidades de SP



Percebemos uma tendência forte de queda da taxa de homicídios nas maiores cidades do estado quando analisamos a dinâmica de longo prazo.

Hide

```
ssp2 %>%
  filter(nome == "São Paulo") %>%
  ggplot(aes(x=data))+
  geom_line(aes(y=total_de_roubo_outros, color = "Total de Roubos"))+
  geom_line(aes(y=furto_outros, color = "Total de Furtos")) +
  labs(x = "Data", y = "Total",
        title = "Total de roubos e furtos na cidade de São Paulo") +
  scale_color_manual(name = "Colors",
                     values = c("Total de Roubos" = "blue", "Total de Furtos" = "red"))
```

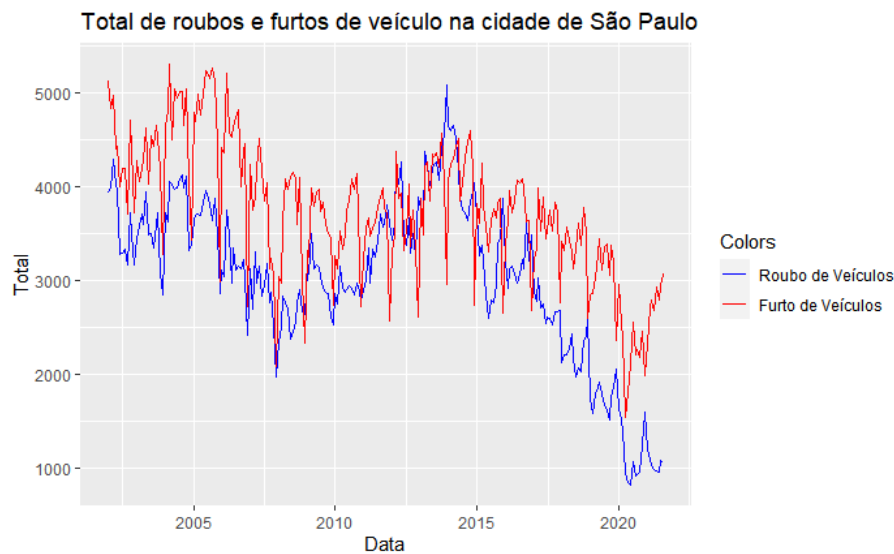


Vemos que tanto os roubos (menos numerosos) quanto os furtos (mais numerosos) têm tido uma tendência de crescimento durante a série histórica. Notamos também a queda brusca com o lockdown no início de 2020, com retorno à tendência de crescimentos após alguns meses.

Pensamos na existência de um trade-off entre furto e roubo de veículos, dada a necessidade de uma logística para desmonte, venda de peças, etc. Qual o tipo de crime mais recorrente entre os dois?

Hide

```
ssp2 %>%
  filter(nome == "São Paulo") %>%
  ggplot(aes(x=data))+
  geom_line(aes(y=roubo_de_veiculo, color = "Roubo de Veículos"))+
  geom_line(aes(y=furto_de_veiculo, color="Furto de Veículos")) +
  labs(x = "Data", y = "Total",
        title = "Total de roubos e furtos de veículo na cidade de São Paulo") +
  scale_color_manual(name = "Colors",
                     values = c("Roubo de Veículos" = "blue", "Furto de Veículos" = "red"))
```



Podemos tentar deixar os dados anualizados, para simplificar os gráficos?

Hide

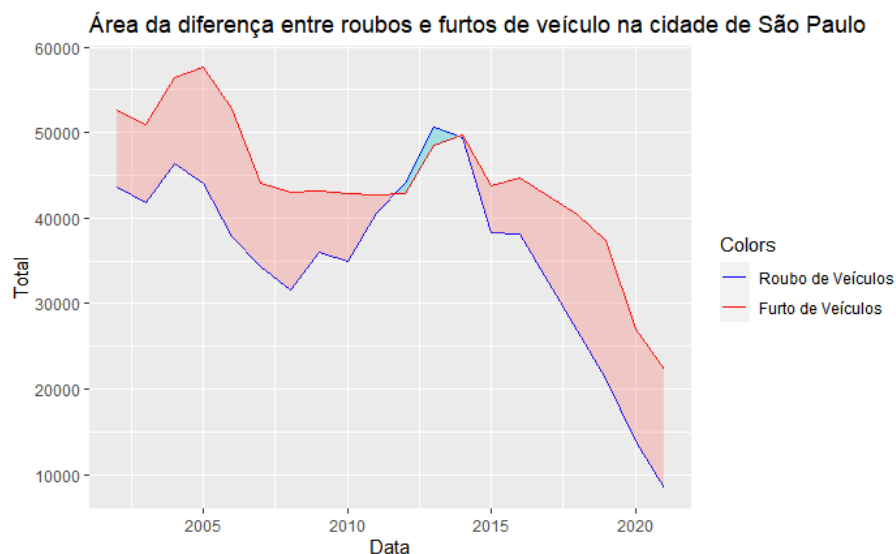
```
ssp_anual <- ssp2 %>%
  group_by(nome, ano) %>%
  summarise(across(where(is.numeric), sum)) %>%
  select(-mes)
```

`summarise()` has grouped output by 'nome'. You can override using the `.groups` argument.

Simplificamos o gráfico com as taxas anualizadas.

Hide

```
ssp_anual %>%
  filter(nome == "São Paulo") %>%
  ggplot(aes(x=ano)) +
  geom_line(aes(y=roubo_de_veiculo, color = "Roubo de Veículos")) +
  geom_line(aes(y=furto_de_veiculo, color="Furto de Veículos")) +
  labs(x = "Data", y = "Total",
       title = "Área da diferença entre roubos e furtos de veículo na cidade de São Paulo") +
  scale_color_manual(name = "Colors",
                    values = c("Roubo de Veículos" = "blue", "Furto de Veículos" = "red")) +
  ggh4x::stat_difference(aes(ymin = roubo_de_veiculo, ymax = furto_de_veiculo), alpha = 0.3, show.legend = FALSE) #da biblioteca ggh4x
```



Utilizando as taxas anualizadas, criamos outro df apenas com os dados de 2020 das variáveis furto\_outros e homicidio\_doloso, realizando um inner join com a tabela mun2 pela variável nome (semelhante ao que fizemos anteriormente para inserir o nome na tabela ssp2). Além disso, criamos novas variáveis com as taxas de roubo e homicídio doloso por 100mil habitantes.

Hide



```
ssp_taxas2020 <- ssp_anual %>%
  filter(ano == 2020) %>%
  select(furto_outros, homicidio_doloso) %>%
  inner_join(mun2, by = "nome") %>%
  mutate(furto_taxa = furto_outros/pop*100000, homicidio_taxa = homicidio_doloso/pop*100000)
```

Adding missing grouping variables: `nome`

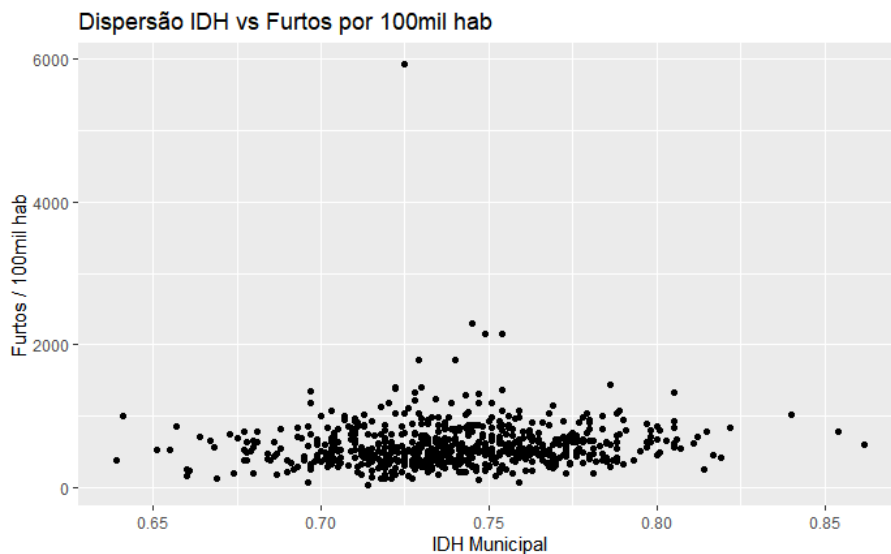
## Gráficos de Dispersão

### Furto vs IDH e PIB

Comparando furto e IDH municipal.

Hide

```
ggplot(ssp_taxas2020, aes(x=idh, y=furto_taxa)) +
  geom_point() +
  labs(x="IDH Municipal", y="Furtos / 100mil hab",
       title = "Dispersão IDH vs Furtos por 100mil hab")
```



Removemos o outlier:

Hide

```
ssp_taxas2020 %>%
  filter(furto_taxa>4000)
```

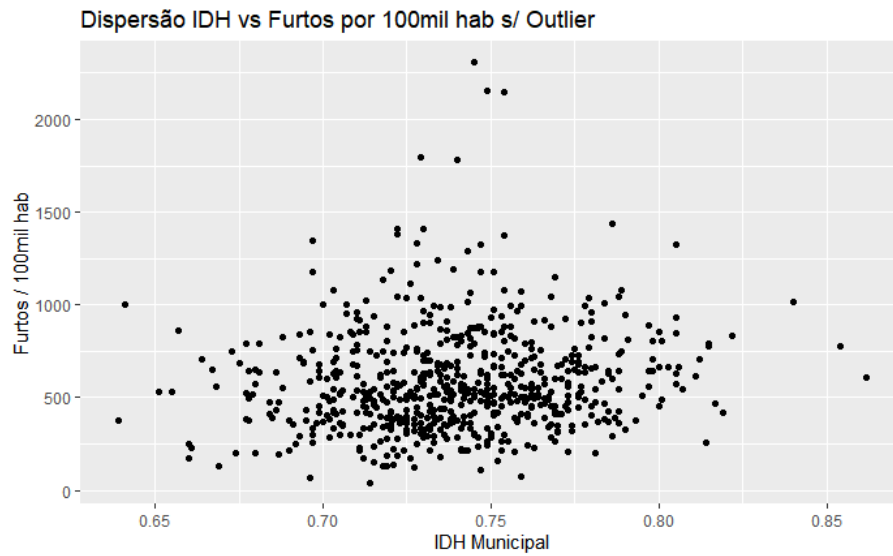
nome <chr>	furto_outros <dbl>	homicidio_doloso <dbl>	id_municipio <dbl>	area <dbl>	pop <dbl>	den <dbl>	escol <dbl>	idh <dbl>	mort_inf <dbl>
Ilha Comprida	684	0	3520426	196.567	11552	47.01	98.2	0.725	13.79

1 row | 1-10 of 13 columns

Em 2020 a cidade com maior número de roubos por 100 mil habitantes foi "Ilha Comprida".

Hide

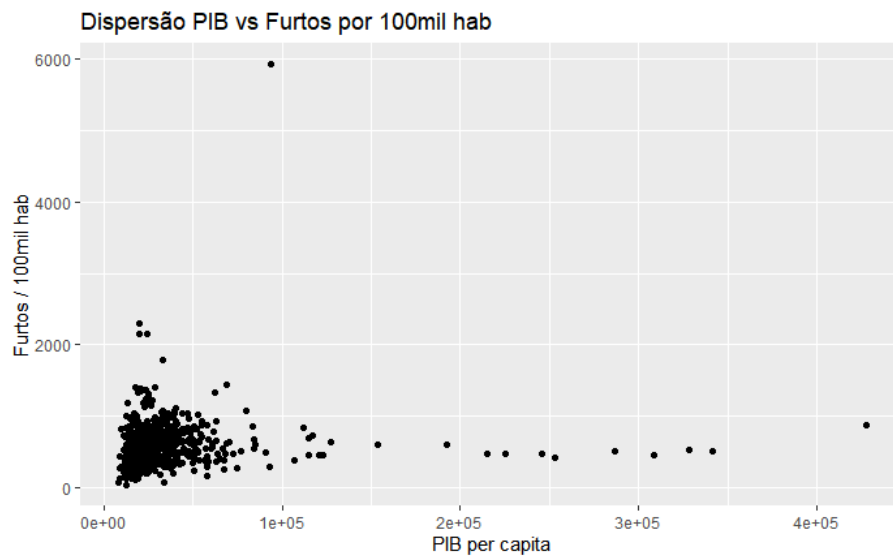
```
ssp_taxas2020 %>%
  filter(nome != "Ilha Comprida") %>%
  ggplot(aes(x=idh, y=furto_taxa)) +
  geom_point() +
  labs(x="IDH Municipal", y="Furtos / 100mil hab",
       title = "Dispersão IDH vs Furtos por 100mil hab s/ Outlier")
```



Comparamos furto contra PIB per capita.

Hide

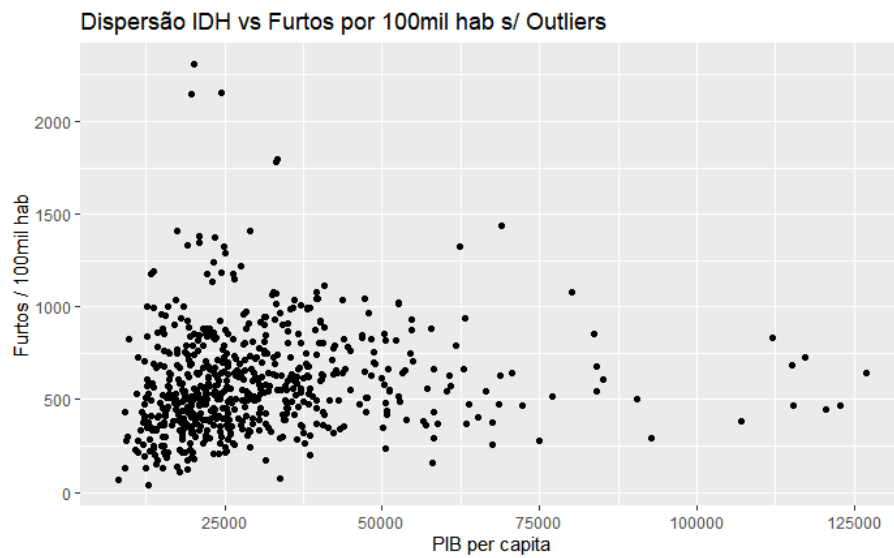
```
ggplot(ssp_taxas2020, aes(x=pib_percap, y=furto_taxa)) +
  geom_point() +
  labs(x="PIB per capita", y="Furtos / 100mil hab",
       title = "Dispersão PIB vs Furtos por 100mil hab")
```



Retiramos outliers.

Hide

```
ssp_taxas2020 %>%
  filter(pib_percap < 1.5*10^5 & furto_taxa <4000) %>%
  ggplot(aes(x=pib_percap, y=furto_taxa)) +
  geom_point() +
  labs(x="PIB per capita", y="Furtos / 100mil hab",
       title = "Dispersão IDH vs Furtos por 100mil hab s/ Outliers")
```

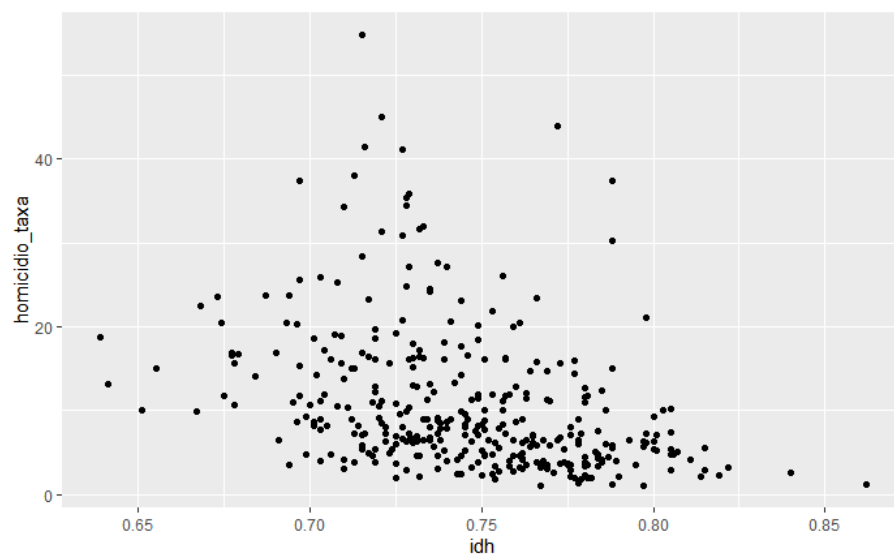


## Homicídio vs IDH e PIB

Homicídio doloso contra IDH.

Hide

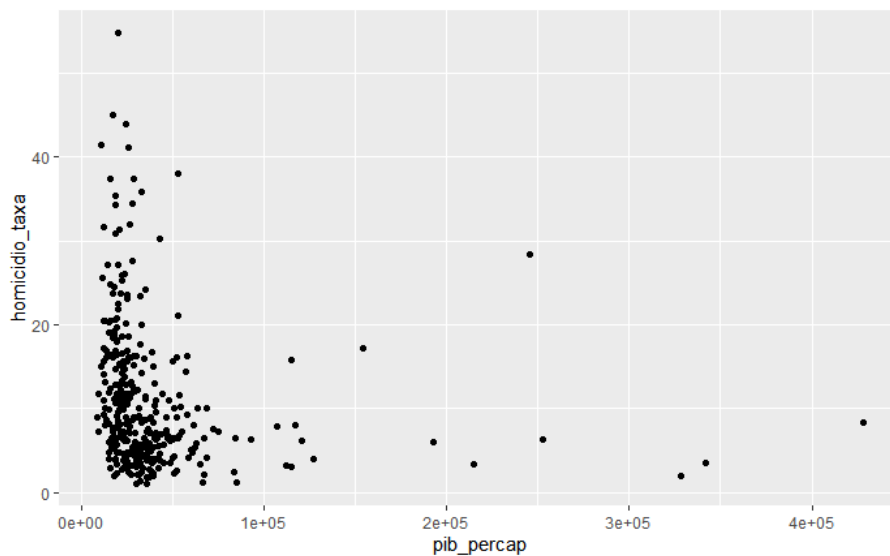
```
ssp_texas2020 %>%  
  filter(homicidio_doloso > 0) %>% #apenas os valores positivos  
  ggplot(aes(x=idh, y=homicidio_taxa)) +  
    geom_point()
```



Homicídio doloso contra PIB per capita.

Hide

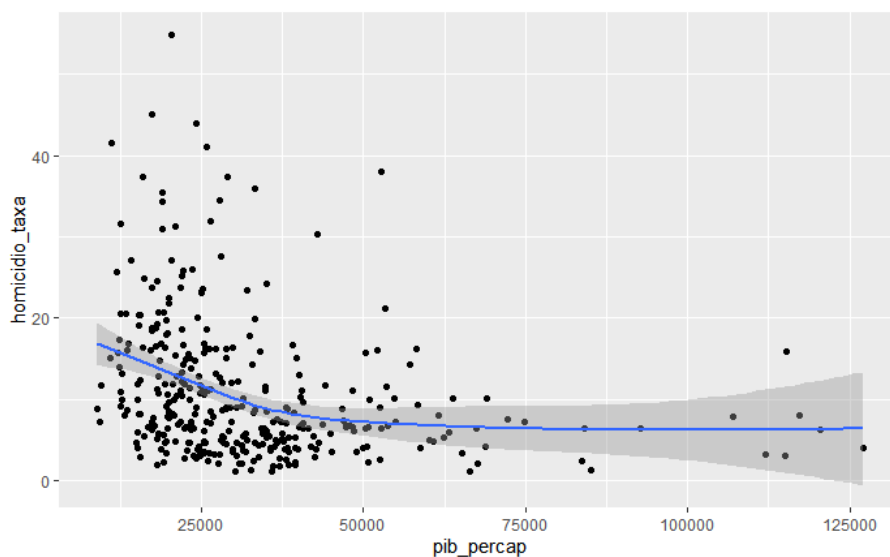
```
ssp_texas2020 %>%  
  filter(homicidio_taxa > 0) %>%  
  ggplot(aes(x=piib_percap, y=homicidio_taxa)) +  
    geom_point()
```



Hide

```
ssp_texas2020 %>%
  filter(pib_percap < 1.5*10^5 & homicidio_taxa > 0) %>%
  ggplot(aes(x=pib_percap, y=homicidio_taxa)) +
  geom_point() +
  geom_smooth(method = "gam")
```

`geom\_smooth()` using formula 'y ~ s(x, bs = "cs")'



## Referências

### Fontes de técnicas de programação em R

- [https://github.com/swirldev/swirl\\_courses/tree/master/R\\_Programming](https://github.com/swirldev/swirl_courses/tree/master/R_Programming) ([https://github.com/swirldev/swirl\\_courses/tree/master/R\\_Programming](https://github.com/swirldev/swirl_courses/tree/master/R_Programming)). Acessos durante o semestre.
- <https://github.com/sysilviakim/swirl-tidy> (<https://github.com/sysilviakim/swirl-tidy>), acessos durante o semestre.
- Wickham & Grommond, **R for Data Science**. Disponível em <https://r4ds.had.co.nz/> (<https://r4ds.had.co.nz/>). Acessos durante o semestre.
- Cheat Sheets Rstudio <https://www.rstudio.com/resources/cheatsheets/> (<https://www.rstudio.com/resources/cheatsheets/>). Utilizamos as de Ggplot2, Dplyr, Lubridate e RMarkdown.
- Aprendendo a usar Rstudio aliado ao Github <https://happygitwithr.com/index.html> (<https://happygitwithr.com/index.html>). Acesso em 30/05/2022.

### Fontes específicas de código

- Legendas para múltiplos geom\_line <https://stackoverflow.com/questions/34379268/ggplot-and-two-different-geom-line-the-legend-does-not-appear> (<https://stackoverflow.com/questions/34379268/ggplot-and-two-different-geom-line-the-legend-does-not-appear>) acesso em 13/06/2022
- Scatterplot <https://r-graph-gallery.com/272-basic-scatterplot-with-ggplot2.html> (<https://r-graph-gallery.com/272-basic-scatterplot-with-ggplot2.html>) acesso em 13/06/2022
- Time series with conditional area fill <https://r-graph-gallery.com/web-time-series-and-facetting.html> (<https://r-graph-gallery.com/web-time-series-and-facetting.html>). Acesso 13/06/2022.
- Soma de variáveis por grupo (uso do helper "where") com dplyr <https://stackoverflow.com/questions/1660124/how-to-sum-a-variable-by-group> (<https://stackoverflow.com/questions/1660124/how-to-sum-a-variable-by-group>). Acesso em 12/06/2022.

- Uso de faceting <https://r-graph-gallery.com/223-faceting-with-ggplot2.html> (<https://r-graph-gallery.com/223-faceting-with-ggplot2.html>). Acesso em 06/06/2022.
- Uso de joins <https://rpubs.com/odenipinedo/joining-data-with-dplyr> (<https://rpubs.com/odenipinedo/joining-data-with-dplyr>). Acesso em 06/06/2022.
- Uso da função `psych::describe` <https://www.rdocumentation.org/packages/psych/versions/1.0-17/topics/describe> (<https://www.rdocumentation.org/packages/psych/versions/1.0-17/topics/describe>) acesso em 06/06/2022.