

Mapa eleitoral - Pref. SP. 2024 - 1º turno

Caio Durazzo

2024-10-16

Parâmetros gerais

```
rm(list = ls()) # limpar o ambiente antes de começar  
  
set.seed(123) # para reprodutibilidade exata do código
```

Biblioteca

```
library(sf) # manipulação de dados espaciais  
library(dplyr) # manipulação de dados  
library(ggplot2) # visualização de dados  
library(readr) # leitura de arquivos .csv  
library(patchwork) # combinação múltiplos gráficos
```

Importação dos dados

```
# Criando objetos para armazenar o caminho dos arquivos  
  
caminho_votacao <- # votação nominal por município e zona eleitoral  
  "Planilha/votacao_candidato_munzona_2024_SP.csv"  
  
caminho_zonas <- # zonas eleitorais de SP município georreferenciadas  
  "Planilha/zonas_sp/zonas_sp.shp"  
  
# Lendo os dados e salvando nos objetos  
  
votacao <-  
  read_csv2(caminho_votacao,  
            locale = locale(encoding = "Latin1"))  
  
zonas_sp <-  
  st_read(caminho_zonas)
```

Tratamento dos dados

```
votacao_pref_sp <- # filtrando os dados de Prefeito para São Paulo  
  votacao %>%  
  filter(CD_MUNICIPIO == 71072 & DS_CARGO == "Prefeito")  
  
zonas_sp$zona <- # limpando a coluna 'zona' para conter apenas o número
```

```

gsub("^((\\d+).*)",
      "\\1",
      zonas_sp$zona)

zonas_sp$zona <- # convertendo para variável numérica
  as.numeric(zonas_sp$zona)

dados <- # juntando os dados de votação com o shapefile
  zonas_sp %>%
  left_join(votacao_pref_sp,
            by = c("zona" = "NR_ZONA"))

dados <- # coluna de percentual de votos por zona por candidato
  dados %>%
  group_by(zona) %>%
  mutate(total_votos_zona = sum(QT_VOTOS_NOMINAIS, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(percentual_votos = (QT_VOTOS_NOMINAIS / total_votos_zona) * 100)

dados <- # coluna com os nomes formatados corretamente
  dados %>%
  mutate(NM_FORMATADO = case_when(
    NM_URNA_CANDIDATO == "RICARDO NUNES" ~ "Ricardo Nunes (MDB)",
    NM_URNA_CANDIDATO == "GUILHERME BOULOS" ~ "Guilherme Boulos (PSOL)",
    NM_URNA_CANDIDATO == "PABLO MARÇAL" ~ "Pablo Marçal (PRTB)",
    NM_URNA_CANDIDATO == "TABATA AMARAL" ~ "Tabata Amaral (PSB)",
    TRUE ~ NM_URNA_CANDIDATO))

dados_vencedores <- # coluna com os nomes dos vencedores
  dados %>%
  group_by(zona) %>%
  filter(QT_VOTOS_NOMINAIS == max(QT_VOTOS_NOMINAIS)) %>%
  ungroup() %>%
  filter(NM_FORMATADO %in% c("Ricardo Nunes (MDB)",
                           "Guilherme Boulos (PSOL)",
                           "Pablo Marçal (PRTB)"))

```

Visualização dos dados

```

# Mapa geral destacando as zonas eleitorais vencidas

mapa_vencedores <-
  ggplot() +
  geom_sf(data = zonas_sp,
          fill = NA,
          color = "black",
          size = 0.5) +
  geom_sf(data = dados_vencedores,
          aes(fill = NM_FORMATADO),
          color = NA) +
  scale_fill_manual(values = c("Ricardo Nunes (MDB)" = "green",
                              "Guilherme Boulos (PSOL)" = "yellow",
                              "Pablo Marçal (PRTB)" = "blue"),

```

```

        name = "Candidato") +
theme_void() +
theme(legend.position = c(1.1, 0.3),
      legend.justification = c(1, 0),
      plot.title = element_text(hjust = 0.5))

# Função do mapa de calor de cada candidato

gerar_mapa_calor <-
  function(candidato_formatado,
           cor) {

    dados_candidato <-
      dados %>%
        filter(NM_FORMATADO == candidato_formatado)

    if (nrow(dados_candidato) == 0) {
      stop(paste("Nenhum dado encontrado para", candidato_formatado))
    }

    ggplot() +
      geom_sf(data = zonas_sp,
              fill = NA,
              color = "black",
              size = 0.5) +
      geom_sf(data = dados_candidato,
              aes(fill = percentual_votos),
              color = NA) +
      scale_fill_gradient(low = "white",
                          high = cor,
                          name = "Percentual de votos (%)",
                          na.value = "grey") +
      labs(title = candidato_formatado) +
      theme_void() +
      theme(legend.position = "right",
            plot.title = element_text(hjust = 0.5))

  }

mapa_nunes <- # gerando mapas de calor para cada candidato
  gerar_mapa_calor("Ricardo Nunes (MDB)", "green") +
  theme(plot.margin = margin(5, 5, 5, 5))

mapa_boulos <-
  gerar_mapa_calor("Guilherme Boulos (PSOL)", "yellow") +
  theme(plot.margin = margin(5, 5, 5, 5))

mapa_marçal <- gerar_mapa_calor("Pablo Marçal (PRTB)", "blue") +
  theme(plot.margin = margin(5, 5, 5, 5))

mapa_tabata <- gerar_mapa_calor("Tabata Amaral (PSB)", "red") +
  theme(plot.margin = margin(5, 5, 5, 5))

```

```

mapas_calor_com_titulo <- # combinando mapas de calor
  (mapa_nunes + mapa_boulos) / (mapa_marçal + mapa_tabata)

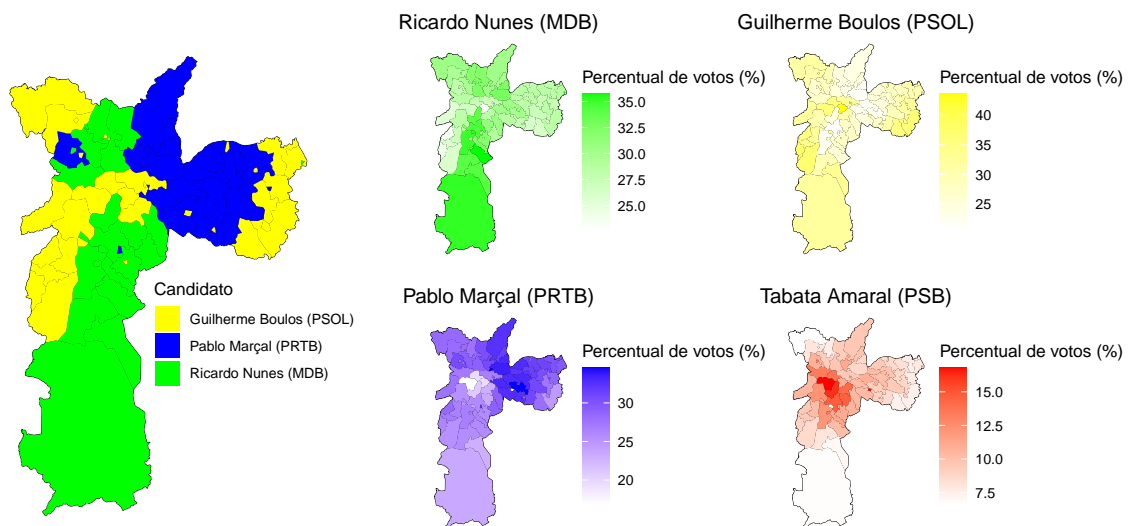
mapa_combinado <- # combinando o mapa de vencedores com os mapas de calor
  mapa_vencedores +
  mapas_calor_com_titulo +
  plot_layout(widths = c(1, 1)) +
  plot_annotation(
    title = "Eleição para a prefeitura de São Paulo de 2024 - 1º turno por zonas eleitorais",
    caption = "Fonte: Elaboração própria a partir de dados do TSE",
    theme = theme(
      plot.title = element_text(size = 18, face = "bold", hjust = 0.5),
      plot.caption = element_text(size = 10, hjust = 1, vjust = 1)
    )
  )

# Exibindo a figura final

print(mapa_combinado)

```

Eleição para a prefeitura de São Paulo de 2024 – 1º turno por zonas eleitorais



Fonte: Elaboração própria a partir de dados do TSE

Exportação dos resultados

```

# Salvando o mapa .png

png("Gráficos/Mapa eleitoral - Pref. SP 1º turno - 2024.png",
  width = 30,
  height = 15,
  units = "cm",
  res = 200)
print(mapa_combinado)
dev.off()

```