

Análise espacial da criminalidade em uma grande metrópole:

Um estudo de caso para o município de São Paulo - SP

*Raul de Sá Durló**

21 fevereiro 2018

Abstract

Contents

1	Introdução e pacotes utilizados	2
2	Estado de São Paulo	3
2.1	Dados	3
2.2	Estatísticas descritivas	3
2.3	Carregando arquivo:	3
2.4	Estatísticas descritivas	5
3	Município de São Paulo	13
3.1	Dados	13
3.2	carregando arquivo:	13
3.3	Estatísticas descritivas	14
4	Análise bivariada	29

*Mestre em Economia - Unesp/FCLAr

1 Introdução e pacotes utilizados

O objetivo deste projeto é reescrever minha dissertação de mestrado em ambiente 100% R. O relatório é gerado em `Rmarkdown` e tem como *output* um arquivo `.pdf`.

Série de gráficos da taxa de homicídios por cem mil habitantes nas localidades: 1. Estado de São Paulo (Capital, Interior e Região Metropolitana - exclusive Capital), do ano 2000 até o ano 2010 e 2. na Capital (Município de São Paulo, Distritos Policiais e Seccionais), somente nos anos 2003 e 2013

Os seguintes pacotes foram utilizados:

- `readr`
- `tidyverse`
- `huxtable`
- `lubridate`
- `ggpubr`
- `ggrepel`
- `spdep`
- `wesanderson`

```
library(readr)
library(tidyverse)
library(huxtable)
library(lubridate)
library(ggpubr)
library(ggrepel)
library(treemapify)
library(spdep)
library(wesanderson)
```

2 Estado de São Paulo

2.1 Dados

Os dados referem-se ao número de ocorrências de homicídio registradas entre os anos de 2000 e 2010. Como a interpretação de ocorrências criminais é sensível à mudanças demográficas, os dados foram normalizados em relação à população residente, sendo calculado, portanto, uma taxa de homicídios por 100.000 habitantes:

$$txhomicidio_{ij} = \left(\frac{homicidio_{ij}}{populacao_{ij}} \right) 100000$$

Na equação acima, a taxa de homicídio no período i é calculada para a localidade j por 100.000 habitantes.

- Os dados de ocorrências criminais são provenientes das Estatísticas Trimestrais¹ da Secretaria Estadual de Segurança Pública do Estado de São Paulo. para esta análise os dados trimestrais foram agrupados em anos.
- Já os dados da população residente foram extraídos das estimativas utilizadas pelo Tribunal de Contas da União para determinação das cotas do Fundo de Participação dos Municípios².

2.2 Estatísticas descritivas

Principais pontos:

- As estatísticas trimestrais mostram queda significativa dos registros de homicídio.
- As localidades no interior do estado apresentam um aumento na proporção de homicídios registrados.
- A distribuição da população nas localidades permaneceram relativamente estáveis.
- A queda da taxa de homicídios é persistente em todas as localidades. há uma pequena resistência no decréscimo da taxa de homicídio no interior, com aumento a partir do ano de 2008.

2.3 Carregando arquivo:

2.3.1 Para taxas anuais de homicídio por 100mil habitantes

O código abaixo carrega dados anuais de homicídios e população por região no estado de São Paulo, onde:

- **ano** é o ano de registro,
- **população** é a contagem da população residente.
- **homicidio** é o número total de registros de homicídio doloso e
- **local** são as localidades, com:
 - *Capital*: município de São Paulo,
 - *Grande SP*: para os municípios da região Metropolitana de São Paulo (exclusive MSP),

¹<http://www.ssp.sp.gov.br/estatistica/trimestrais.aspx>

²<http://tabnet.datasus.gov.br/cgi/deftohtm.exe?ibge/cnv/poptsp.def>

- *Interior*: para os demais municípios e
- *Total*: Todo o estado de São Paulo.

```
#
# lendo .rds
estado_sp <- read_rds("C:\\Users\\raul\\Google Drive\\meu_projeto\\dados e scripts\\tabelas_output\\ta
) %>% select(seq(4))

estado_sp <- estado_sp %>%
  # agrupa para obter valores para todo o estado: "Total"
  group_by(ano) %>%
  summarise(populacao = sum(populacao),
            homicidio = sum(homicidio)) %>%
  ungroup() %>%
  mutate(local = rep("Total", 11)) %>%
  # une o agrupamento à tabela com as regiões
  bind_rows(estado_sp) %>%
  mutate(tx_homicidio = (homicidio/populacao)*100000) %>%
  glimpse()
```

```
## Observations: 44
## Variables: 5
## $ ano          <int> 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2...
## $ populacao    <int> 36351316, 37630106, 38177742, 38709320, 39825226,...
## $ homicidio    <dbl> 12638, 12475, 11847, 10954, 8753, 7076, 6057, 487...
## $ local        <chr> "Total", "Total", "Total", "Total", "Total", "Tot...
## $ tx_homicidio <dbl> 34.76628, 33.15165, 31.03117, 28.29809, 21.97853,...
```

```
#
```

2.3.2 Para números totais de homicídios, por trimestre:

O código abaixo carrega dados absolutos das Estatísticas Trimestrais da Secretaria de Segurança Pública. A variável `trimestre` apresenta valores correspondentes aos trimestres do ano de referência (p.e. `trimestre = "1"` → 1º Trimestre) e a periodicidade total é de 3º Trimestre de 1995 até 1º Trimestre de 2016:

```
#
# .rds
estado_sp_trim <- read_rds(
  "C:\\Users\\raul\\Google Drive\\meu_projeto\\dados e scripts\\tabelas_output\\tab_análise_1\\tab_trin
) %>% select(seq(4))

# agregando
estado_sp_trim <- estado_sp_trim %>%
  group_by(trimestre, ano) %>%
  summarise(homicidio = sum(homicidio)) %>%
  ungroup() %>%
  mutate(local = rep("Total", 83)) %>%
  bind_rows(estado_sp_trim) %>%
  arrange(local, ano, trimestre)
#
```

Aplicamos `lubridate::quarter()` e `lubridate::ymd()` para lidar com anos e trimestres:

```
#
trim <- quarter(
```

```

seq(ymd("1995/7/1"), ymd("2016/1/30"), by = "quarter"),
  with_year = T,
  fiscal_start = 1)

estado_sp_trim <- estado_sp_trim %>%
  mutate(data = rep(trim, 4)) %>%
  select(local, trimestre, ano, data, homicidio) %>%
  glimpse()

## Observations: 332
## Variables: 5
## $ local      <chr> "Capital", "Capital", "Capital", "Capital", "Capital..."
## $ trimestre  <dbl> 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4...
## $ ano        <int> 1995, 1995, 1996, 1996, 1996, 1996, 1997, 1997, 1997...
## $ data       <dbl> 1995.3, 1995.4, 1996.1, 1996.2, 1996.3, 1996.4, 1997...
## $ homicidio  <dbl> 1134, 1142, 1331, 1109, 1150, 1092, 1140, 1051, 1145...
#

```

2.4 Estatísticas descritivas

2.4.1 Agrupando os dados com dplyr

Agrupa-se a taxa de homicídio segundo as localidades:

```

estat_descr <- estado_sp %>%
  group_by(local) %>%
  # summarize() define as variáveis
  summarise(`Média` = mean(tx_homicidio),
            `Desvio padrão` = sd(tx_homicidio),
            Mediana = median(tx_homicidio),
            `IQR` = IQR(tx_homicidio),
            `Mínimo` = min(tx_homicidio),
            `Máximo` = max(tx_homicidio)) %>%
  ungroup() %>%
  rename("Localidade" = local)

```

2.4.2 Tabela 1: estatísticas descritivas

Cria a tabela `huxtable::hux()`:

```
ht <- hux(estat_descr, add_colnames = TRUE)
```

Para Editar a tabela, basta alterar os parâmetros no código abaixo:

```
ht <- ht                                %>%  
  set_bold(1, everywhere, TRUE)        %>% # negrito  
  set_number_format(3)                 %>% # casas decimais  
  set_top_border(1, everywhere, 1)     %>% # borda superior  
  set_bottom_border(c(1,5), everywhere, 1) %>% # borda inferior  
  set_align(everywhere, everywhere, 'center') %>% # alinhamento de texto na célula  
  set_right_padding(3)                 %>% # para posicionar  
  set_left_padding(3)                  %>% # para posicionar  
  set_width(.9)                        %>% # para posicionar no pdf  
  set_position('center')               %>% # para posicionar no pdf  
  set_caption(  
'Estatísticas descritivas - homicídios por 100.000 habitantes no Estado de São Paulo - Capital, RMSP e Interior'  
)
```

Table 1: Estatísticas descritivas - homicídios por 100.000 habitantes no Estado de São Paulo - Capital, RMSP e Interior - no período de 2000 a 2010

Localidade	Média	Desvio padrão	Mediana	IQR	Mínimo	Máximo
Capital	27.756	16.323	22.593	29.021	10.636	53.221
Grande SP	27.130	13.242	22.477	24.174	12.221	46.173
Interior	14.099	4.734	12.843	9.172	8.511	20.354
Total	20.549	9.652	17.496	18.028	10.484	34.766

2.4.3 Figura: estatísticas trimestrais

Os gráficos com dados por trimestre são gerados a partir da função `homic_trimestre()`, que aceita como argumentos as variáveis de `estado_sp_trim$local`:

```
homic_trimestre <- function(x) { # x:("Capital", "Interior", "Grande SP", "total")

  ggplot(filter(estado_sp_trim, local == x), aes(x = trim, y = homicidio, fill = local)) +
    geom_line() +
    theme(plot.title = element_text(size = 10),
          axis.text.x = element_text(angle = 90, size = 6.5, color = 'black'),
          axis.text.y = element_text(size = 6.5, color = 'black'),
          axis.line.x = element_line(size = .3),
          axis.line.y = element_line(size = .3),
          panel.background = element_blank(),
          axis.title.x = element_text(size = 7.5),
          axis.title.y = element_text(size = 7.5)) +
    labs(title = paste(x),
         y = "",
         x = "Ano") +
    geom_vline(aes(xintercept = trim[34]),
              linetype = "dashed", color = wes_palette("Cavalcanti")[5]) +
    geom_vline(aes(xintercept = trim[47]),
              linetype = "dashed", color = wes_palette("Cavalcanti")[1])
} # don't know how to add breaks and labels in x-axis
```

Utilizamos o `ggpubr::ggarrange` para enquadrar as localidades.

```
quadro_homic_trimestre <- ggarrange( # os gráficos
  homic_trimestre("Capital") +
    ylab("Taxa de homicídios"),
  homic_trimestre("Grande SP"),
  homic_trimestre("Interior") +
    ylab("Taxa de homicídios"),
  homic_trimestre("Total"),
  # colunas, linhas, alinhamento e legenda
  ncol = 2,
  nrow = 2,
  align = "hv",
  legend = "top",
  common.legend = TRUE)
```

Com `ggpubr::annotate_figure` edita-se o quadro:

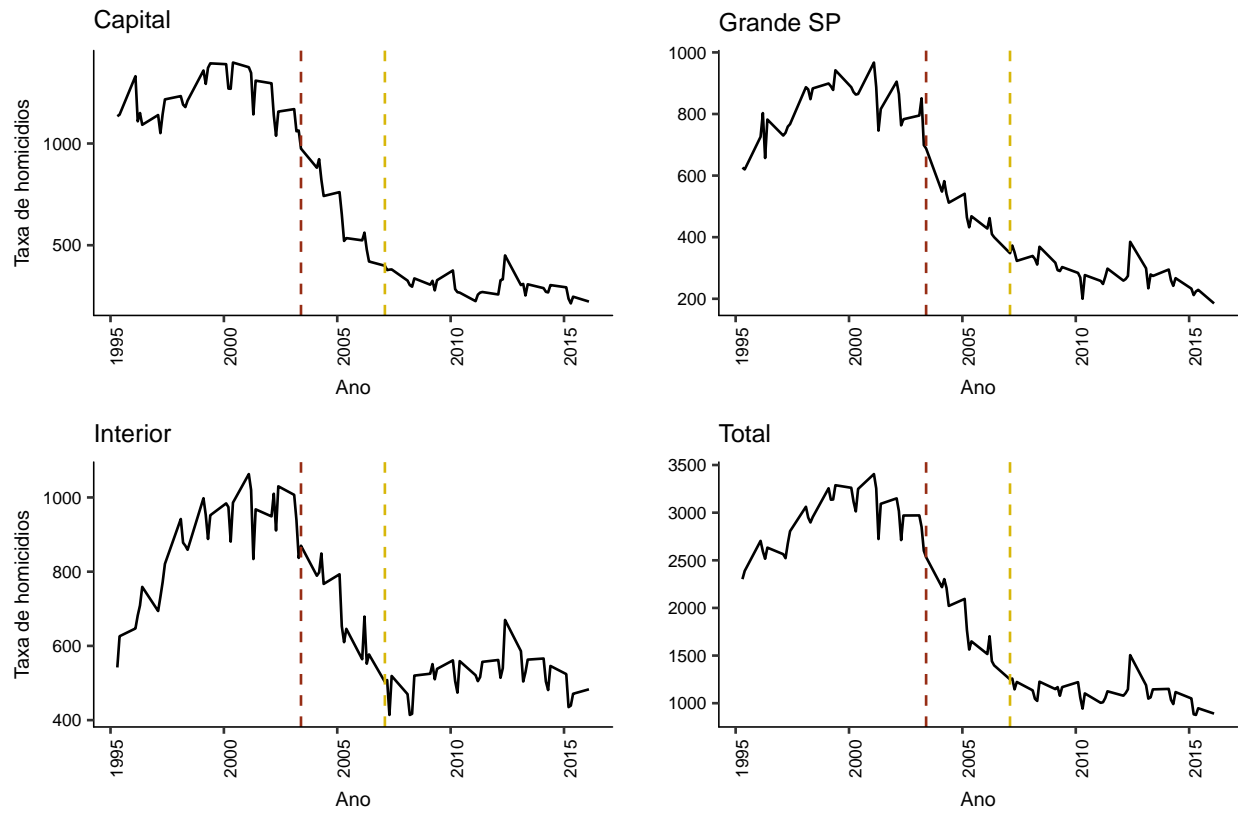
```
quadro_homic_trimestre <- annotate_figure( # A figura:
  quadro_homic_trimestre,
  # Configurando o quadro:
  top = text_grob(
    "Figura: Ocorrências de homicídios no Estado de São Paulo\n3ºTrim - 1995 até 1ºTrim - 2016",
    color = "black",
    vjust = .5,
    size = 10,
    family = "Times", just = "center"),
  bottom = NA,
```

```

left    = NA,
right   = NA,
fig.lab = NA,
fig.lab.face = NA)

```

Figura: Ocorrências de homicídios no Estado de São Paulo
3ºTrim – 1995 até 1ºTrim – 2016



2.4.4 Figura: Distribuição percentual de ocorrências de homicídio e da população residente

A função `perc(y,z)` aceita argumentos do vetor de interesse (`$homicidio` ou `$populacao`) e sua posição (2=homicídios e 3=população):

```
perc <- function(y,z) { # y: .$homicidio; .$populacao;
                        # z: 3=homicidio; 2=populacao;

  ggplot(estado_sp[-seq(11),], aes(x = as.character(ano),
                                   y = as.numeric(y),
                                   color = 'black', fill = local)) +
  geom_bar(stat = "identity", position = 'fill', alpha = .7,
          color = 'black', size = .2) +
  theme(legend.position = c(.8, .75),
        legend.title = element_text(size = 7.5),
        legend.key.size = unit(.3,"cm"),
        axis.ticks = element_blank(),
        legend.text = element_text(size = 7.5),
        plot.title = element_text(size = 8, hjust = 0.5),
        axis.text.x = element_text(angle = 90, size = 7.5, color = 'black'),
        axis.text.y = element_text(size = 7.5, color = 'black'),
        axis.line.x = element_line(),
        axis.line.y = element_blank(),
        panel.background = element_blank(),
        axis.title.x = element_text(size = 6.5),
        axis.title.y = element_text(size = 7.5)) +
  labs(x = "Ano",
       y = '',
       fill = "Região:") +
  scale_fill_manual(values = c(wes_palette("Moonrise1")))
}
```

Gerando o quadro:

```
quadro_percs <- ggarrange(perc(estado_sp[-seq(11),]$homicidio,3) +
                             ggtitle("Proporção de homicídios"),
                          perc(estado_sp[-seq(11),]$populacao,2) +
                             ggtitle("Proporção de população residente"),
                          ncol = 2,
                          nrow = 1,
                          align = "hv",
                          legend = "top",
                          common.legend = TRUE)
```

Editando o quadro:

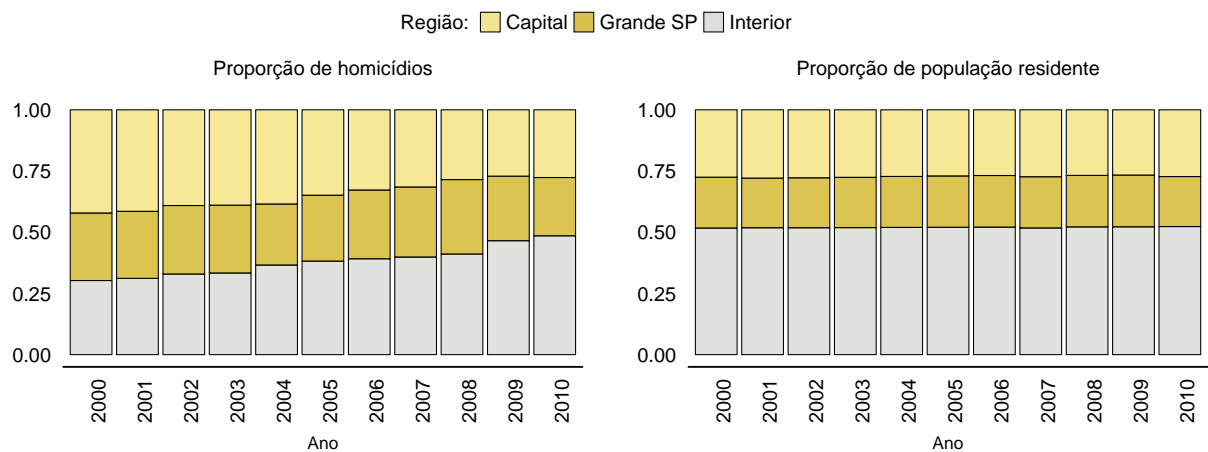
```
quadro_percs <- annotate_figure(quadro_percs,
                               top = text_grob(
"Figura: Proporção de ocorrências de homicídios e população residente por região\n(2000 até 2010)",
                               color = "black",
                               vjust = .5,
                               size = 10,
                               family = "Times", just = "center"),
                               bottom = NA,
```

```

left    = NA,
right   = NA,
fig.lab = NA,
fig.lab.face = NA)

```

Figura: Proporção de ocorrências de homicídios e população residente por região
(2000 até 2010)



2.4.5 Figura: Taxa de homicídios anuais - de 2000 até 2010

O Código abaixo faz os gráficos de taxas de homicídio anuais, novamente temos uma função(`homic_tx(x)`). Ela aceita como argumento as localidades "Capital", "Interior", "Grande SP" e "Total":

```
anual <- year(seq(as.POSIXct("2000-01-01"), by = "year", length.out = 11))

homic_tx <- function(x){

  # x: nome da Localidade
  ggplot(filter(estado_sp, local == x),
    aes(x = as.factor(anual), y = tx_homicidio, group = rep(1,11))) +
    geom_line() +
    geom_point(size = .5) +
    geom_text(aes(label = round(tx_homicidio,2)), size = 2,
      hjust = -0.1, vjust = 0, angle = 30) +
    theme(plot.title = element_text(size = 10),
      axis.text.x = element_text(angle = 90, size = 6.5, color = 'black',
        hjust = 1, vjust = .5),
      axis.text.y = element_text(size = 6.5, color = 'black'),
      axis.line.x = element_line(size = .3),
      axis.line.y = element_line(size = .3),
      panel.background = element_blank(),
      axis.title.x = element_text(size = 7.5),
      axis.title.y = element_text(size = 7.5)) +
    labs(title = paste(x, x="", y="") +
    scale_y_continuous(limits = c(0,60),
      breaks = seq(0,60, by = 15)) +
    geom_vline(aes(xintercept = as.numeric(as.factor(anual)[4])),
      linetype = "dashed", color = wes_palette("Cavalcanti")[5]) +
    geom_vline(aes(xintercept=as.numeric(as.factor(anual)[7])),
      linetype = "dashed", color = wes_palette("Cavalcanti")[1])

}
```

Para gerar o quadro, analogamente às figuras anteriores:

```
quadro_homic_tx <- ggarrange(homic_tx("Capital") +
  ylab("Taxa de Homicídios"),
  homic_tx("Grande SP"),
  homic_tx("Interior") +
  ylab("Taxa de Homicídios") + xlab("Ano"),
  homic_tx("Total") + xlab("Ano"),
  ncol = 2,
  nrow = 2,
  align = "hv",
  legend = "top",
  common.legend = TRUE)
```

Editando o quadro:

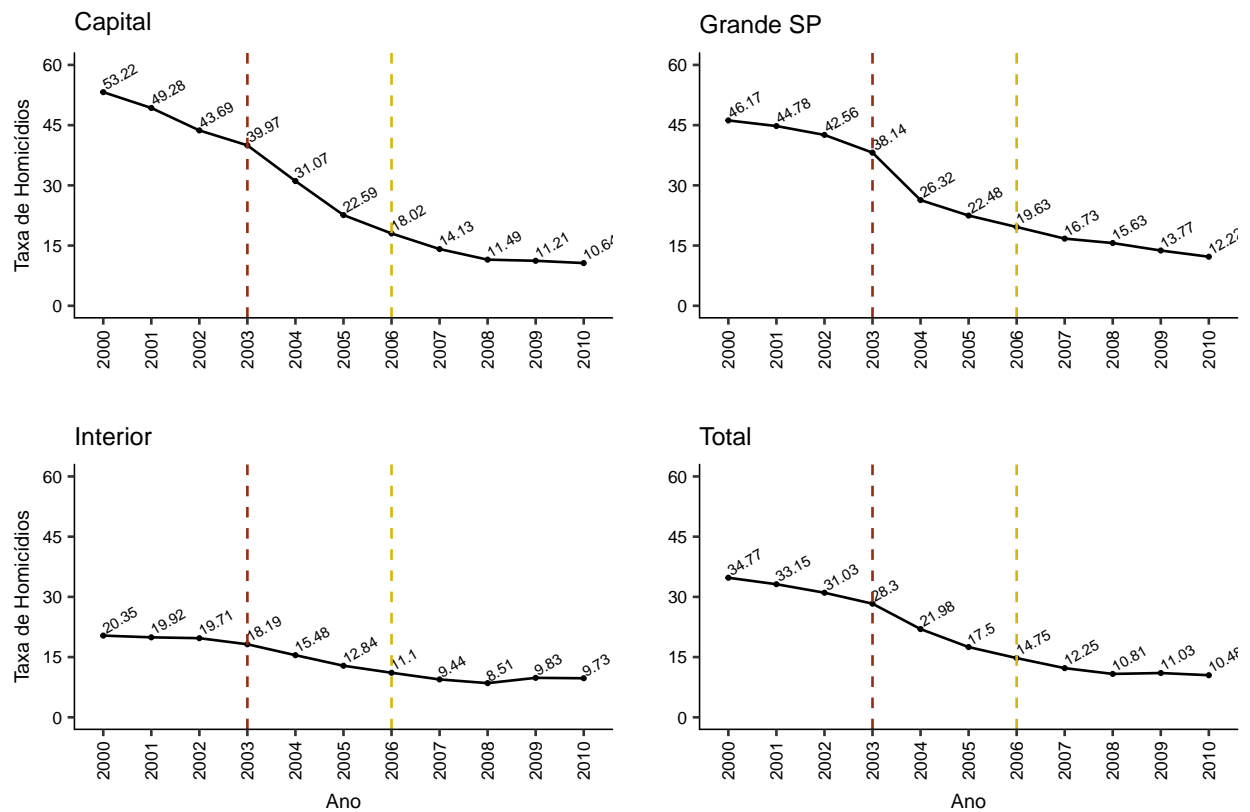
```
quadro_homic_tx <- annotate_figure(quadro_homic_tx,
  top = text_grob(
    "Taxa de homicídios anuais - Estado de São Paulo\n2000 até 2010",
```

```

color = "black",
vjust = .5,
size = 10,
family = "Times", just = "center"),
bottom = NA,
left = NA,
right = NA,
fig.lab = NA,
fig.lab.face = NA)

```

Taxa de homicídios anuais – Estado de São Paulo
2000 até 2010



3 Município de São Paulo

3.1 Dados

Os dados referem-se ao número de ocorrências de homicídio registradas entre os anos de 2003 e 2013. Como a interpretação de ocorrências criminais é sensível à mudanças demográficas, os dados foram normalizados em relação à população residente, sendo calculado, portanto, uma taxa de homicídios por 100.000 habitantes:

$$txhomicidio_{ij} = \left(\frac{homicidio_{ij}}{populacao_{ij}} \right) 100000$$

Na equação acima, a taxa de homicídio no período i é calculada para a localidade j por 100.000 habitantes.

- Os dados de ocorrências criminais são provenientes da Secretaria de Segurança Pública do Estado de São Paulo (SSP/SP)³, que disponibiliza os dados referentes aos registros de Boletins de Ocorrência para diferentes modalidades de delitos em seus 93 Distritos Policiais.
- Os 93 Distritos Policiais são compatibilizados com os 96 Distritos Municipais do município de São Paulo, resultando em 80 regiões de atuação policial.

3.2 carregando arquivo:

```
# lê os dados
msp <- read_rds("C:\\Users\\raul\\Google Drive\\meu_projeto\\dados e scripts\\tabelas_output\\tab_FINAL")

# subset
msp <- msp %>%
  select(ano, distrito, dpol = distrito_num, seccional, homicidio, populacao)

# taxa de homicídio
msp$tx_homicidio <- (msp$homicidio/msp$populacao)*100000
```

³<http://www.ssp.sp.gov.br/Estatistica/Pesquisa.aspx>

3.3 Estatísticas descritivas

3.3.1 Figura: Histograma e boxplot da taxa de homicídios por ano de referência:

- Histograma:

```
msp$ano <- as.character(msp$ano)

histograma <- ggplot(msp, aes(x = tx_homicidio)) +
  geom_histogram(aes(y = ..density.., fill = ano, color = ano),
                 alpha = .3, size=.2, position = "identity", bins = 80) +
  theme(plot.title       = element_text(hjust = .5, size = 10, family = "Times"),
        legend.position = 'bottom',
        legend.title     = element_text(size = 7.5),
        legend.key.size  = unit(.3,"cm"),
        axis.ticks       = element_line(size=.2),
        legend.text      = element_text(size = 7.5),
        axis.text.x      = element_blank(),
        axis.text.y      = element_text(size = 7.5, color = 'black'),
        axis.line.x      = element_line(size = .3, linetype = '1F'),
        axis.line.y      = element_line(size = .2),
        panel.background = element_blank(),
        axis.title.x     = element_blank(),
        axis.title.y     = element_text(size = 7.5)) +
  labs(x      = "",
        y      = "Densidade",
        fill   = "Ano",
        color  = "Ano") +
  geom_rug(aes(color = ano), size = .2) +
  scale_color_manual(values = c(wes_palette("GrandBudapest")[4],
                                wes_palette("Moonrise1")[4])) +
  scale_fill_manual(values = c(wes_palette("Cavalcanti")[1],
                                wes_palette("Moonrise1")[4])) +
  geom_vline(data = filter(msp, ano == "2003"),
             aes(xintercept = mean(tx_homicidio), color = ano), linetype = "dashed") +
  geom_vline(data = filter(msp, ano == "2013"),
             aes(xintercept = mean(tx_homicidio), color = ano), linetype = "dashed") +
  scale_x_continuous(limits = c(0,255),
                     breaks = c(seq(0,250,by=25)))
```

* Boxplot

```
msp$ano <- as.character(msp$ano)

is_outlier <- function(x) {
  return(x < quantile(x, 0.25) - 1.5 * IQR(x) | x > quantile(x, 0.75) + 1.5 * IQR(x))
}

boxplot <- msp %>%
```

```

group_by(ano)
mutate(outlier = ifelse(is_outlier(tx_homicidio),
                        distrito, as.character(NA)))

ggplot(., aes(x = ano, y = tx_homicidio, color = ano) ) +
geom_boxplot(aes(fill = ano), size=.3, alpha = .3, outlier.size = 1, notch = TRUE) +
  scale_color_manual(values = c(wes_palette("GrandBudapest")[4],
                                wes_palette("Moonrise1")[4]) ) +
  scale_fill_manual(values = c(wes_palette("Cavalcanti")[1],
                                wes_palette("Moonrise1")[4]) ) +
  theme(plot.title      = element_text(hjust = .5, size = 7.5, family="Times"),
        axis.text.x     = element_text(size = 6, color = 'black'),
        axis.text.y     = element_text(size = 7.5, color = 'black'),
        axis.title      = element_text(colour = "black", size = 7.5),
        axis.ticks      = element_line(size = .2),
        axis.line.x     = element_line(size = .2),
        axis.line.y     = element_blank(),
        panel.background = element_rect(fill = "white"),
        legend.position = "none") +
  labs(y = "Taxa de homicídio",
       x = "",
       fill = "Ano",
       color = "Ano") +
  geom_text_repel(aes(label = outlier, color = ano), na.rm = TRUE, hjust = -0.3,
                 size = 2, min.segment.length = .3, segment.size = .07) +
  scale_y_continuous(limits = c(0,255),
                    breaks = c(seq(0,250,by=25))) +
  coord_flip()

```

* E a figura:

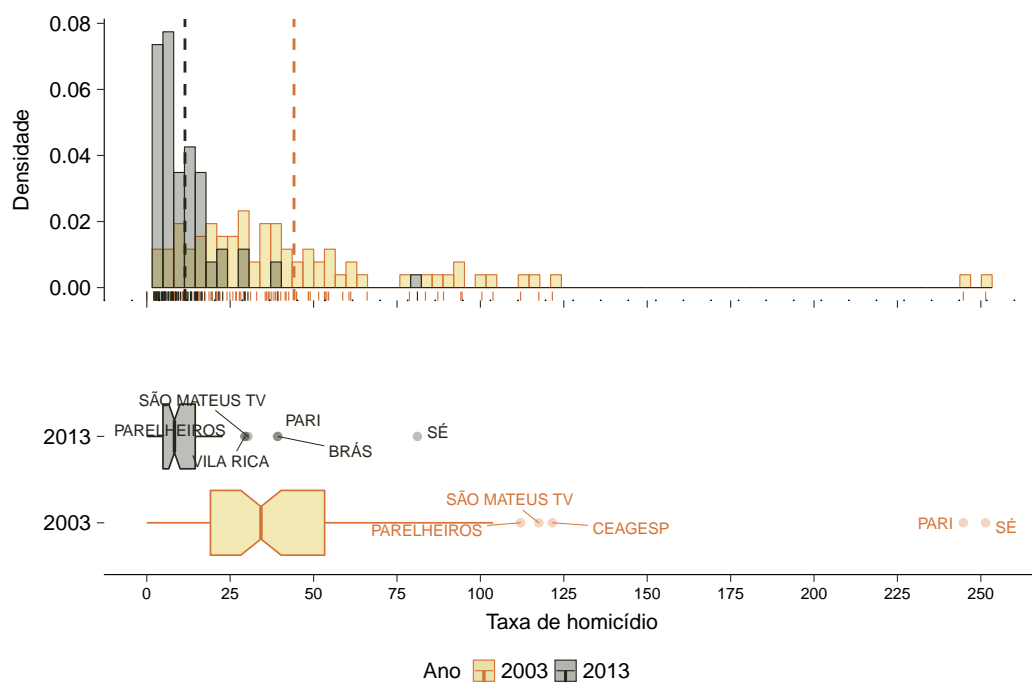
```

quadro_msp_aed <- ggarrange(histograma, boxplot,
                             ncol = 1,
                             nrow = 2,
                             align = "hv",
                             common.legend = TRUE,
                             legend= "bottom",
                             heights = c(1, .75))

quadro_msp_aed <- annotate_figure(quadro_msp_aed,
                                  top = text_grob(
"Taxa de Homicídios (100000 habitantes) - Município de São Paulo\n(2003 e 2013)",
                                  color = "black",
                                  vjust = .5,
                                  size = 10,
                                  family = "Times", just = "center"),
                                  bottom = NA,
                                  left = NA,
                                  right = NA,
                                  fig.lab = NA,
                                  fig.lab.face = NA
                                  )

```

Taxa de Homicídios (100000 habitantes) – Município de São Paulo
(2003 e 2013)



3.3.2 Figura: Taxas de homicídio por seccional (2000 e 2010):

A função abaixo é criada para ordenar as barras dos gráficos da maior para a menor, isso permite uma visualização mais adequada, ressaltando os maiores e menores valores da taxa de homicídios por Seccional:

```
limits <- function(x) { # x: nome da seccional

  msp %>%
  filter(ano == 2003 & seccional == x) %>%
  select(distrito, tx_homicidio) %>%
  arrange(desc(tx_homicidio)) %>%
  select(distrito)
  # Ordena barras em geom_bar()+scale_x_discrete()

} # first function!
```

Uma função para gerar as 8 seccionais:

```
homic_seccional <- function(x, y, z){

  ggplot(data = filter(msp, seccional == x),
          aes(x = distrito, y = tx_homicidio, fill = ano)) +
  scale_x_discrete(limits = as_vector(limits(x))) +
  geom_bar(stat = "identity",
           position = "dodge",
           show.legend = y,
           alpha = .5,
           color = "black") +
  theme(plot.title = element_text(hjust = .5),
        plot.subtitle = element_text(hjust = .5, margin = margin(b = -10)),
        axis.text = element_text(colour = "black"),
        axis.text.x = element_text(size = 10, angle = 90, hjust = 1, vjust = .3),
        axis.text.y = element_text(size = 10),
        axis.ticks = element_line(),
        axis.line = element_line(size = .3, colour = "black"),
        axis.title.x = element_blank(),
        axis.title.y = element_text(size = 10),
        panel.background = element_rect(fill = "white")) +
  labs(fill = "Ano:",
        color = "Ano:",
        subtitle = paste("Seccional: ", x),
        y = z) +
  scale_fill_manual(values = c(wes_palette("GrandBudapest")[4],
                               wes_palette("Moonrise1")[4])) +
  scale_color_manual(msp$ano, values = c(wes_palette("GrandBudapest")[4],
                                           wes_palette("Moonrise1")[4])) +
  scale_y_continuous(limits = c(0,255),
                     breaks = c(seq(0,250,by=25)))

}
```

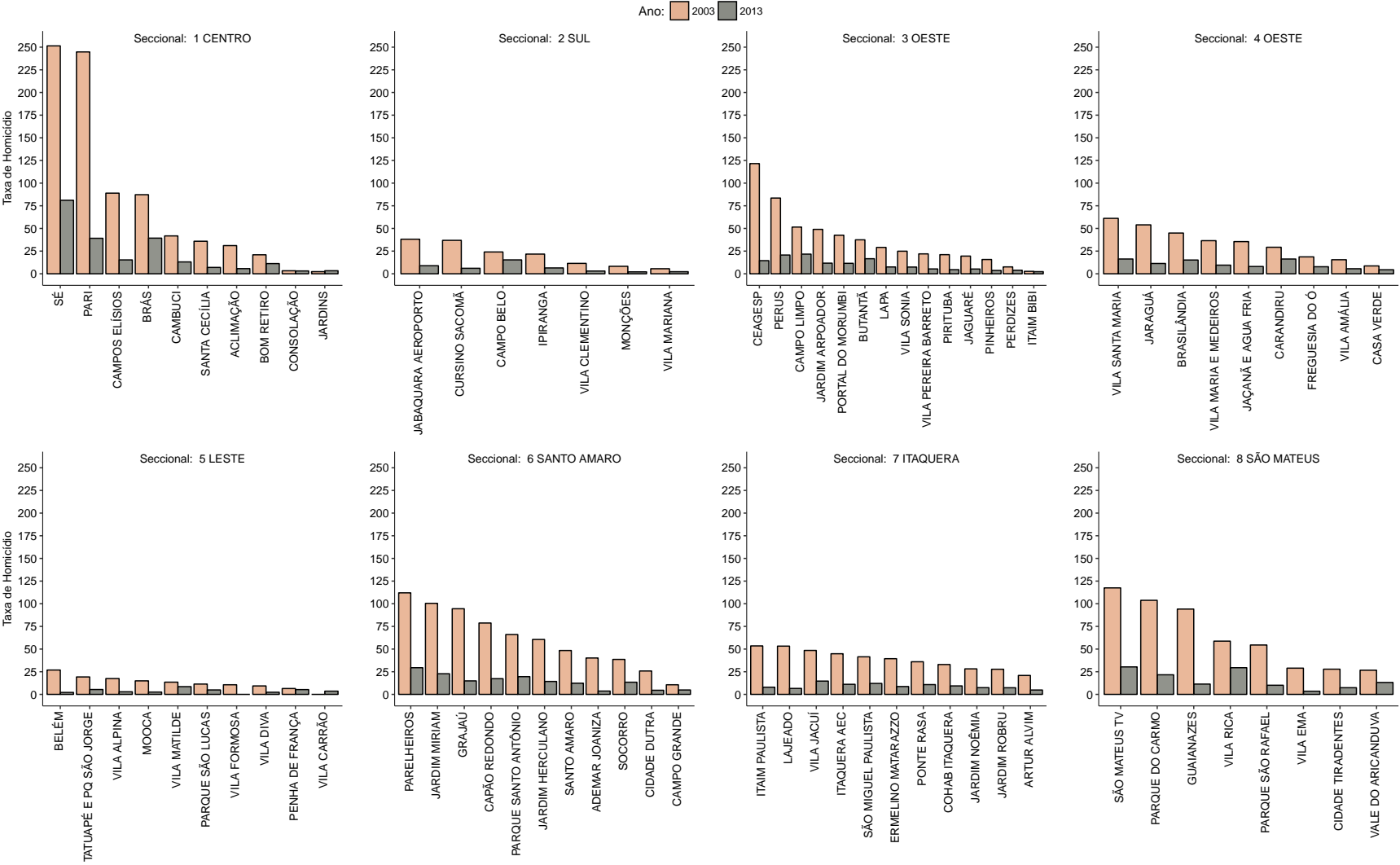
* A figura:

```
quadro_msp_secc <- ggarrange(homic_seccional("1 CENTRO", T, "Taxa de Homicídio"),
  homic_seccional("2 SUL", F, ""),
  homic_seccional("3 OESTE", F, ""),
  homic_seccional("4 OESTE", F, ""),
  homic_seccional("5 LESTE", F, "Taxa de Homicídio"),
  homic_seccional("6 SANTO AMARO", F, ""),
  homic_seccional("7 ITAQUERA", F, ""),
  homic_seccional("8 SÃO MATEUS", F, ""),
  ncol = 4,
  nrow = 2,
  align = "hv",
  common.legend = TRUE,
  legend= "top")
```

Editando a figura

```
quadro_msp_secc <- annotate_figure(quadro_msp_secc,
  top = text_grob(
    "Taxa de Homicídios (100000 habitantes) - Seccionais do Município de São Paulo (2003 e 2013)",
    color = "black",
    vjust = .5,
    size = 16,
    family = "Times", just = "center"),
  bottom = text_grob("Fonte: SSP/SP",
    color = "black",
    face = "italic",
    size = 10),
  left = NA,
  right = NA,
  fig.lab = NA,
  fig.lab.face = NA
)
```

Taxa de Homicídios (100000 habitantes) – Seccionais do Município de São Paulo (2003 e 2013)



Fonte: SSP/SP

3.3.3 Análise Exploratória de Dados Espaciais (AEDE)

Carrega os dados:

```
# subset
msp <- msp %>%
  select(ano, distrito, dpol, seccional, homicidio, populacao) %>%
  mutate(tx_homicidio = (homicidio / populacao) * 100000)

glimpse(msp)

## Observations: 160
## Variables: 7
## $ ano      <chr> "2003", "2013", "2013", "2003", "2013", "2003", "...
## $ distrito <chr> "SÉ", "SÉ", "BOM RETIRO", "BOM RETIRO", "CAMPOS E...
## $ dpol     <int> 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9...
## $ seccional <chr> "1 CENTRO", "1 CENTRO", "1 CENTRO", "1 CENTRO", "...
## $ homicidio <dbl> 53, 20, 4, 6, 9, 45, 4, 4, 4, 20, 5, 13, 5, 18, 1...
## $ populacao <dbl> 21079, 24654, 35567, 28656, 58784, 50595, 128196,...
## $ tx_homicidio <dbl> 251.435078, 81.122739, 11.246380, 20.938023, 15.3...
```

3.3.3.1 Matriz de contiguidade:

A matriz de vizinhança é calculada à partir de uma matriz esparsa criada manualmente. O comando abaixo carrega o arquivo .txt

```
# diretório
queen <- read.table("C:\\Users\\rauld\\Google Drive\\meu_projeto\\dados e scripts\\tabelas_output\\plan.
dim(queen) # 80x80

## [1] 80 80

queen <- as.matrix(queen) # matriz feita à mão
is.matrix(queen)

## [1] TRUE

#
```

As linhas e as colunas em uma matriz de vizinhança são as localidades, nesse caso elas serão referenciadas pelo número do distrito policial:

```
# Nomeando colunas e linhas com o num. do distrito_pol correspondente
distrito_pol <- c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15",
  "16", "17", "21", "22", "23", "24", "25", "27", "28", "29", "31", "33", "34",
  "36", "37", "38", "40", "41", "42", "44", "46", "47", "48", "50", "51", "53",
  "54", "55", "56", "58", "59", "62", "63", "65", "66", "67", "68", "70", "74",
  "75", "77", "78", "81", "87", "89", "91", "92", "93", "96", "98", "99", "100",
  "102", "103", "1857", "2073", "3052", "3264", "3597", "4380", "4572", "4969",
  "85101", "193990", "268395")

colnames(queen) <- distrito_pol
rownames(queen) <- distrito_pol
```

Após criar o objeto, é possível checar sua estrutura com `summary()`. Agora que temos a matriz pronta no R, trabalharemos com objetos `class(listw)`

```
# objeto 'listw', style="W" (padronizada na linha)
w <- mat2listw(queen, row.names = NULL, style="M")
listw <- nb2listw(w$neighbours, glist=NULL, style="W", zero.policy=NULL)
summary(listw)
```

```
## Characteristics of weights list object:
## Neighbour list object:
## Number of regions: 80
## Number of nonzero links: 414
## Percentage nonzero weights: 6.46875
## Average number of links: 5.175
## Link number distribution:
##
##  1  2  3  4  5  6  7  8  9 10
##  1  4  8 11 23 18 10  2  2  1
## 1 least connected region:
## 25 with 1 link
## 1 most connected region:
## 3264 with 10 links
##
## Weights style: W
## Weights constants summary:
##      n  nn S0      S1      S2
## W 80 6400 80 33.77879 329.0915
```

```
#
```

3.3.3.2 Variáveis defasadas espacialmente:

Aqui temos:

- `tx_homicidio_z`: é a taxa de homicídio por 100000 habitantes nos distritos, centrada na média $Z = \frac{(x-\mu)}{\sigma}$,
- `lag_tx_homicidio_z`: é a taxa de homicídio por 100000 habitantes defasada pela matriz de contiguidade de ordem 1,
- `I_moran`: é a estatística *I de Moran* calculada globalmente,
- `local_moran`: é o índice local de Moran,
- `local_moran_pvalor`: é a significância estatística a 5% do indicador de Moran local,
- `quad_sig`: é uma variável categórica criada para identificar os *clusters* calculados pelo *I de Moran* local.

Com a matriz de contiguidade pronta, podemos criar variáveis espacialmente defasadas. Na função abaixo isso é feito de acordo com o ano de referência:

```
defasando <- function(x) { # x: "2003" e "2013"

  msp                                     %>%
  filter(ano == x)                       %>%
  arrange(dpol)                           %>%
  mutate(tx_homicidio_z = (tx_homicidio - mean(tx_homicidio)) / sd(tx_homicidio),
         # lag.listw defasa uma variável qualquer de acordo com W
         lag_tx_homicidio = lag.listw(listw, tx_homicidio),
         lag_tx_homicidio_z = lag.listw(listw, tx_homicidio_z),
         # i de moran Global
         i_moran = rep(moran(tx_homicidio,
                             listw=listw, 80, Szero(listw))[[1]],80)) %>%
  # os resultados do I de Moran local saem em um data.frame à parte:
  bind_cols(., as_data_frame(localmoran(msp %>%
                                         filter(ano == x)
                                         select(tx_homicidio)
                                         as_vector(), listw = listw))) %>%
  select(-E.Ii, -Var.Ii, -Z.Ii) %>%
  rename(local_moran = Ii,
         local_moran_pvalor = `Pr(z > 0)`)
}

# Guarda os dados novamente na tabela
msp <- bind_rows(defasando("2003"), defasando("2013"))

# identify the Local Moran plot quadrant for each observation this is some
# serious slicing and illustrate the power of the bracket

msp$quad_sig <- NA

msp[(msp$tx_homicidio_z >= 0 & msp$lag_tx_homicidio_z >= 0) &
     (msp$local_moran_pvalor <= 0.05), "quad_sig"] <- "Alto-alto"

msp[(msp$tx_homicidio_z <= 0 & msp$lag_tx_homicidio_z <= 0) &
     (msp$local_moran_pvalor <= 0.05), "quad_sig"] <- "Baixo-baixo"

msp[(msp$tx_homicidio_z >= 0 & msp$lag_tx_homicidio_z <= 0) &
     (msp$local_moran_pvalor <= 0.05), "quad_sig"] <- "Alto-baixo"
```

```
msp[(msp$tx_homicidio_z <= 0 & msp$lag_tx_homicidio_z >= 0) &
      (msp$local_moran_pvalor <= 0.05), "quad_sig"] <- "Baixo-alto"

msp[(msp$local_moran_pvalor > 0.05), "quad_sig"] <- "Não sig."

glimpse(msp)
```

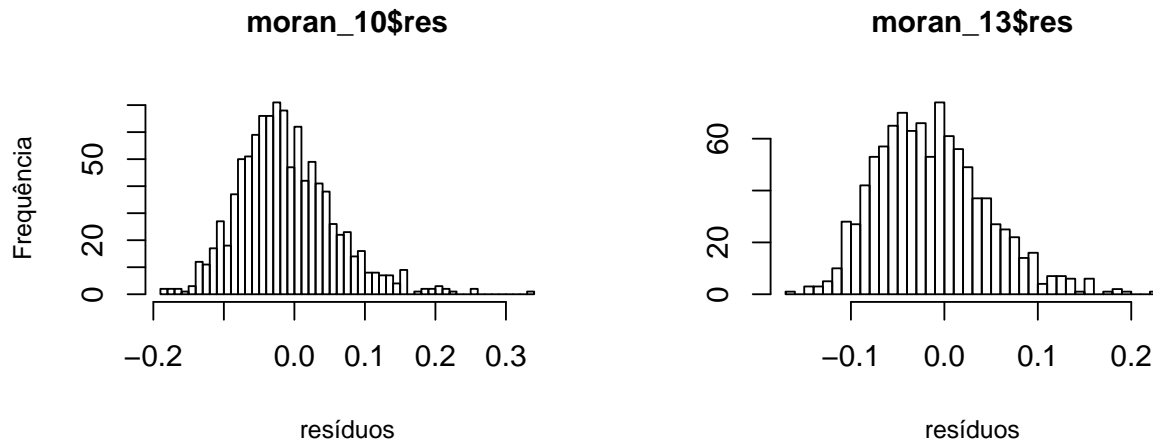
```
## Observations: 160
## Variables: 14
## $ ano          <chr> "2003", "2003", "2003", "2003", "2003", "20...
## $ distrito     <chr> "SÉ", "BOM RETIRO", "CAMPOS ELÍSIOS", "CONS...
## $ dpol         <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, ...
## $ seccional    <chr> "1 CENTRO", "1 CENTRO", "1 CENTRO", "1 CENT...
## $ homicidio     <dbl> 53, 6, 45, 4, 20, 13, 18, 23, 15, 17, 31, 3...
## $ populacao     <dbl> 21079, 28656, 50595, 120821, 64310, 31114, ...
## $ tx_homicidio  <dbl> 251.435078, 20.938023, 88.941595, 3.310683,...
## $ tx_homicidio_z <dbl> 4.73573143, -0.52990521, 1.02361616, -0.932...
## $ lag_tx_homicidio <dbl> 74.01164, 106.21720, 111.26871, 61.86524, 6...
## $ lag_tx_homicidio_z <dbl> 0.68254580, 1.41827228, 1.53367253, 0.40506...
## $ i_moran       <dbl> 0.194438, 0.194438, 0.194438, 0.194438, 0.1...
## $ local_moran    <dbl> 3.27326949, -0.76106316, 1.58976404, -0.382...
## $ local_moran_pvalor <dbl> 1.080103e-22, 9.793767e-01, 3.635768e-05, 8...
## $ quad_sig      <chr> "Alto-alto", "Não sig.", "Alto-alto", "Não ...
```

```
#
```

3.3.3.3 Análise *I* de Moran Global

A função `spdep::moran.mc()` calcula a estatística *I* de Moran e testa sua aleatoriedade com simulações de Monte-Carlo. Veja os resíduos obtidos por 999 simulações:

```
par(mfrow = c(1, 2))
hist(moran.mc(msp$tx_homicidio[msp$ano == "2003"], listw=listw, nsim=999)[[7]],
      breaks = 50,
      main = list("moran_10$res", cex = 1),
      xlab = list("resíduos", cex = .8),
      ylab = list("Frequência", cex=.8))
hist(moran.mc(msp$tx_homicidio[msp$ano == "2013"], listw=listw, nsim=999)[[7]],
      breaks = 50,
      main = list("moran_13$res", cex = 1),
      xlab = list("resíduos", cex = .8),
      ylab = list("", cex = .8))
```



O diagrama de dispersão de Moran pode ser visualizado rapidamente com a função `spdep::moran.plot()`

```
par(mfrow = c(1, 2))
moran.plot(as.vector(msp$tx_homicidio_z[msp$ano == "2003"]),
           listw,
           zero.policy = T,
           spChk = NULL,
           xlab = list("Taxa de homicídios", cex = .8),
           ylab = list("Taxa de homicídios defasada", cex = .8),
           labels = as.character(msp$distrito[msp$ano == "2003"]),
           quiet = NULL)
title(main = list("2003", cex = .8))

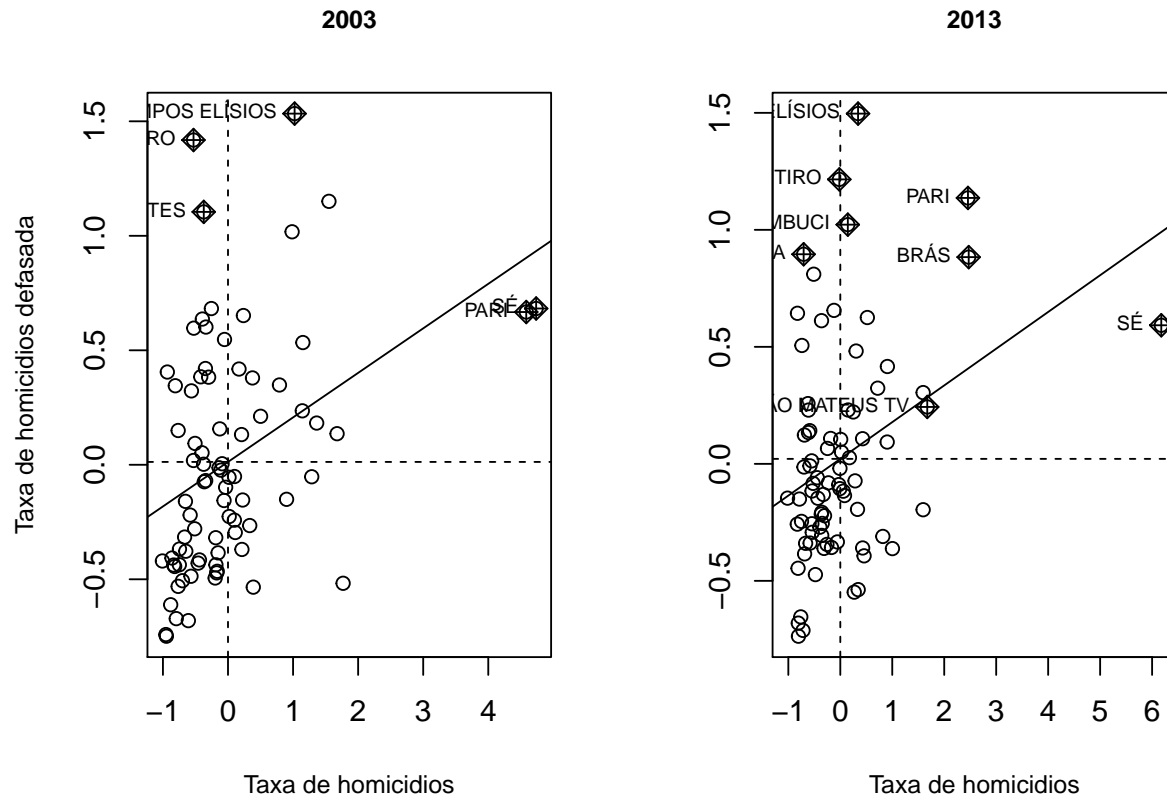
moran.plot(as.vector(msp$tx_homicidio_z[msp$ano == "2013"]),
           listw,
           zero.policy = T,
           spChk = NULL,
           xlab = list("Taxa de homicídios", cex = .8),
           ylab = list("", cex = .8))
```



```

labels = paste(msp$distrito[msp$ano == "2013"]),
quiet = NULL)
title(main=list("2013", cex = .8))

```



Também podemos tabular os resultados regredindo a taxa de homicídios contra seus valores defasados espacialmente:

```

moran_i <- function(x){
  lm(lag_tx_homicidio_z~tx_homicidio_z, data=msp, subset=(ano==x))
}

tabela_moran <- huxreg("2003"=moran_i("2003"),
                      "2013"=moran_i("2013"),
                      coefs = "tx_homicidio_z",
                      statistics = "r.squared") %>%
  set_number_format(1, c(2,3), NA) %>%
  set_bold(1, everywhere, TRUE) %>% # negrito
  set_top_border(1, everywhere, 1) %>% # borda superior
  set_right_padding(3) %>% # para posicionar
  set_left_padding(3) %>% # para posicionar
  set_width(.6) %>% # para posicionar no pdf
  set_position('center') %>% # para posicionar no pdf
  set_align(everywhere,c(2,3), 'center') %>% # alinhamento do texto na célula

```

```

set_align(everywhere,1,'left')           %>% #
set_escape_contents(4, 1, FALSE)         %>% # para aparecer potenciação no pdf
set_font_size(3,everywhere, 8)           %>% # italico
set_italic(3,everywhere, TRUE)            %>%
set_caption(
'Tabela: Estatística I de Moran (2003 e 3013)'
)

tabela_moran[2,1] <- "Taxa de homicídios"
tabela_moran[4,1] <- "$R^2$"

```

Table 2: Tabela: Estatística I de Moran (2003 e 3013)

	2003	2013
Taxa de homicídios	0.194 *** (0.053)	0.157 ** (0.049)
R^2	0.148	0.117

*** p < 0.001; ** p < 0.01; * p < 0.05.

3.3.3.4 Figura: Diagrama de dispersão de *Moran* - autocorrelação espacial

O gráfico será gerado com os seguintes parâmetros:

```

moran_ggplot <- function(x){

  ggplot(filter(msp, ano==x),
    aes(x = tx_homicidio_z,
      y = lag_tx_homicidio_z, color = as.factor(quad_sig))) +
  geom_point(shape= 21,
    fill = "white",
    size = 1.2,
    stroke = .6) +
  theme_bw(base_size = 8) +
  theme(plot.title      = element_text(hjust = .5),
    plot.subtitle      = element_text(hjust = .5, margin = margin(b = -10)),
    axis.text          = element_text(colour = "black"),
    axis.text.x        = element_text(size = 6.5),
    axis.text.y        = element_text(size = 6.5),
    axis.ticks         = element_line(size=.3),
    axis.line          = element_blank(),
    axis.title.x       = element_blank(),
    axis.title.y       = element_text(size = 10),
    panel.background   = element_rect(size = .3),
    panel.grid         = element_blank()) +
  labs(title = paste(x),
    x      = "Taxa de homicídios",
    y      = "",
    color = "I de Moran Local (p-valor<0,05)") +
  scale_y_continuous(limits = c(-2,2), breaks = seq(-2,2, by = .5)) +
  scale_x_continuous(limits = c(-8,8), breaks = seq(-8,8, by = 2)) +

```

```

scale_color_manual(values=c('Alto-alto' = wes_palette("Royal1")[2],
                             'Baixo-baixo' = wes_palette("Darjeeling2")[2],
                             'Não sig.' = 'black')) +
geom_vline(xintercept = 0, size = .3) +
geom_hline(yintercept = 0, size = .3) +
geom_abline(slope = ifelse(x == "2003", 0.194, 0.157),
            intercept = ifelse(x == "2003", 0.012, 0.022), size = .5, linetype = 'dashed') +
geom_text_repel(data = subset(msp, ano == x & quad_sig == "Alto-alto" | ano == x & quad_sig == "Ba
                aes(label = distrito),
                size = 2)+
geom_label( label = "Alto-alto", x = 7, y = 2, size = 1.5, colour = "black") +
geom_label( label = "Alto-baixo", x = 7, y = -2, size = 1.5, colour = "black") +
geom_label( label = "Baixo-baixo", x = -6.8, y = -2, size = 1.5, colour = "black") +
geom_label( label = "Baixo-alto", x = -7, y = 2, size = 1.5, colour = "black")
}

```

O quadro final do Diagrama de Dispersão de Moran:

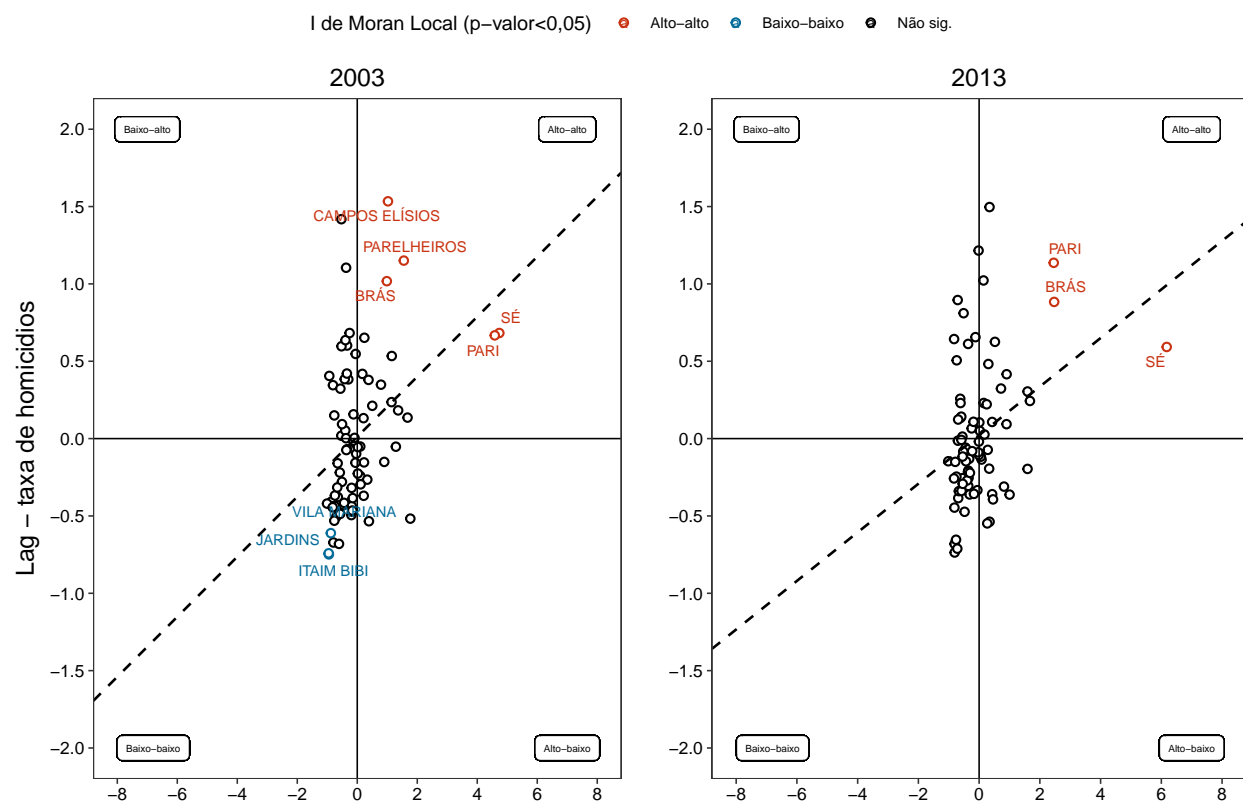
```

# enquadrando gráficos
figura_moran <- ggarrange(moran_ggplot("2003") + ylab("Lag - taxa de homicídios") +
                        xlab("Taxa de homicídio"), moran_ggplot("2013") +
                        xlab("Taxa de homicídio"),
                        ncol=2, nrow=1, # align="hv",
                        common.legend = TRUE, legend = "top")

figura_moran <- annotate_figure(figura_moran,
                                top = text_grob("Figura: Diagramas de dispersão de Moran (2003/2013)",
                                                color = "black",
                                                vjust = .5,
                                                size = 10,
                                                family = "Times"),
                                bottom = NA,
                                left = NA,
                                right = NA,
                                fig.lab = NA, fig.lab.face = NA
                                )

```

Figura: Diagramas de dispersão de Moran (2003/2013)
Índice global e local



4 Análise bivariada

```
msp <- read_rds("C:\\Users\\raul\\Google Drive\\meu_projeto\\dados e scripts\\tabelas_output\\tab_FINAL")

msp <- msp %>%
  select(ano, distrito, pop_jovem_onu, homicidio, populacao, area_constr_resid_baixop, renda_domic_media,
         filter(ano == "2003") %>%
         mutate(tx_homicidio = (homicidio/populacao)*100000,
                eformais = (emprego_formal/populacao)*100000,
                tx_belica = (armas_apreendidas/populacao)*100000) %>%
         rename(jovens1524 = pop_jovem_onu)

summary(lm(tx_homicidio ~ jovens1524 + area_constr_resid_baixop + renda_domic_media + renda_domic_dp +
           favela + eformais + tx_belica, data = msp))

##
## Call:
## lm(formula = tx_homicidio ~ jovens1524 + area_constr_resid_baixop +
##     renda_domic_media + renda_domic_dp + favela + eformais +
##     tx_belica, data = msp)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -59.712  -9.297   0.200  10.516  46.901
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.014e+01  3.359e+01   0.897  0.37254
## jovens1524        6.535e-05  3.976e-04   0.164  0.86991
## area_constr_resid_baixop -1.275e-05  1.040e-05  -1.226  0.22415
## renda_domic_media    -6.457e+00  8.211e-01  -7.864 2.75e-11 ***
## renda_domic_dp       4.734e+00  3.124e+00   1.516  0.13401
## favela             1.288e-03  8.222e-04   1.566  0.12162
## eformais           1.792e-04  6.032e-05   2.971  0.00404 **
## tx_belica          2.529e-01  2.734e-02   9.251 7.13e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.1 on 72 degrees of freedom
## Multiple R-squared:  0.7882, Adjusted R-squared:  0.7676
## F-statistic: 38.27 on 7 and 72 DF,  p-value: < 2.2e-16
```

