



UFRJ

UNIVERSIDADE FEDERAL
DO RIO DE JANEIRO

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

**TRABALHO 2 | TÓP. ESP. EM ENG. DE DADOS
USO DE R E TIDYVERSE
RELATÓRIO DE ANÁLISE DE DADOS: PERFIL DO ELEITORADO E DADOS
METEOROLÓGICOS**

**DHAYSE DE LIMA TITO DRE: 120019062
GIOVANNI ANSELMO FERNANDES COELHO DE ORNELLAS DRE: 120136123
GUILHERME GUEDES GUIMARÃES MERÇON DRE: 117098748
EDUARDO DAEMON TEIXEIRA DOS SANTOS DRE: 120154798**

Rio de Janeiro-RJ

2024

SUMÁRIO

1	ANÁLISE DO PERFIL DO ELEITORADO	3
1.1	INTRODUÇÃO	3
1.2	DESCRIÇÃO DO DATASET	3
1.2.1	Origem dos Dados	3
1.3	CARGA	3
1.3.1	Leitura dos Dados	3
1.3.2	Dimensões do Dataset	3
1.4	TRANSFORMAÇÕES	4
1.4.1	Seleção de Colunas	4
1.4.2	Comparação de Dimensões	4
1.5	ANÁLISES	5
1.5.1	Seleção de agrupamentos específicos	5
1.5.2	Agrupamentos e Estatísticas	6
1.5.3	Visualização Gráfica	7
1.5.3.1	Distribuição Percentual por Estado Civil	8
1.5.3.2	Pirâmide Etária por Gênero	9
1.5.3.3	Top 20 Municípios por Total de Eleitores	10
1.5.3.4	Top 20 Municípios por Total de Eleitores	10
1.5.3.5	Distribuição de Eleitores por Grau de Escolaridade	11
1.5.3.6	Distribuição de Eleitores por Grau de Escolaridade	11
1.5.3.7	Distribuição de Eleitores por Estado Civil	12
1.5.3.8	Distribuição de Eleitores por Estado Civil	12
1.5.3.9	Distribuição de Eleitores por Faixa Etária	13
1.5.3.10	Distribuição de Eleitores por Faixa Etária	13
1.5.3.11	Distribuição de Eleitores por Gênero	14
1.5.3.12	Distribuição de Eleitores por Gênero	14
1.5.4	Mapas de Calor	14
1.5.4.1	Mapa de Calor: Faixa Etária vs Estado Civil (Homens e Mulheres)	15
1.6	ANÁLISE FINAL	17
1.6.1	Análise Comparativa entre os Estados da Bahia e Rio de Janeiro	17
1.6.1.1	Municípios com Mais e Menos Eleitores	17
1.6.1.2	Faixa de Variação da Quantidade de Eleitores por Município e Gênero	17
1.6.1.3	Características que Melhor Descrevem a Maioria dos Eleitores	18
1.6.1.4	Conclusão Geral	18
2	ANÁLISE DE DADOS METEOROLÓGICOS	20
2.1	INTRODUÇÃO	20
2.2	DESCRIÇÃO DO DATASET	20

2.2.1	Origem dos Dados	20
2.2.2	Carga	20
2.2.2.1	Importação dos Dados	20
2.2.2.2	Contagem de linhas, colunas e células	21
2.2.3	Variáveis Seleccionadas	21
2.3	TRANSFORMAÇÕES NOS DADOS	22
2.3.1	Eliminação de Colunas	22
2.3.2	Eliminação de Linhas Inválidas	22
2.4	ANÁLISES REALIZADAS	23
2.4.1	Análise Trimestral da Temperatura do Ar	23
2.4.2	Cálculos das Estatísticas Básicas	24
2.5	ANÁLISE COMPARATIVA ENTRE 2022 E 2023	26
2.5.1	Precipitação Total	26
2.5.1.1	Tendência Geral ao Longo do Ano	27
2.5.1.2	Comparação Mensal e Horária	27
2.5.1.3	Padrão Horário de 2023	28
2.5.1.4	Pontos de Destaque	28
2.5.2	Pressão Atmosférica	29
3	RESULTADO DAS ANÁLISE	32
3.1	LINK PARA O REPOSITÓRIO	32

1 ANÁLISE DO PERFIL DO ELEITORADO

1.1 INTRODUÇÃO

Nesta análise, investigaremos o perfil do eleitorado para os estados da Bahia e do Rio de Janeiro. Usaremos ferramentas do *Tidyverse* em *R* para realizar transformações, seleções e análises de um *dataset* contendo dados eleitorais.

1.2 DESCRIÇÃO DO DATASET

1.2.1 Origem dos Dados

Os dados foram obtidos no site de dados abertos do Tribunal Superior Eleitoral - Assessoria de Gestão Eleitoral (TSE/AGEL: https://dadosabertos.tse.jus.br/pt_PT/dataset/eleitorado-atual), contendo informações sobre os eleitores de todo o Brasil, se encontrando, por exemplo, separados por Município e por local de votação (todos UFs). Aqui tomamos apenas os datasets avulsos do Rio de Janeiro e de São Paulo.

1.3 CARGA

1.3.1 Leitura dos Dados

O código abaixo realiza a leitura do *dataset* no formato CSV, garantindo que separadores, delimitadores e codificação sejam configurados corretamente.

```
library(tidyverse)

# Caminho do dataset
caminho_datasetBA <- "perfil_eleitor_secao_ATUAL_BA.csv"

# Carregar o dataset
dataset_BA <- read_delim(
  file = caminho_datasetBA,
  delim = ";",
  quote = "\"",
  locale = locale(encoding = "ISO-8859-1")
)
```

Listing 1.1 – Código para leitura do dataset

1.3.2 Dimensões do Dataset

As dimensões iniciais do dataset foram:

- Linhas: 5.991.511
- Colunas: 30
- Células: 179.745.330

Esses valores foram obtidos pelo código a seguir.

```
linhas <- nrow(dados)
colunas <- ncol(dados)
células <- linhas * colunas

cat("O dataset contém", linhas, "linhas,", colunas,
    "colunas e um total de", células, "células com dados.\n")
```

Listing 1.2 – Dimensões do dataset

1.4 TRANSFORMAÇÕES

1.4.1 Seleção de Colunas

Selecionaremos apenas as colunas relevantes para análise:

```
library(dplyr)

# Colunas de interesse
dados_selecionados <- dados %>% select(
  NM_MUNICIPIO, CD_GENERO, DS_GENERO,
  CD_ESTADO_CIVIL, DS_ESTADO_CIVIL,
  CD_FAIXA_ETARIA, DS_FAIXA_ETARIA,
  CD_GRAU_ESCOLARIDADE, DS_GRAU_ESCOLARIDADE,
  QT_ELEITORES_PERFIL
)

# Comparação do tamanho antes e depois
dim_antes <- dim(dados)
dim_depois <- dim(dados_selecionados)
```

Listing 1.3 – Seleção de colunas

1.4.2 Comparação de Dimensões

Após a seleção das colunas de interesse e, finalizando a limpeza dos dados, excluimos as linhas com "inválidos" e "NÃO INFORMADOS". Obtendo as seguintes dimensões:

- Linhas: 5.990.152
- Colunas: 10
- Células: 59.901.520

Assim, apresentou uma redução de cerca 66% em relação ao número de células do dataset original.

O código utilizado para limpeza encontra-se a seguir.

```
# Remover linhas com dados inválidos
dataset_final_BA <- dataset_reduzido_BA %>%
  filter(
    DS_FAIXA_ETARIA != "Inválido",
    !if_any(everything(), ~ . == "NÃO INFORMADO")
  )
```

1.5 ANÁLISES

Após limpeza do dataset obtivemos um total de eleitores de 11.262.188 e 12.958.188, na Bahia e Rio de Janeiro, respectivamente.

Código para verificar a quantidade total de eleitores:

```
# Total geral de eleitores
total_geral <- sum(dataset_final_BA$QT_ELEITORES_PERFIL, na.rm = TRUE)
```

1.5.1 Seleção de agrupamentos específicos

Foram feito 5 agrupamentos, sendo eles:

- Homens, 35-39 anos, casados, ensino médio completo
- Mulheres, 45-49 anos, solteiras, ensino superior completo
- Homens, 25-29 anos, divorciados, ensino fundamental incompleto
- Ambos os gêneros, 60-64 anos, viúvos, lê e escreve
- Ambos os gêneros, 21-24 anos, analfabeto

Abaixo, segue uma parte do código de como foi feito esse agrupamento.

```
# Critérios específicos
criterios <- list(
  "Homens, 35-39 anos, casados, ensino médio completo" = dataset_final_BA %>%
    filter(
      DS_GENERO == "MASCULINO",
      DS_FAIXA_ETARIA == "35 a 39 anos",

```

```
    DS_ESTADO_CIVIL == "CASADO",
    DS_GRAU_ESCOLARIDADE == "ENSINO MÉDIO COMPLETO"
  ),
  ...
"Ambos os gêneros, 21-24 anos, analfabeto" = dataset_final_BA %>%
  filter(
    DS_FAIXA_ETARIA == "21 a 24 anos",
    DS_GRAU_ESCOLARIDADE == "ANALFABETO"
  )
)

# Calcular os totais e porcentagens para cada critério
resultados_criterios <- map_dfr(
  criterios,
  ~ data.frame(
    Total = sum(.x$QT_ELEITORES_PERFIL, na.rm = TRUE),
    Porcentagem = sum(.x$QT_ELEITORES_PERFIL, na.rm = TRUE) / total_geral * 100
  ),
  .id = "Critério"
)
```

Listing 1.4 – Filtragem de dados

1.5.2 Agrupamentos e Estatísticas

Foi calculado o total de eleitores para cada variável categórica, informadas abaixo:

- Por Gênero: DS_GENERO
- Por Estado Civil: DS_ESTADO_CIVIL
- Por Faixa Etária: DS_FAIXA_ETARIA
- Por Grau de Escolaridade: DS_GRAU_ESCOLARidade
- Por Município: NM_MUNICIPIO

Além disso, também foram feitos agrupamentos:

- Por Gênero e Idade
- Por Gênero e Município
- Por Escolaridade e Estado Civil
- Por Grau de Escolaridade: DS_GRAU_ESCOLARidade
- Por Município: NM_MUNICIPIO

A seguir um trecho de como é calculado o total de eleitores para as variáveis categóricas.

```
# Contagem de eleitores por gênero
contagem_por_sexo <- dataset_final_BA %>%
  group_by(DS_GENERO) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))
...
# Contagem de eleitores por genero e idade
contagem_sexo_idade <- dataset_final_BA %>%
  group_by(DS_GENERO, DS_FAIXA_ETARIA) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))
```

Listing 1.5 – Agrupamento e Estatísticas

1.5.3 Visualização Gráfica

Para gerar os gráficos utilizamos principalmente o Gráfico de barras.

```
ggplot(data = contagem_por_sexo, aes(x = DS_GENERO, y = Total_Eleitores, fill =
  DS_GENERO)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Distribuição de Eleitores por Gênero",
    x = "Gênero",
    y = "Total de Eleitores"
  ) +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")
```

Listing 1.6 – Gráficos com ggplot2

1.5.3.1 Distribuição Percentual por Estado Civil



Figura 1 – Distribuição percentual dos eleitores por estado civil no estado da Bahia. O gráfico demonstra as proporções de eleitores solteiros, casados, divorciados, viúvos e separados judicialmente.



Figura 2 – Distribuição percentual dos eleitores por estado civil no estado do Rio de Janeiro. O gráfico demonstra as proporções de eleitores solteiros, casados, divorciados, viúvos e separados judicialmente.

1.5.3.2 Pirâmide Etária por Gênero

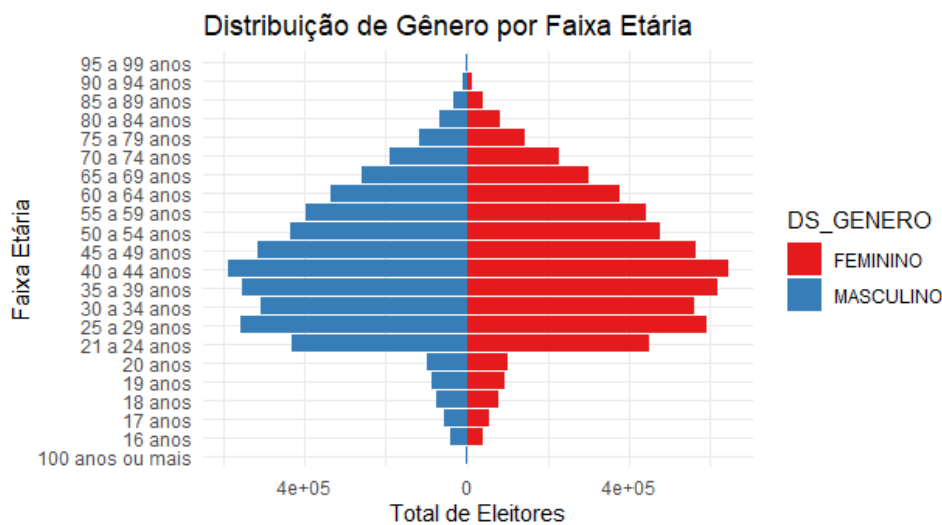


Figura 3 – Pirâmide etária mostrando a distribuição de eleitores por faixa etária e gênero no estado da Bahia. Eleitores masculinos estão representados à esquerda (negativo) e femininos à direita (positivo).

subsubsectionPirâmide Etária por Gênero

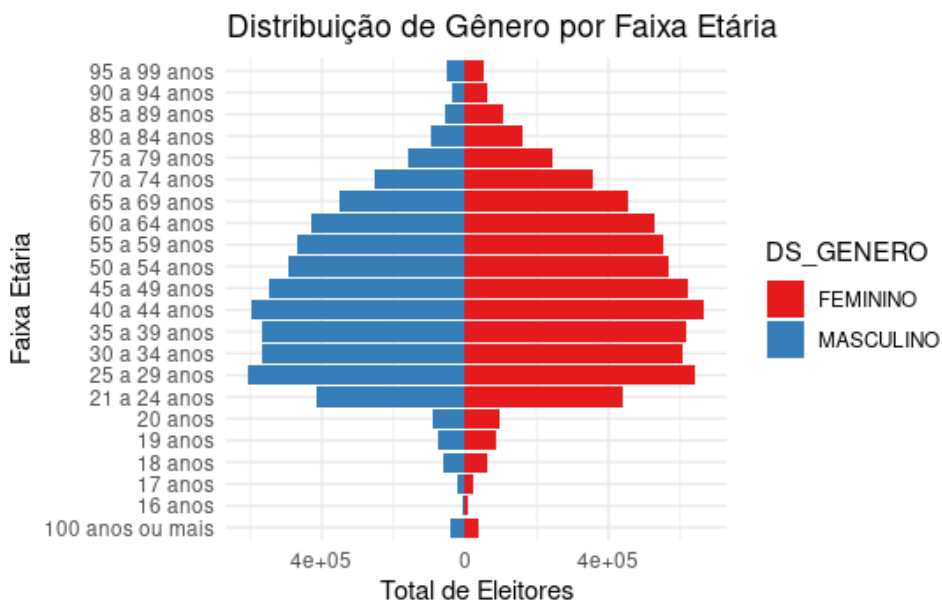


Figura 4 – Pirâmide etária mostrando a distribuição de eleitores por faixa etária e gênero no estado do Rio de Janeiro. Eleitores masculinos estão representados à esquerda (negativo) e femininos à direita (positivo).

1.5.3.3 Top 20 Municípios por Total de Eleitores

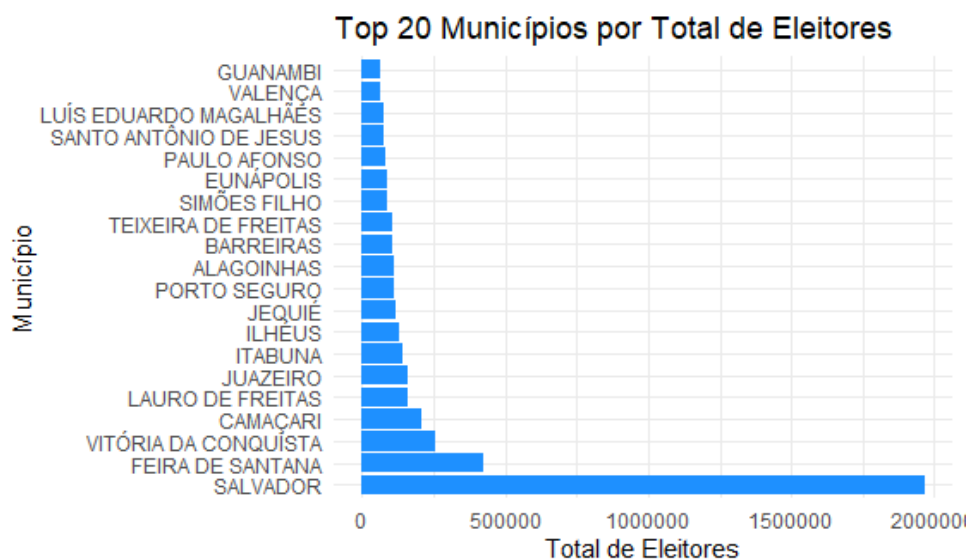


Figura 5 – Os 20 municípios com maior número de eleitores no estado da Bahia. Salvador, a capital, lidera com ampla vantagem, seguida por Feira de Santana e Vitória da Conquista.

1.5.3.4 Top 20 Municípios por Total de Eleitores

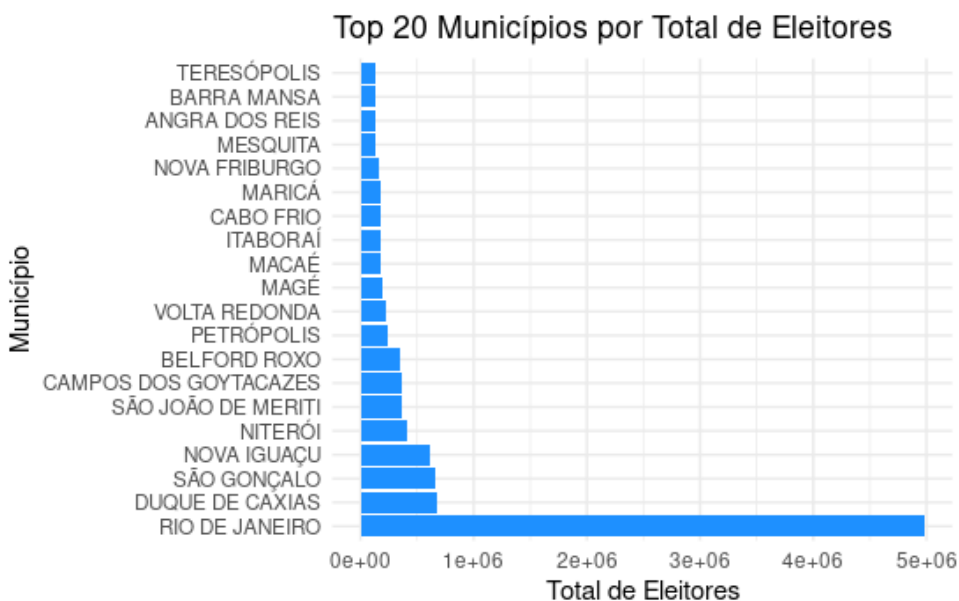


Figura 6 – Os 20 municípios com maior número de eleitores no estado do Rio de Janeiro. Rio de Janeiro, a capital, lidera com ampla vantagem, seguida por Duque de Caxias e São Gonçalo.

1.5.3.5 Distribuição de Eleitores por Grau de Escolaridade

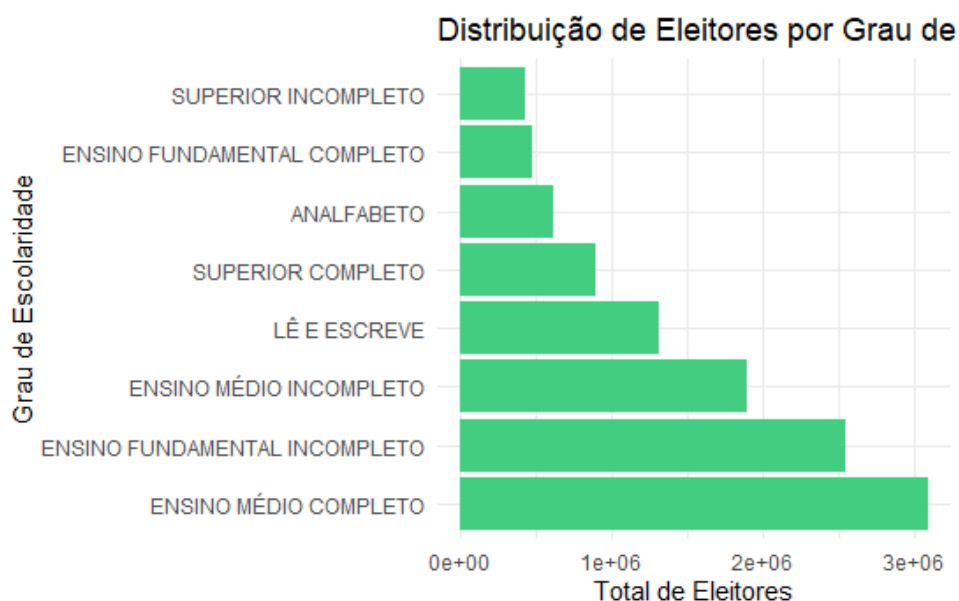


Figura 7 – Distribuição dos eleitores por grau de escolaridade no estado da Bahia. O maior grupo é formado por eleitores com ensino médio completo, seguido por ensino fundamental incompleto.

1.5.3.6 Distribuição de Eleitores por Grau de Escolaridade

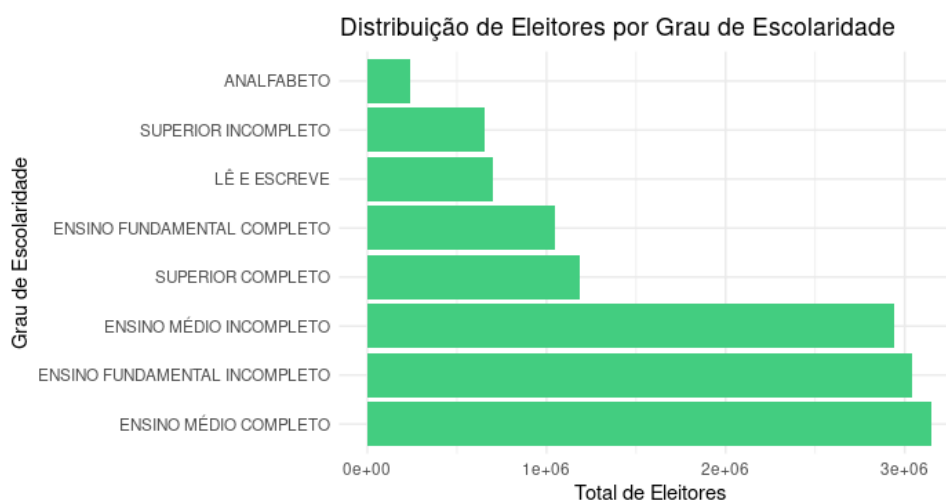


Figura 8 – Distribuição dos eleitores por grau de escolaridade no estado do Rio de Janeiro. O maior grupo é formado por eleitores com ensino médio completo, seguido por ensino fundamental incompleto.

1.5.3.7 Distribuição de Eleitores por Estado Civil

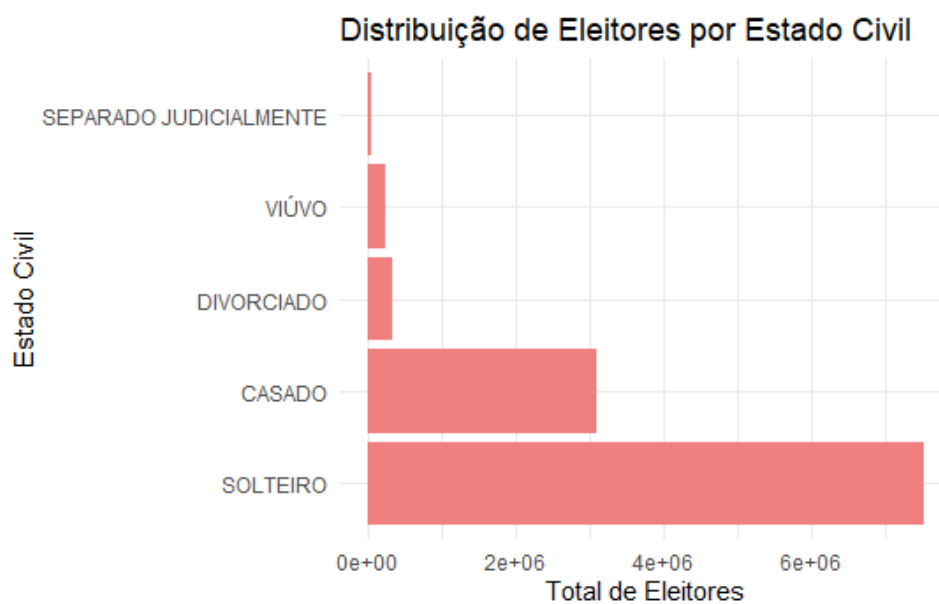


Figura 9 – Distribuição dos eleitores por estado civil no estado da Bahia. Eleitores solteiros formam o maior grupo, seguidos por casados, enquanto os divorciados e viúvos são minoria.

1.5.3.8 Distribuição de Eleitores por Estado Civil

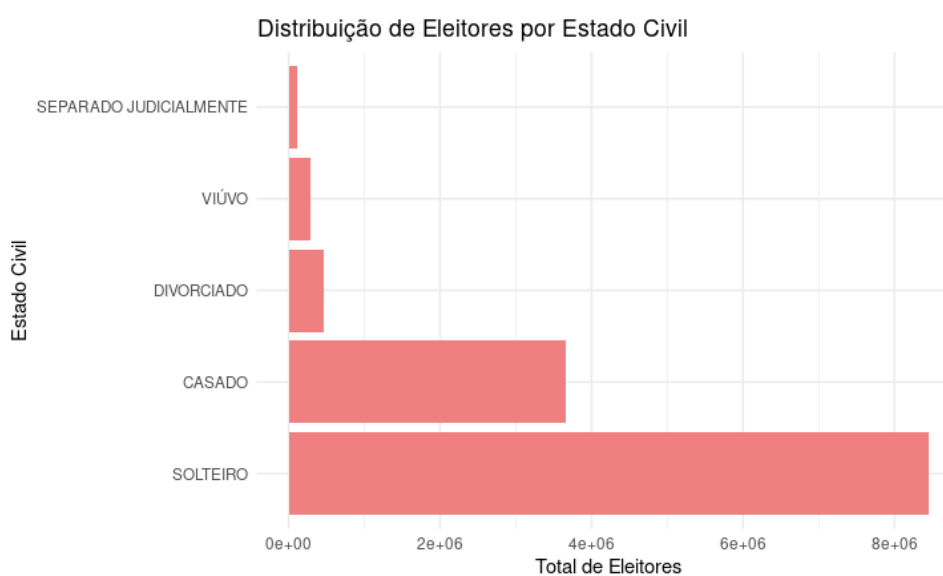


Figura 10 – Distribuição dos eleitores por estado civil no estado do Rio de Janeiro. Eleitores solteiros formam o maior grupo, seguidos por casados, enquanto os divorciados e viúvos são minoria.

1.5.3.9 Distribuição de Eleitores por Faixa Etária

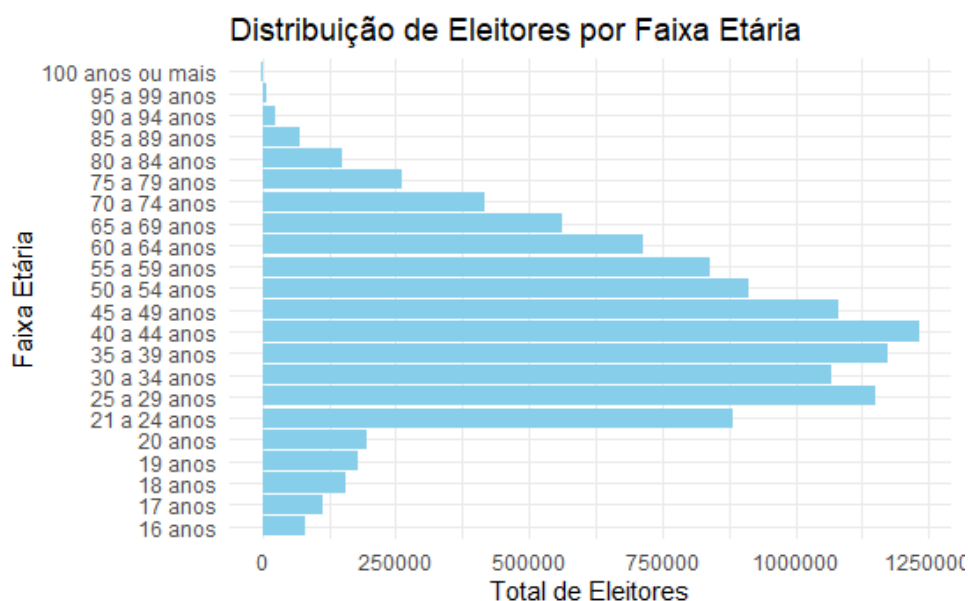


Figura 11 – Distribuição dos eleitores por faixa etária no estado da Bahia. Eleitores entre 25 e 44 anos formam o maior grupo, com uma queda significativa em faixas etárias mais avançadas.

1.5.3.10 Distribuição de Eleitores por Faixa Etária

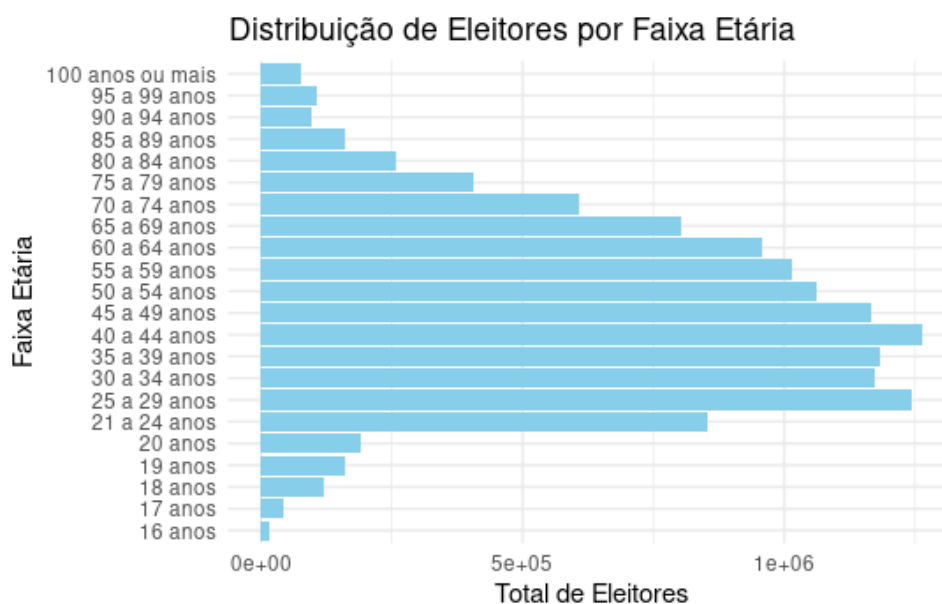


Figura 12 – Distribuição dos eleitores por faixa etária no estado do Rio de Janeiro. Eleitores entre 25 e 44 anos formam o maior grupo, com uma queda significativa em faixas etárias mais avançadas.

1.5.3.11 Distribuição de Eleitores por Gênero

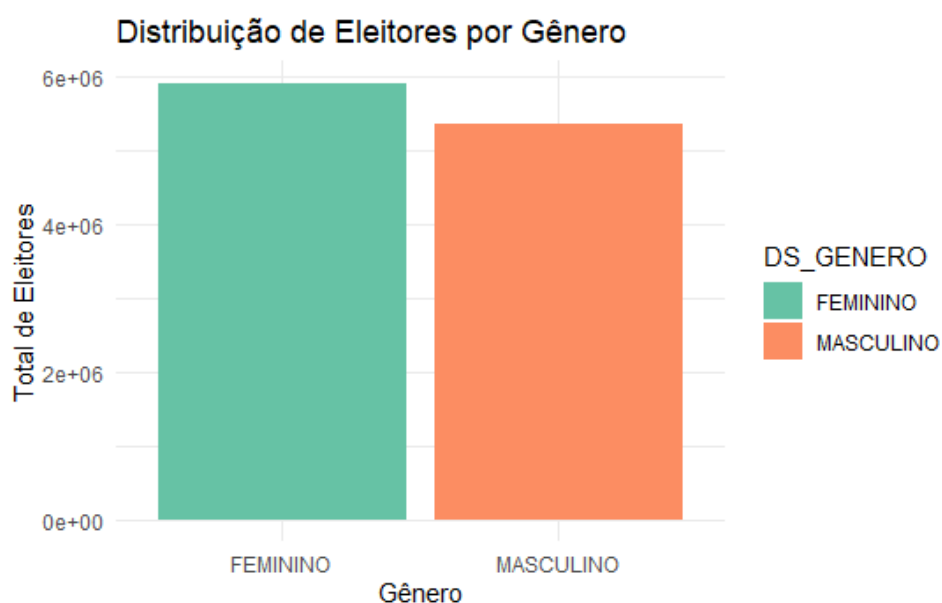


Figura 13 – Comparação da distribuição de eleitores por gênero no estado da Bahia. Eleitoras femininas superam os masculinos por uma margem significativa.

1.5.3.12 Distribuição de Eleitores por Gênero

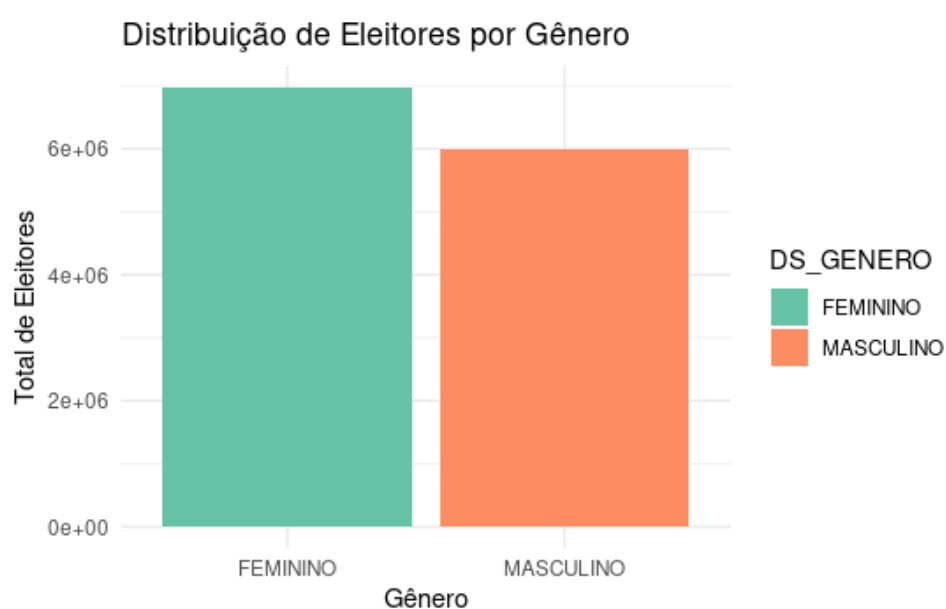


Figura 14 – Comparação da distribuição de eleitores por gênero no estado do Rio de Janeiro. Eleitoras femininas superam os masculinos por uma margem significativa.

1.5.4 Mapas de Calor

Abaixo encontra-se um trecho do código da criação do Mapa de Calor.

```
# Criar o mapa de calor
```

```
ggplot(heatmap_homens, aes(x = DS_FAIXA_ETARIA, y = DS_ESTADO_CIVIL, fill =
  Total_Eleitores)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "lightpink", high = "darkred", name = "Total
    Eleitores") +
  labs(
    title = "Mapa de Calor: Faixa Etária vs Estado Civil (Homens)",
    x = "Faixa Etária",
    y = "Estado Civil"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
```

Listing 1.7 – Mapa de calor com ggplot2

1.5.4.1 Mapa de Calor: Faixa Etária vs Estado Civil (Homens e Mulheres)

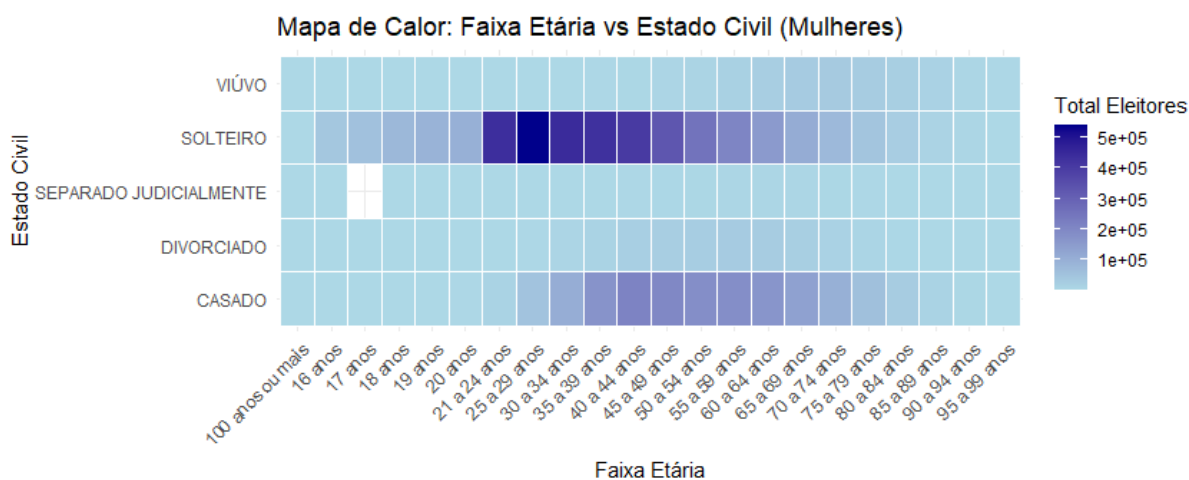


Figura 15 – Distribuição de eleitores mulheres por faixa etária e estado civil no estado da Bahia. Apresenta um padrão semelhante ao dos homens, mas com uma concentração maior de mulheres em faixas etárias mais altas.

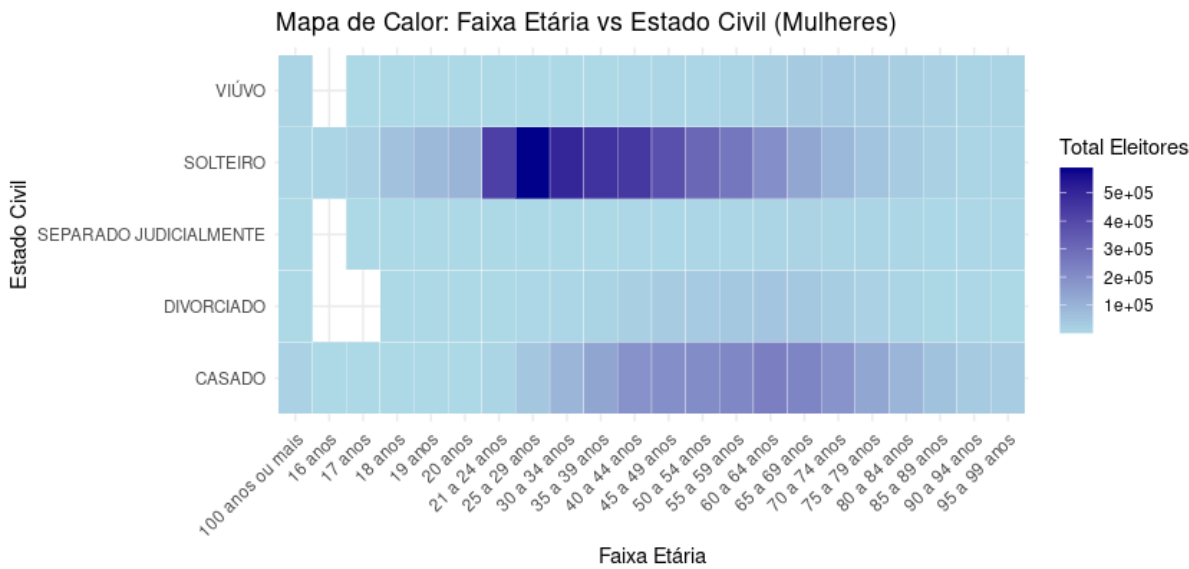


Figura 16 – Distribuição de eleitores mulheres por faixa etária e estado civil no estado do Rio de Janeiro. Apresenta um padrão semelhante ao dos homens, mas com uma concentração maior de mulheres em faixas etárias mais altas.

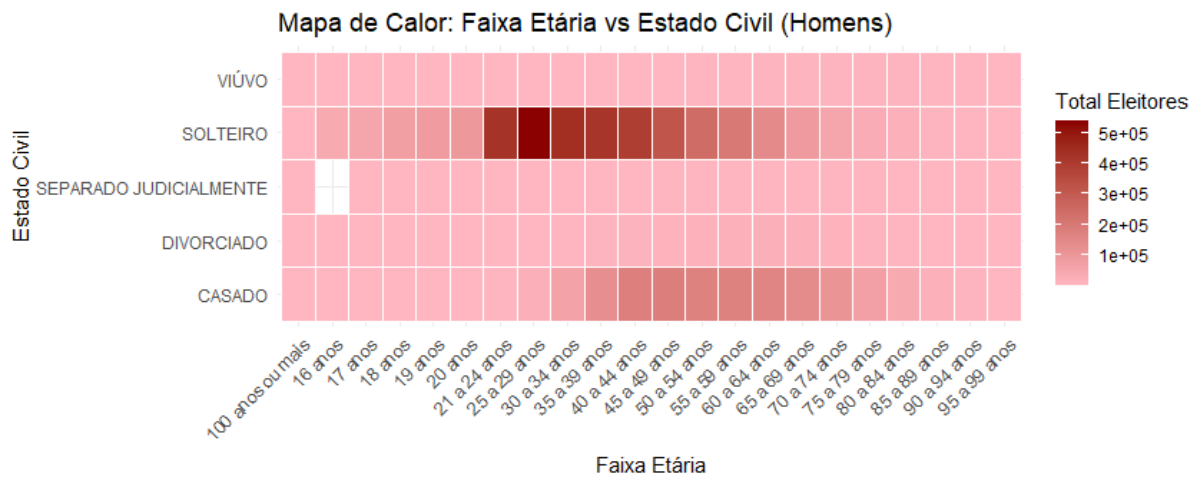


Figura 17 – Distribuição de eleitores homens por faixa etária e estado civil no estado da Bahia. Mostra a concentração de homens solteiros e casados em faixas etárias predominantes, como 21 a 29 anos e 35 a 39 anos.

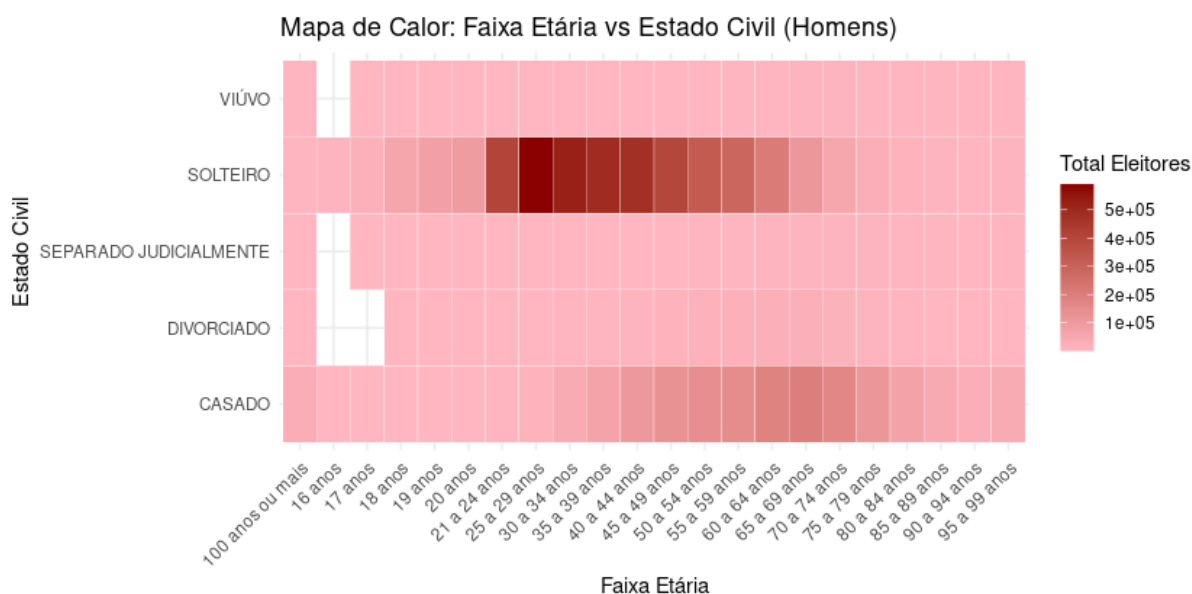


Figura 18 – Distribuição de eleitores homens por faixa etária e estado civil no estado do Rio de Janeiro. Mostra a concentração de homens solteiros e casados em faixas etárias predominantes, como 21 a 29 anos e 35 a 39 anos.

1.6 ANÁLISE FINAL

1.6.1 Análise Comparativa entre os Estados da Bahia e Rio de Janeiro

1.6.1.1 Municípios com Mais e Menos Eleitores

Ao analisar os municípios com maior e menor número de eleitores em ambos os estados, observamos:

- **Bahia:**

- **Município com menos eleitores:** Lajedinho, com 3.636 eleitores.
- **Município com mais eleitores (depois da capital):** Feira de Santana, com 426.034 eleitores.

- **Rio de Janeiro:**

- **Município com menos eleitores:** Macuco, com 7.401 eleitores.
- **Município com mais eleitores (depois da capital):** Duque de Caxias, com 671.417 eleitores.

Análise: Enquanto o menor município da Bahia, Lajedinho, possui quase metade dos eleitores de Macuco no Rio de Janeiro, o maior município da Bahia (após a capital) tem significativamente menos eleitores do que Duque de Caxias, no Rio de Janeiro.

1.6.1.2 Faixa de Variação da Quantidade de Eleitores por Município e Gênero

- **Bahia:**

- **Feminino:** 1.776 a 1.085.599 eleitores, com uma variação de 1.083.823 eleitores.
- **Masculino:** 1.860 a 879.184 eleitores, com uma variação de 877.324 eleitores.

- **Rio de Janeiro:**

- **Feminino:** 3.844 a 2.719.825 eleitores, com uma variação de 2.715.981 eleitores.
- **Masculino:** 3.557 a 2.258.319 eleitores, com uma variação de 2.254.762 eleitores.

Análise: A variação na quantidade de eleitores por gênero é muito maior no Rio de Janeiro do que na Bahia, tanto para o gênero feminino quanto para o masculino. Isso reflete a maior concentração populacional em grandes municípios no Rio de Janeiro.

1.6.1.3 Características que Melhor Descrevem a Maioria dos Eleitores

- **Bahia:**

- **Estado Civil:** Solteiro(a), representando 66,9% do total.
- **Gênero:** Feminino, representando 52,5% do total.
- **Faixa Etária:** 40 a 44 anos, representando 10,9% do total.
- **Escolaridade:** Ensino médio completo, representando 27,5% do total.

- **Rio de Janeiro:**

- **Estado Civil:** Solteiro(a), representando 65,2% do total.
- **Gênero:** Feminino, representando 53,8% do total.
- **Faixa Etária:** 40 a 44 anos, representando 9,8% do total.
- **Escolaridade:** Ensino médio completo, representando 24,3% do total.

Análise: Ambos os estados apresentam maior número de eleitores solteiros e femininos, com a faixa etária predominante sendo de 40 a 44 anos. No entanto, o Rio de Janeiro possui uma maior proporção de mulheres e uma ligeiramente menor concentração de eleitores solteiros. A Bahia tem uma maior proporção de eleitores com ensino médio completo.

1.6.1.4 Conclusão Geral

A comparação entre Bahia e Rio de Janeiro revela semelhanças no perfil geral do eleitorado, mas diferenças significativas em termos de concentração populacional. O Rio de Janeiro apresenta uma maior variação na quantidade de eleitores por município

e gênero, indicando uma desigualdade maior na distribuição populacional entre seus municípios. Já a Bahia tem uma proporção maior de eleitores com ensino médio completo, refletindo possíveis diferenças educacionais entre os dois estados.

2 ANÁLISE DE DADOS METEOROLÓGICOS

2.1 INTRODUÇÃO

Neste capítulo, é realizada a análise comparativa de dados meteorológicos de dois anos consecutivos para uma estação meteorológica específica. O objetivo é identificar variações climáticas significativas.

Observação sobre o UTC:

O UTC não é 0000 durante os períodos matutinos e vespertinos analisados, pois os horários apresentados (como 0300 UTC a 1500 UTC) correspondem à conversão do horário local de Curitiba, que está em UTC-3.

2.2 DESCRIÇÃO DO DATASET

2.2.1 Origem dos Dados

Os dados meteorológicos foram obtidos no site do Instituto Nacional de Meteorologia (INMET), acessível via <https://portal.inmet.gov.br/dadoshistoricos>.

2.2.2 Carga

2.2.2.1 Importação dos Dados

Para realizarmos a importação de dados no R, usando o Tidyverse através do RStudio, fizemos o seguinte:

1. Configuração Inicial

Antes de carregar os dados, foram configurados pacotes e parâmetros para garantir que o arquivo fosse lido corretamente:

- **Pacotes Usados:** `readr`: Parte do tidyverse, facilita a leitura de arquivos CSV com diversas opções. `locale`: Permite ajustar parâmetros regionais, como a codificação de caracteres e o símbolo de separador decimal.
- **Arquivos CSV:** Dois arquivos foram carregados: um com dados de 2022 e outro de 2023, representados pelas variáveis `data_2022` e `data_2023`.

2. Leitura do Arquivo CSV

Foi usada a função **`read_delim()`** do pacote `readr`. Aqui estão as configurações detalhadas: Parâmetros Gerais:

- **file:** Especifica o caminho do arquivo a ser lido (`file_2022` e `file_2023`).

- `delim = ";"`: Define o delimitador de campo como ponto e vírgula (;). Isso é comum em arquivos CSV com formatação brasileira.
- `locale = locale()`: Configurações regionais para a leitura: `decimal_mark = ","`: Define a vírgula como separador decimal, usado frequentemente em arquivos de dados brasileiros. `encoding = "ISO-8859-1"`: Define a codificação de caracteres para leitura correta de acentos e caracteres especiais.

3. Inspeção dos Dados

Após a carga de cada arquivo:

- A função `glimpse(data)` (e também `head(data)`) foi usada para inspecionar a estrutura do dataframe. Essa função: Mostra as primeiras linhas de cada coluna e exibe os tipos de dados detectados automaticamente.

2.2.2.2 Contagem de linhas, colunas e células

Planilha de 2022 - Sem Tratamento:

- **Linhas:** 8760
- **Colunas:** 19
- **Células:** 125411

Planilha de 2023 - Sem Tratamento:

- **Linhas:** 8760
- **Colunas:** 19
- **Células:** 161478

Observação: Pode-se notar que o número de células não corresponde a multiplicação das linhas com as colunas, isso se deve ao fato da existência de alguns valores vazios ou marcados como NA em ambas as tabelas. Vale notar também que há uma vigésima coluna inteiramente em branco nos dados brutos que foi ignorada.

2.2.3 Variáveis Seleccionadas

As variáveis quantitativas seleccionadas para a análise incluem:

- DATA e HORA UTC
- PRECIPITAÇÃO TOTAL, HORÁRIO (mm)
- PRESSÃO ATMOSFÉRICA (mB) (Horária, Max e Min na hora anterior)
- TEMPERATURA DO AR (°C) (Horária, Max e Min na hora anterior)
- VENTO (direção e velocidade horário, rajada máxima)

2.3 TRANSFORMAÇÕES NOS DADOS

2.3.1 Eliminação de Colunas

Usando a função **select()** do pacote dplyr foram removidas colunas que não eram relevantes para a análise, o que otimizou o tamanho do dataset. Após a retirada, o número de células ficaram: 101.560 e 104.363, dos anos de 2022 e 2023 respectivamente.

2.3.2 Eliminação de Linhas Inválidas

Linhas com valores nulos (NA no R) foram eliminadas, utilizando a função **drop_na()** do pacote tidyr incluso no tidyverse, para melhorar a precisão da análise. Após as eliminações, os dados de 2022 ficaram com 8400 (7428 removendo outliers?) linhas, já os de 2023 ficaram com 8681 (7468 removendo outliers?). Para observar a dispersão dos dados bem como os outliers, tanto de 2022 quanto de 2023, geramos alguns boxplots. Utilizando o conceito de primeiro e terceiro quartil, a função boxplot gera os gráficos calculando e apontando automaticamente os outliers.

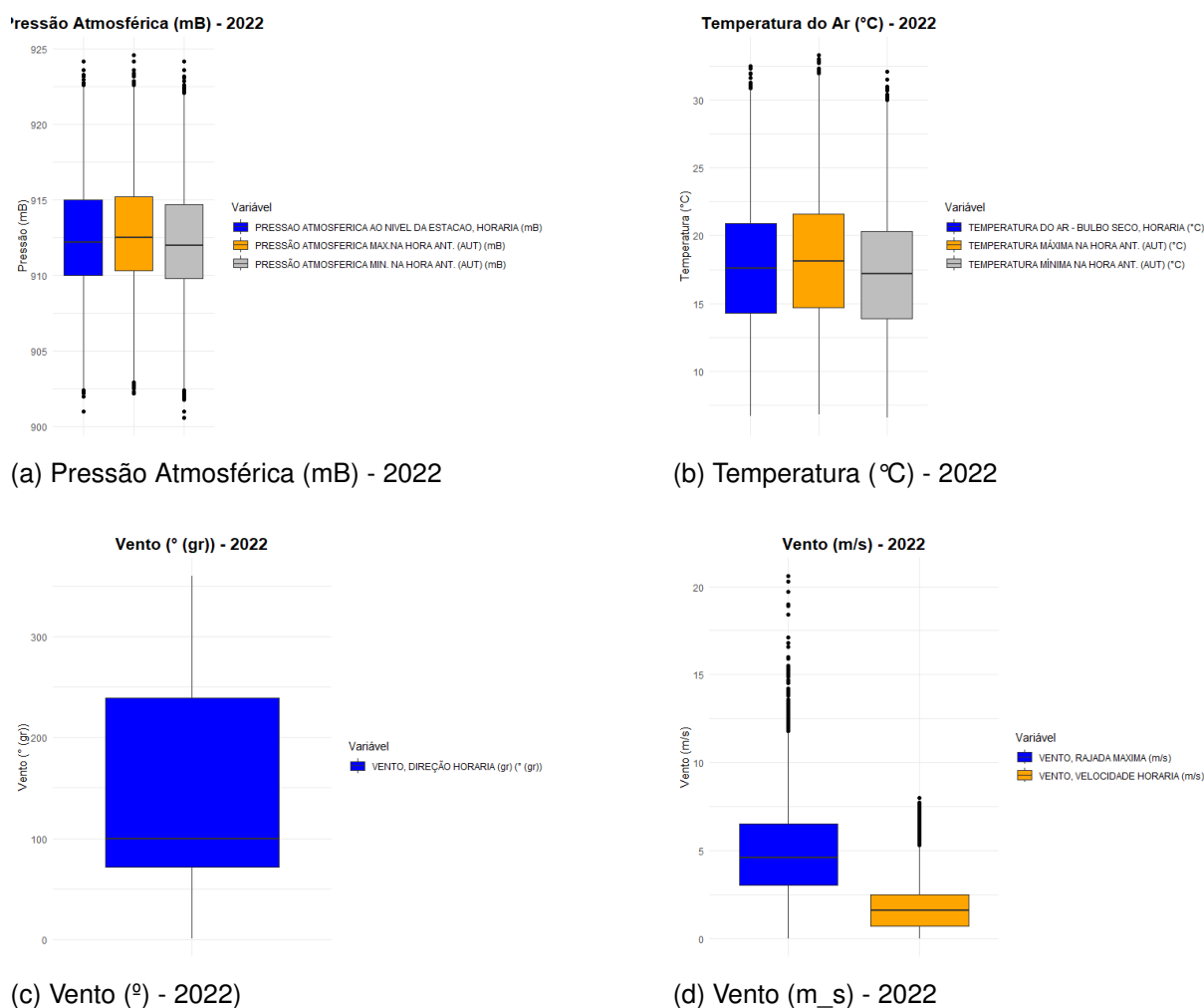
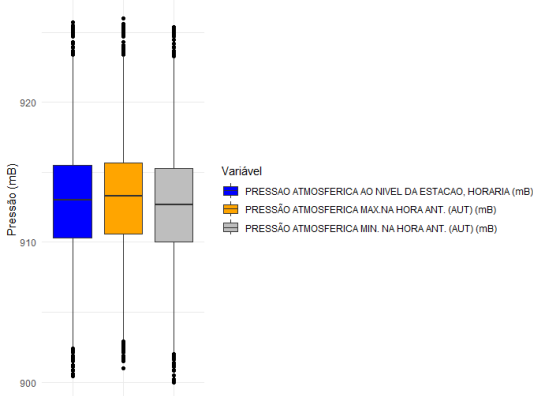


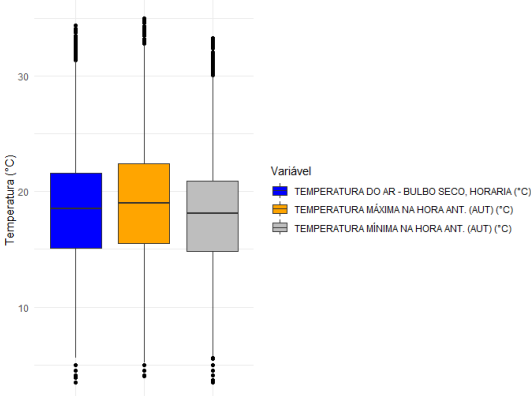
Figura 19 – Organização dos gráficos de 2022

Pressão Atmosférica (mB) - 2023



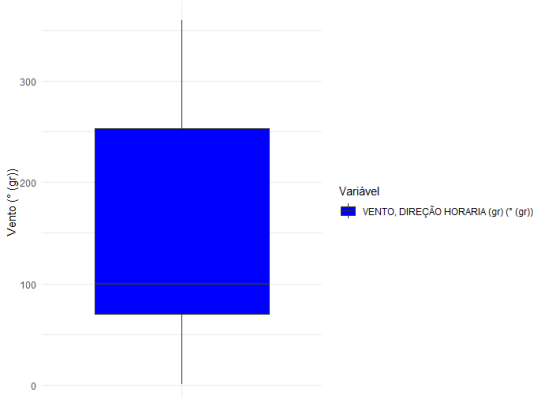
(a) Pressão Atmosférica (mB) - 2023

Temperatura do Ar (°C) - 2023



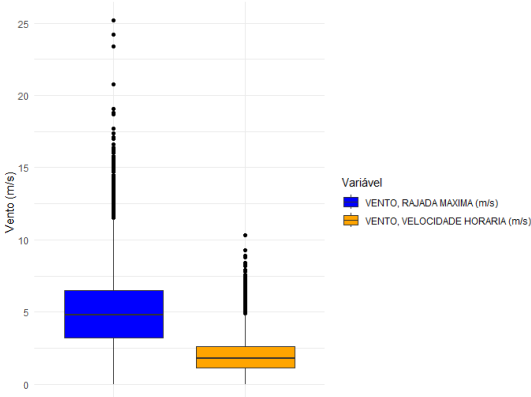
(b) Temperatura (°C) - 2023

Vento (° gr) - 2023



(c) Vento (°) - 2023

Vento (m/s) - 2023



(d) Vento (m_s) - 2023

Figura 20 – Organização dos gráficos de 2023

2.4 ANÁLISES REALIZADAS

2.4.1 Análise Trimestral da Temperatura do Ar

Trimestres	Máximo	Mínimo	Média
Trimestre 1	32.50	10.80	21.43
Trimestre 2	28.40	6.70	16.13
Trimestre 3	29.20	6.90	15.36
Trimestre 4	30.90	7.60	18.45
Total Geral	32.50	6.70	17.90

Tabela 1 – Máximos, Mínimos e Médias de Temperatura por Trimestre de 2022

Trimestres	Máximo	Mínimo	Média
Trimestre 1	30.90	12.90	21.13
Trimestre 2	28.10	6.50	16.74
Trimestre 3	33.00	3.50	16.51
Trimestre 4	34.40	8.70	20.12
Total Geral	34.40	3.50	18.63

Tabela 2 – Máximos, Mínimos e Médias de Temperatura por Trimestre de 2023

2.4.2 Cálculos das Estatísticas Básicas

2022	Média	Moda	Mediana
PRECIPITAÇÃO TOTAL, HORÁRIO (mm)	0.20	0.00	0.00
P.Atm AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)	912.50	911.10	912.20
P.Atm MÁX. NA HORA ANT. (AUT) (mB)	912.70	911.80	912.50
P.Atm MÍN. NA HORA ANT. (AUT) (mB)	912.20	911.10	912.00
TEMP DO AR - BULBO SECO, HORÁRIA (°C)	17.90	17.30	17.60
TEMP MÁXIMA NA HORA ANT. (AUT) (°C)	18.50	17.60	18.10
TEMP MÍNIMA NA HORA ANT. (AUT) (°C)	17.40	17.20	17.20
VENTO, DIREÇÃO HORÁRIA (gr) (° (gr))	142.10	82.00	100.00
VENTO, RAJADA MÁXIMA (m/s)	4.90	0.00	4.60
VENTO, VELOCIDADE HORÁRIA (m/s)	1.70	0.00	1.60

Tabela 3 – Estatísticas Descritivas para 2022

2022	Variância	Desvio Padrão	Amplitude
PRECIPITAÇÃO TOTAL, HORÁRIO (mm)	2.20	1.50	51.20
P.Atm AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)	12.90	3.60	23.20
P.Atm MÁX. NA HORA ANT. (AUT) (mB)	12.80	3.60	22.40
P.Atm MÍN. NA HORA ANT. (AUT) (mB)	13.00	3.60	23.60
TEMP DO AR - BULBO SECO, HORÁRIA (°C)	21.90	4.70	25.80
TEMP MÁXIMA NA HORA ANT. (AUT) (°C)	23.80	4.90	26.50
TEMP MÍNIMA NA HORA ANT. (AUT) (°C)	20.10	4.50	25.50
VENTO, DIREÇÃO HORÁRIA (gr) (° (gr))	9989.20	99.90	359.00
VENTO, RAJADA MÁXIMA (m/s)	6.90	2.60	20.60
VENTO, VELOCIDADE HORÁRIA (m/s)	1.60	1.30	8.00

Tabela 4 – Estatísticas Descritivas para 2022

2022	IQR	Máximo	Mínimo
PRECIPITAÇÃO TOTAL, HORÁRIO (mm)	0.00	51.20	0.00
P.Atm AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)	5.00	924.20	901.00
P.Atm MÁX. NA HORA ANT. (AUT) (mB)	4.8	919.8	905.2
P.Atm MÍN. NA HORA ANT. (AUT) (mB)	4.90	924.20	900.60
TEMP DO AR - BULBO SECO, HORÁRIA (°C)	6.60	32.50	6.70
TEMP MÁXIMA NA HORA ANT. (AUT) (°C)	6.90	33.30	6.80
TEMP MÍNIMA NA HORA ANT. (AUT) (°C)	6.40	32.10	6.60
VENTO, DIREÇÃO HORÁRIA (gr) (° (gr))	167.20	360.00	1.00
VENTO, RAJADA MÁXIMA (m/s)	3.50	20.60	0.00
VENTO, VELOCIDADE HORÁRIA (m/s)	1.80	8.00	0.00

Tabela 5 – Estatísticas Descritivas para 2022

2023	Média	Moda	Mediana
PRECIPITAÇÃO TOTAL, HORÁRIO (mm)	0.20	0.00	0.00
P.Atm AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)	912.80	912.70	913.00
P.Atm MÁX. NA HORA ANT. (AUT) (mB)	913.10	913.90	913.30
P.Atm MÍN. NA HORA ANT. (AUT) (mB)	912.60	913.60	912.70
TEMP DO AR - BULBO SECO, HORÁRIA (°C)	18.60	19.50	18.50
TEMP MÁXIMA NA HORA ANT. (AUT) (°C)	19.20	19.50	19.00
TEMP MÍNIMA NA HORA ANT. (AUT) (°C)	18.10	18.60	18.10
VENTO, DIREÇÃO HORÁRIA (gr) (° (gr))	146.20	84.00	100.00
VENTO, RAJADA MÁXIMA (m/s)	5.10	4.10	4.80
VENTO, VELOCIDADE HORÁRIA (m/s)	1.90	1.70	1.80

Tabela 6 – Estatísticas Descritivas para 2023

2023	Variância	Desvio Padrão	Amplitude
PRECIPITAÇÃO TOTAL, HORÁRIO (mm)	1.20	1.10	30.20
P.Atm AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)	14.40	3.80	25.30
P.Atm MÁX. NA HORA ANT. (AUT) (mB)	14.10	3.80	25.00
P.Atm MÍN. NA HORA ANT. (AUT) (mB)	14.60	3.80	25.40
TEMP DO AR - BULBO SECO, HORÁRIA (°C)	23.90	4.90	30.90
TEMP MÁXIMA NA HORA ANT. (AUT) (°C)	25.70	5.10	31.00
TEMP MÍNIMA NA HORA ANT. (AUT) (°C)	22.20	4.70	29.80
VENTO, DIREÇÃO HORÁRIA (gr) (° (gr))	10599.00	103.00	359.00
VENTO, RAJADA MÁXIMA (m/s)	6.80	2.60	25.20
VENTO, VELOCIDADE HORÁRIA (m/s)	1.60	1.20	10.30

Tabela 7 – Estatísticas Descritivas para 2023

2023	IQR	Máximo	Mínimo
PRECIPITAÇÃO TOTAL, HORÁRIO (mm)	0.00	30.20	0.00
P.Atm AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)	5.20	925.70	900.40
P.Atm MÁX. NA HORA ANT. (AUT) (mB)	5.10	926.00	901.00
P.Atm MÍN. NA HORA ANT. (AUT) (mB)	5.30	925.40	900.00
TEMPERATURA DO AR - BULBO SECO, HORÁRIA (°C)	6.50	34.40	3.50
TEMP MÁXIMA NA HORA ANT. (AUT) (°C)	6.90	35.00	4.00
TEMP MÍNIMA NA HORA ANT. (AUT) (°C)	6.10	33.30	3.50
VENTO, DIREÇÃO HORÁRIA (gr) (° (gr))	183.00	360.00	1.00
VENTO, RAJADA MÁXIMA (m/s)	3.30	25.20	0.00
VENTO, VELOCIDADE HORÁRIA (m/s)	1.50	10.30	0.00

Tabela 8 – Estatísticas Descritivas para 2023

2.5 ANÁLISE COMPARATIVA ENTRE 2022 E 2023

2.5.1 Precipitação Total

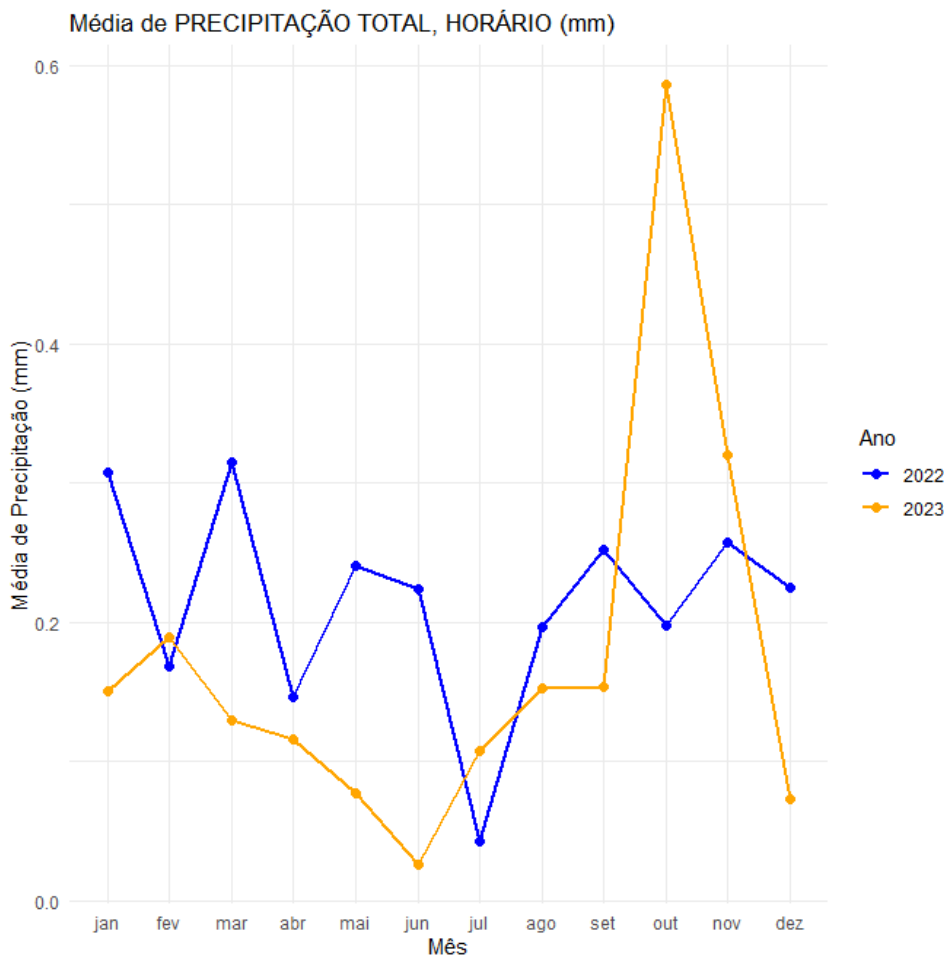


Figura 21 – Precipitação Ano

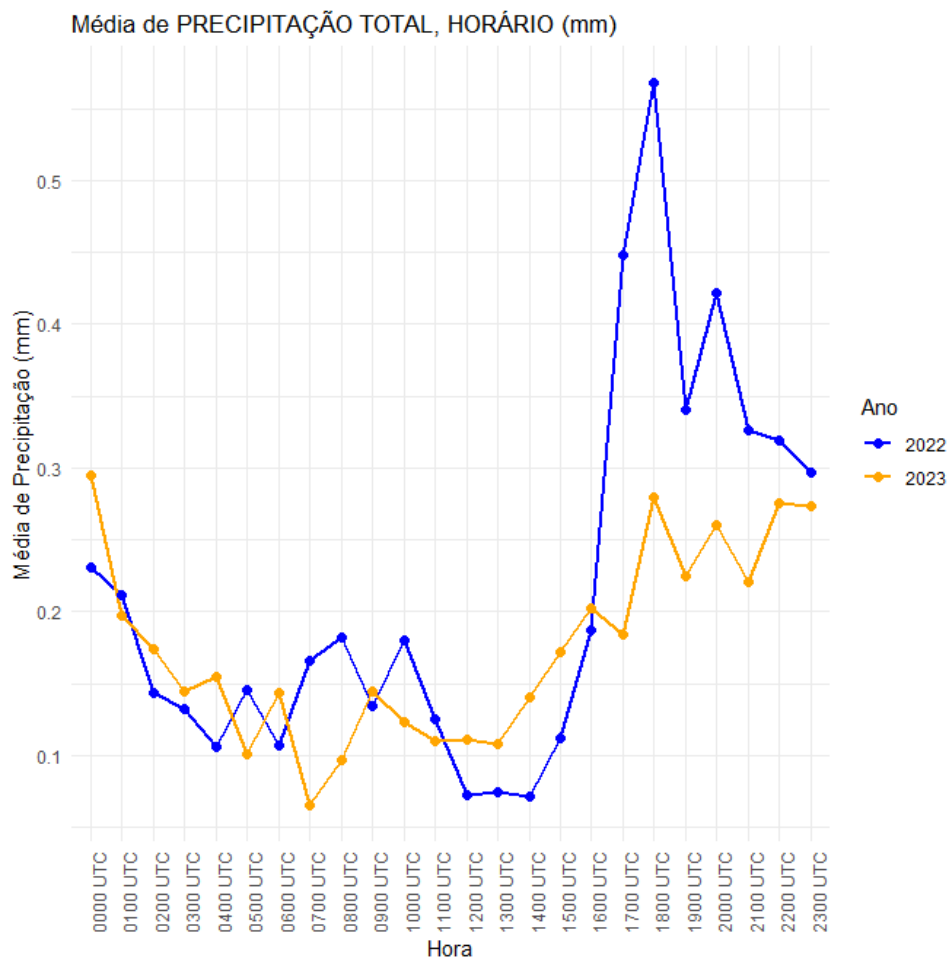


Figura 22 – Precipitação Dia

Os dois gráficos apresentam a média de precipitação total (mm) para os anos de 2022 e 2023, com o primeiro gráfico focado na variação mensal e o segundo na variação horária (UTC). A análise conjunta revela um padrão de variações mais acentuadas em 2023 em ambas as escalas temporais.

2.5.1.1 Tendência Geral ao Longo do Ano

Em **2022**, a precipitação se manteve relativamente estável ao longo dos meses, sem grandes picos, refletindo um comportamento mais uniforme. Em **2023**, por outro lado, apresentou-se uma variação significativa ao longo do ano, especialmente a partir de **setembro**, com um grande **pico de precipitação em outubro**, seguido por uma queda brusca em novembro e dezembro.

2.5.1.2 Comparação Mensal e Horária

O primeiro gráfico mostra que, enquanto nos primeiros meses de ambos os anos (de janeiro a agosto) as médias mensais de precipitação são bastante semelhantes, **2023** se destaca com um **aumento acentuado em outubro**. Esse comportamento é

reforçado pelo segundo gráfico, que mostra a variação diária. Nele, em **2023**, há dois picos principais de precipitação por hora: um no início da madrugada e outro mais significativo no final da tarde, que não aparecem em 2022.

2.5.1.3 Padrão Horário de 2023

Em termos de precipitação horária, **2023** apresenta variações mais pronunciadas ao longo do dia. O aumento mais expressivo ocorre no final da tarde, indicando que, além de maior volume anual, há também uma concentração da precipitação em horários específicos. **2022** mantém um comportamento muito mais estável em todas as horas do dia, com pouca variação entre os diferentes horários.

2.5.1.4 Pontos de Destaque

O comportamento de **2023**, tanto em termos de meses quanto de horas, sugere eventos de chuva mais intensos e concentrados, especialmente no **outono** e em horários específicos do dia. O pico de **outubro de 2023** no primeiro gráfico é particularmente notável e coincide com o aumento de precipitação nas horas da tarde, conforme evidenciado no segundo gráfico. Além disso a média anual de aumentou em 70% de 2022 para 2023.

O ano de **2023** foi marcado por padrões de precipitação mais variáveis e intensos, tanto ao longo do ano quanto do dia, com destaque para o mês de **outubro** e as **tardes**. Já **2022** apresentou uma precipitação mais estável e previsível, sem grandes oscilações.

2.5.2 Pressão Atmosférica

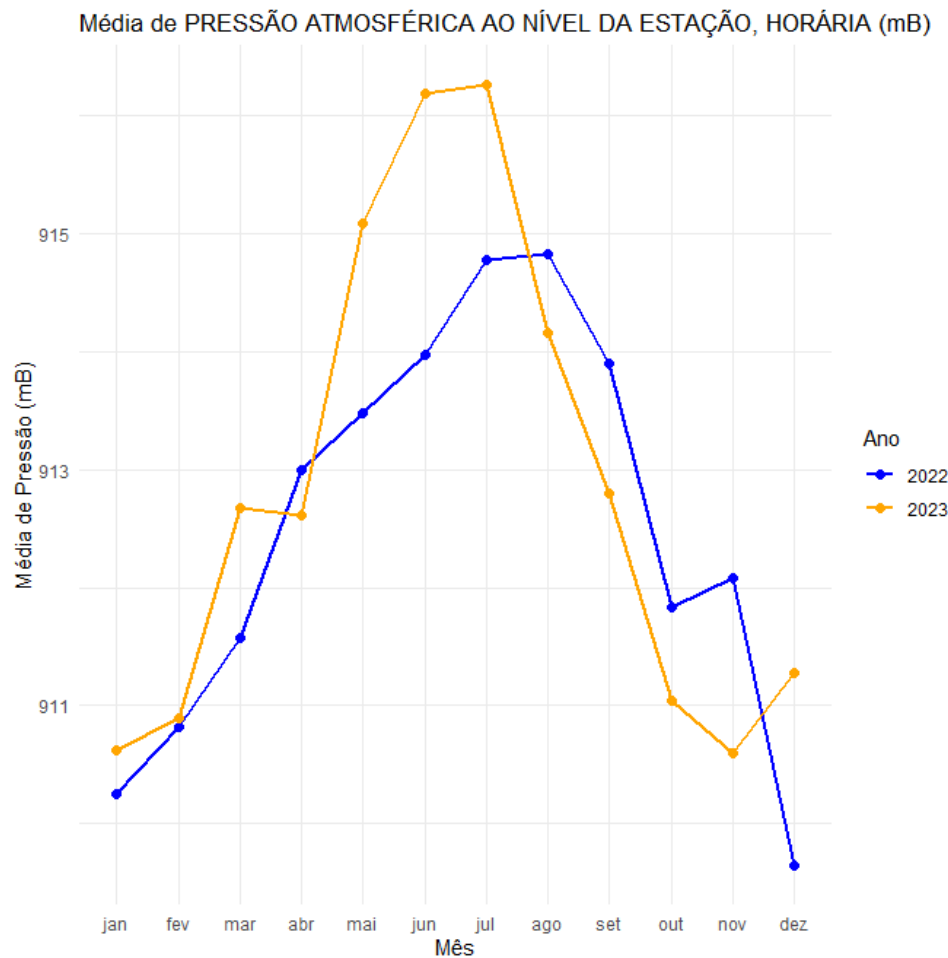


Figura 23 – Pressão Ano

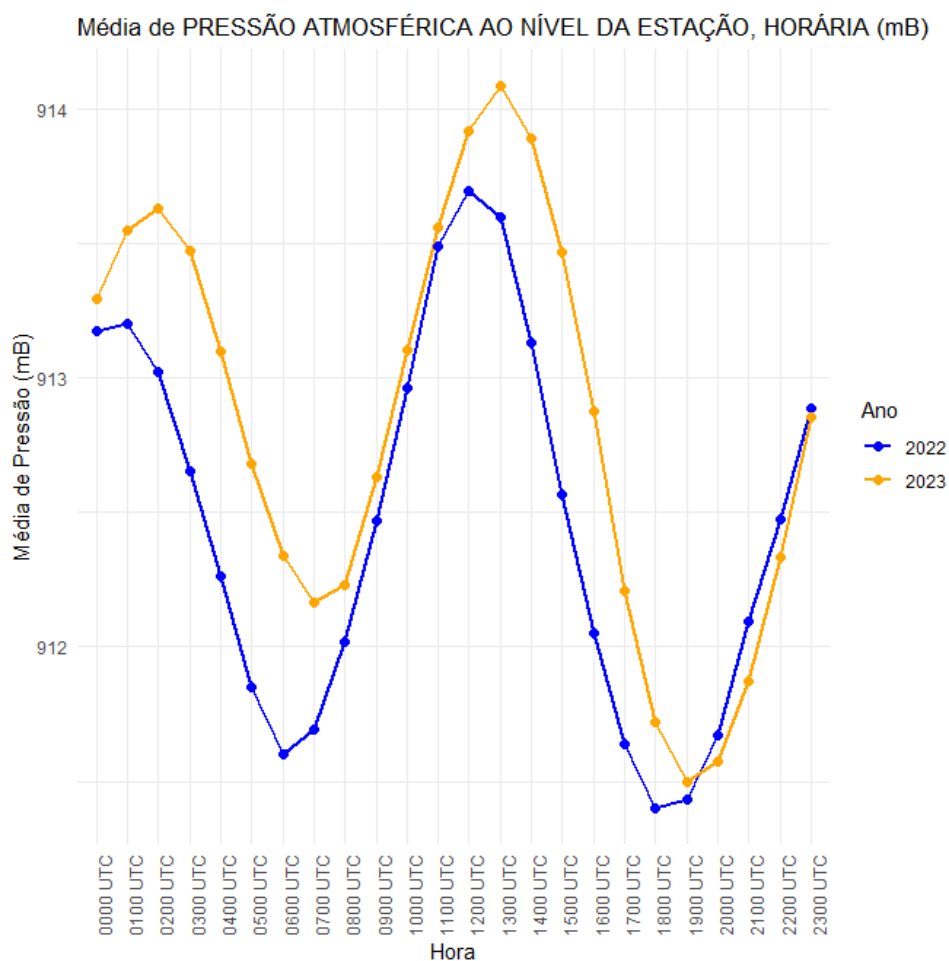


Figura 24 – Pressão Dia

Variação Horária

Tendências Observadas:

- Em ambos os anos, a pressão atmosférica é mais baixa durante a manhã (por volta das 05:00 BRT) e atinge seu pico à tarde (entre 11:00 e 12:00 BRT).
- Após o pico à tarde, a pressão diminui novamente até a meia-noite.

Comparação:

- A pressão atmosférica em 2023 tende a ser ligeiramente mais alta do que em 2022, especialmente durante as horas da tarde, onde a diferença chega a ser de aproximadamente 1 mB.
- A amplitude das variações diárias é maior em 2023, com um pico mais acentuado e uma depressão menos pronunciada comparada a 2022.

Pontos de Destaque:

- Em 2022, a pressão atmosférica atinge seu ponto mais baixo (~ 911.5 mB) por volta das 05:00 BRT e seu pico (~ 913.5 mB) às 12:00 BRT.

- Em 2023, a pressão atmosférica atinge seu ponto mais baixo (~ 912 mB) por volta das 05:00 BRT e seu pico (~ 914.5 mB) às 11:00 BRT.
- A diferença mais notável é durante o pico da tarde, onde 2023 mostra uma pressão mais alta comparada a 2022.

Variação Mensal

Tendências Observadas:

- Em ambos os anos, há um aumento gradual na pressão atmosférica de janeiro a meio do ano (junho/julho), seguido por uma diminuição na segunda metade do ano.
- O pico anual ocorre em julho para 2022 e junho para 2023.

Comparação:

- A pressão atmosférica média mensal em 2023 é geralmente mais alta do que em 2022, com uma diferença notável no meio do ano (junho/julho), onde a pressão atinge cerca de 916 mB em 2023 comparado a 914 mB em 2022.
- A queda na pressão atmosférica após o pico de meio ano é mais acentuada em 2023 do que em 2022, resultando em valores similares no final do ano.

Pontos de Destaque:

- Em 2022, a pressão atinge seu ponto mais baixo (~ 910 mB) em janeiro e dezembro, e seu pico (~ 914 mB) em julho.
- Em 2023, a pressão atinge seu ponto mais baixo (~ 910 mB) em janeiro e dezembro, e seu pico (~ 916 mB) em junho.
- Os valores de pressão atmosférica em 2023 são consistentemente mais altos que em 2022, especialmente no primeiro semestre do ano.
- A pressão atmosférica em 2023 apresenta um aumento mais rápido no início do ano e uma queda mais pronunciada no final do ano.

Resumo Final

Os dois gráficos mostram padrões consistentes de variação horária e mensal da pressão atmosférica entre os anos de 2022 e 2023, com 2023 exibindo valores ligeiramente mais altos em ambos os períodos. As tendências diárias e sazonais são similares, mas 2023 apresenta uma maior amplitude nas variações horárias e um pico mais elevado no meio do ano. Essas observações indicam que, apesar das variações diárias e sazonais, os padrões gerais se mantêm consistentes entre os anos.

3 RESULTADO DAS ANÁLISE

3.1 LINK PARA O REPOSITÓRIO

Repositório Git que se encontra os códigos. Se encontra junto ao primeiro trabalho com atualização da pasta R-Analysis e README.

<https://github.com/dhaysetito/TSE-INMET-Analysis>.

ANEXO A: CÓDIGO EM R PARA ANÁLISE DO PERFIL DO ELEITORADO DA BAHIA E RIO DE JANEIRO

Nesse anexo encontra-se o código para análise do estado do Rio de Janeiro. O código para análise do Bahia é similar, modificando apenas o dataset o qual é carregado.

```
# -----  
# ETAPA INICIAL: Carga  
# -----  
  
# Carregar o pacote necessário  
library(tidyverse)  
  
# Caminho do dataset  
caminho_dataset <- "perfil_eleitor_secao_ATUAL_RJ.csv"  
  
# Carregar o dataset  
dataset <- read_delim(  
  file = caminho_dataset,  
  delim = ";",  
  quote = "\"",  
  locale = locale(encoding = "ISO-8859-1")  
)  
  
# Exibir uma amostra do dataset  
print(head(dataset))  
  
# -----  
# ETAPA 1: Preparação e Limpeza dos Dados  
# -----  
  
# Dimensões do dataset antes da projeção  
linhas_antes <- nrow(dataset)  
colunas_antes <- ncol(dataset)  
celulas_antes <- linhas_antes * colunas_antes  
  
# Selecionar colunas de interesse  
colunas_interesse <- c(  
  "NM_MUNICIPIO", "CD_GENERO", "DS_GENERO",  
  "CD_ESTADO_CIVIL", "DS_ESTADO_CIVIL",  
  "CD_FAIXA_ETARIA", "DS_FAIXA_ETARIA",
```

```
"CD_GRAU_ESCOLARIDADE", "DS_GRAU_ESCOLARIDADE",
"QT_ELEITORES_PERFIL"
)

dataset_reduzido <- dataset %>%
  select(all_of(colunas_interesse))

# Remover linhas com dados inválidos
dataset_final <- dataset_reduzido %>%
  filter(
    DS_FAIXA_ETARIA != "Inválido",
    !if_any(everything(), ~ . == "NÃO INFORMADO")
  )

# Dimensões finais do dataset
linhas_finais <- nrow(dataset_final)
colunas_finais <- ncol(dataset_final)
celulas_finais <- linhas_finais * colunas_finais

# -----
# ETAPA 2: Análise por Critérios Específicos
# -----

# Total geral de eleitores
total_geral <- sum(dataset_final$QT_ELEITORES_PERFIL, na.rm = TRUE)

# Critérios específicos
critérios <- list(
  "Homens, 35-39 anos, casados, ensino médio completo" = dataset_final %>%
    filter(
      DS_GENERO == "MASCULINO",
      DS_FAIXA_ETARIA == "35 a 39 anos",
      DS_ESTADO_CIVIL == "CASADO",
      DS_GRAU_ESCOLARIDADE == "ENSINO MÉDIO COMPLETO"
    ),
  "Mulheres, 45-49 anos, solteiras, ensino superior completo" = dataset_final
    %>%
    filter(
      DS_GENERO == "FEMININO",
      DS_FAIXA_ETARIA == "45 a 49 anos",
      DS_ESTADO_CIVIL == "SOLTEIRO",
      DS_GRAU_ESCOLARIDADE == "SUPERIOR COMPLETO"
```

```

    ),
    "Homens, 25-29 anos, divorciados, ensino fundamental incompleto" =
    dataset_final %>%
    filter(
      DS_GENERO == "MASCULINO",
      DS_FAIXA_ETARIA == "25 a 29 anos",
      DS_ESTADO_CIVIL == "DIVORCIADO",
      DS_GRAU_ESCOLARIDADE == "ENSINO FUNDAMENTAL INCOMPLETO"
    ),
    "Ambos os gêneros, 60-64 anos, viúvos, lê e escreve" = dataset_final %>%
    filter(
      DS_FAIXA_ETARIA == "60 a 64 anos",
      DS_ESTADO_CIVIL == "VIÚVO",
      DS_GRAU_ESCOLARIDADE == "LÊ E ESCRIVE"
    ),
    "Ambos os gêneros, 21-24 anos, analfabeto" = dataset_final %>%
    filter(
      DS_FAIXA_ETARIA == "21 a 24 anos",
      DS_GRAU_ESCOLARIDADE == "ANALFABETO"
    )
  )

# Calcular os totais e porcentagens para cada critério
resultados_criterios <- map_dfr(
  criterios,
  ~ data.frame(
    Total = sum(.x$QT_ELEITORES_PERFIL, na.rm = TRUE),
    Porcentagem = sum(.x$QT_ELEITORES_PERFIL, na.rm = TRUE) / total_geral * 100
  ),
  .id = "Critério"
)

# -----
# ETAPA 3: Contagens e Estatísticas
# -----

# Contagem de eleitores por gênero
contagem_por_sexo <- dataset_final %>%
  group_by(DS_GENERO) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))

# Gráfico: Total de eleitores por gênero

```

```
ggplot(data = contagem_por_sexo, aes(x = DS_GENERO, y = Total_Eleitores, fill =
DS_GENERO)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Distribuição de Eleitores por Gênero",
    x = "Gênero",
    y = "Total de Eleitores"
  ) +
  theme_minimal() +
  scale_fill_brewer(palette = "Set2")

# Contagem de eleitores por idade
contagem_por_idade <- dataset_final %>%
  group_by(DS_FAIXA_ETARIA) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))

# Contagem por idade
contagem_por_idade <- contagem_por_idade %>%
  mutate(DS_FAIXA_ETARIA = factor(DS_FAIXA_ETARIA, levels = c(
    "16 anos", "17 anos", "18 anos", "19 anos", "20 anos",
    "21 a 24 anos", "25 a 29 anos", "30 a 34 anos", "35 a 39 anos",
    "40 a 44 anos", "45 a 49 anos", "50 a 54 anos", "55 a 59 anos",
    "60 a 64 anos", "65 a 69 anos", "70 a 74 anos", "75 a 79 anos",
    "80 a 84 anos", "85 a 89 anos", "90 a 94 anos", "95 a 99 anos",
    "100 anos ou mais"
  )))

# Criar o gráfico com as faixas etárias organizadas
ggplot(data = contagem_por_idade, aes(x = DS_FAIXA_ETARIA, y =
Total_Eleitores)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  labs(
    title = "Distribuição de Eleitores por Faixa Etária",
    x = "Faixa Etária",
    y = "Total de Eleitores"
  ) +
  theme_minimal() +
  coord_flip()

# Contagem de eleitores por estado civil
contagem_por_estado_civil <- dataset_final %>%
  group_by(DS_ESTADO_CIVIL) %>%
```

```
summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))

# Gráfico: Total de eleitores por estado civil
ggplot(data = contagem_por_estado_civil, aes(x = reorder(DS_ESTADO_CIVIL,
-Total_Eleitores), y = Total_Eleitores)) +
  geom_bar(stat = "identity", fill = "lightcoral") +
  labs(
    title = "Distribuição de Eleitores por Estado Civil",
    x = "Estado Civil",
    y = "Total de Eleitores"
  ) +
  theme_minimal() +
  coord_flip()

# Contagem de eleitores por grau de escolaridade
contagem_por_escolaridade <- dataset_final %>%
  group_by(DS_GRAU_ESCOLARIDADE) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))

# Gráfico: Total de eleitores por grau de escolaridade
ggplot(data = contagem_por_escolaridade, aes(x = reorder(DS_GRAU_ESCOLARIDADE,
-Total_Eleitores), y = Total_Eleitores)) +
  geom_bar(stat = "identity", fill = "seagreen3") +
  labs(
    title = "Distribuição de Eleitores por Grau de Escolaridade",
    x = "Grau de Escolaridade",
    y = "Total de Eleitores"
  ) +
  theme_minimal() +
  coord_flip()

# Contagem de eleitores por município
contagem_por_municipio <- dataset_final %>%
  group_by(NM_MUNICIPIO) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))

# Top 20 municípios com mais eleitores
top_municipios <- contagem_por_municipio %>%
  arrange(desc(Total_Eleitores)) %>%
  slice(1:20)

# Gráfico: Top 20 municípios por total de eleitores
```

```
ggplot(data = top_municipios, aes(x = reorder(NM_MUNICIPIO, -Total_Eleitores),
y = Total_Eleitores)) +
  geom_bar(stat = "identity", fill = "dodgerblue") +
  labs(
    title = "Top 20 Municípios por Total de Eleitores",
    x = "Município",
    y = "Total de Eleitores"
  ) +
  theme_minimal() +
  coord_flip()
```

```
# Exibir as tabelas
```

```
contagem_por_sexo
```

```
contagem_por_idade
```

```
contagem_por_estado_civil
```

```
contagem_por_escolaridade
```

```
contagem_por_municipio
```

```
# Contagem de eleitores por genero e idade
```

```
contagem_sexo_idade <- dataset_final %>%
```

```
  group_by(DS_GENERO, DS_FAIXA_ETARIA) %>%
```

```
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))
```

```
# Transformar valores de um dos gêneros em negativos
```

```
contagem_sexo_idade <- contagem_sexo_idade %>%
```

```
  mutate(
```

```
    Total_Eleitores = ifelse(DS_GENERO == "MASCULINO", -Total_Eleitores,
    Total_Eleitores)
```

```
  )
```

```
# Gráfico: Gênero por faixa etária
```

```
ggplot(data = contagem_sexo_idade, aes(x = DS_FAIXA_ETARIA, y =
Total_Eleitores, fill = DS_GENERO)) +
```

```
  geom_bar(stat = "identity") +
```

```
  labs(
```

```
    title = "Distribuição de Gênero por Faixa Etária",
```

```
    x = "Faixa Etária",
```

```
    y = "Total de Eleitores"
```

```
  ) +
```

```
  scale_y_continuous(labels = abs) +
```

```
theme_minimal() +
scale_fill_brewer(palette = "Set1") +
coord_flip()

# Contagem de eleitores por sexo e município
contagem_sexo_municipio <- dataset_final %>%
  group_by(DS_GENERO, NM_MUNICIPIO) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))

# Contagem de eleitores por grau de escolaridade e estado civil
contagem_escolaridade_estado <- dataset_final %>%
  group_by(DS_GRAU_ESCOLARIDADE, DS_ESTADO_CIVIL) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE))

# Gráfico: Porcentagem por estado civil
ggplot(data = contagem_por_estado_civil, aes(x = "", y = Total_Eleitores, fill
= DS_ESTADO_CIVIL)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y", start = 0) +
  labs(
    title = "Distribuição Percentual por Estado Civil",
    fill = "Estado Civil"
  ) +
  theme_minimal() +
  theme(
    axis.text = element_blank(), # Remove os números ao redor
    axis.ticks = element_blank(), # Remove os ticks dos eixos
    panel.grid = element_blank() # Remove as grades de fundo
  )

# Exibir as tabelas
contagem_sexo_idade
contagem_sexo_municipio
contagem_escolaridade_estado

# Heatmap

# Agregando os dados por faixa etária, estado civil e gênero
heatmap_data <- dataset_final %>%
  group_by(DS_GENERO, DS_FAIXA_ETARIA, DS_ESTADO_CIVIL) %>%
  summarise(Total_Eleitores = sum(QT_ELEITORES_PERFIL, na.rm = TRUE)) %>%
  ungroup()
```



```
# Filtrar dados para homens
heatmap_homens <- heatmap_data %>%
  filter(DS_GENERO == "MASCULINO")

# Criar o mapa de calor
ggplot(heatmap_homens, aes(x = DS_FAIXA_ETARIA, y = DS_ESTADO_CIVIL, fill =
  Total_Eleitores)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "lightpink", high = "darkred", name = "Total
  Eleitores") +
  labs(
    title = "Mapa de Calor: Faixa Etária vs Estado Civil (Homens)",
    x = "Faixa Etária",
    y = "Estado Civil"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )

# Filtrar dados para mulheres
heatmap_mulheres <- heatmap_data %>%
  filter(DS_GENERO == "FEMININO")

# Criar o mapa de calor
ggplot(heatmap_mulheres, aes(x = DS_FAIXA_ETARIA, y = DS_ESTADO_CIVIL, fill =
  Total_Eleitores)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "lightblue", high = "darkblue", name = "Total
  Eleitores") +
  labs(
    title = "Mapa de Calor: Faixa Etária vs Estado Civil (Mulheres)",
    x = "Faixa Etária",
    y = "Estado Civil"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
```

```
## Estatísticas por agrupamento
```

```
# Função para calcular estatísticas básicas
```

```
calcular_estatisticas <- function(data, coluna) {

  estatisticas <- data %>%
    summarise(
      Media = mean(.data[[coluna]], na.rm = TRUE),
      Mediana = median(.data[[coluna]], na.rm = TRUE),
      Moda = as.numeric(names(sort(table(.data[[coluna]]), decreasing =
        TRUE)[1])),
      Desvio_Padiao = sd(.data[[coluna]], na.rm = TRUE),
      Variancia = var(.data[[coluna]], na.rm = TRUE),
      Minimo = min(.data[[coluna]], na.rm = TRUE),
      Maximo = max(.data[[coluna]], na.rm = TRUE),
      Intervalo = Maximo - Minimo
    )

  return(estatisticas)
}

estatisticas_sexo_idade <- calcular_estatisticas(contagem_sexo_idade,
  "Total_Eleitores")
estatisticas_sexo_municipio <- calcular_estatisticas(contagem_sexo_municipio,
  "Total_Eleitores")
estatisticas_escolaridade_estado <-
  calcular_estatisticas(contagem_escolaridade_estado, "Total_Eleitores")

estatisticas_sexo_municipio
estatisticas_sexo_idade
estatisticas_escolaridade_estado
```

```
# -----
# ETAPA 4: Insights e Conclusões
# -----
```

```
# Calcular porcentagens por grupo
```

```
porcentagens <- list(
  "Por Sexo" = contagem_por_sexo %>%
```

```
mutate(Porcentagem = (Total_Eleitores / total_geral) * 100),

"Por Município" = contagem_por_municipio %>%
  mutate(Porcentagem = (Total_Eleitores / total_geral) * 100),

"Por Faixa Etária" = contagem_por_idade %>%
  mutate(Porcentagem = (Total_Eleitores / total_geral) * 100),

"Por Estado Civil" = contagem_por_estado_civil %>%
  mutate(Porcentagem = (Total_Eleitores / total_geral) * 100),

"Por Escolaridade" = contagem_por_escolaridade %>%
  mutate(Porcentagem = (Total_Eleitores / total_geral) * 100)
)

# Município com menos eleitores e porcentagem
municipio_menos <- percentagens[["Por Município"]] %>%
  arrange(Total_Eleitores) %>%
  slice(1)

# Município com mais eleitores (exceto a capital) e porcentagem
municipio_mais <- percentagens[["Por Município"]] %>%
  arrange(desc(Total_Eleitores)) %>%
  filter(NM_MUNICIPIO != "SALVADOR") %>%
  slice(1)

# Características dominantes com porcentagens
estado_civil_dominante <- percentagens[["Por Estado Civil"]] %>%
  arrange(desc(Total_Eleitores)) %>%
  slice(1)

genero_dominante <- percentagens[["Por Sexo"]] %>%
  arrange(desc(Total_Eleitores)) %>%
  slice(1)

faixa_etaria_dominante <- percentagens[["Por Faixa Etária"]] %>%
  arrange(desc(Total_Eleitores)) %>%
  slice(1)

escolaridade_dominante <- percentagens[["Por Escolaridade"]] %>%
  arrange(desc(Total_Eleitores)) %>%
  slice(1)
```

```
# Exibir Conclusões com Porcentagens
list(
  Municipio_Menos = municipio_menos,
  Municipio_Mais = municipio_mais,
  Estado_Civil_Dominante = estado_civil_dominante,
  Genero_Dominante = genero_dominante,
  Faixa_Etaria_Dominante = faixa_etaria_dominante,
  Escolaridade_Dominante = escolaridade_dominante
)
```

ANEXO B: CÓDIGO EM R PARA ANÁLISE DE DADOS METEOROLÓGICOS BÁSICOS

```
# -----  
# 1. Carregar os dados  
# -----  
  
library(lubridate)  
library(xtable)  
  
# Instalar e carregar pacotes necessários  
# install.packages("tidyverse") # Executar apenas se ainda não tiver o pacote  
library(tidyverse)  
old <- options(pillar.sigfig = 4)  
  
# Definir o caminho dos arquivos  
file_2022 <-  
  "D:/Downloads/Dataset/2022/INMET_S_PR_A807_CURITIBA_01-01-2022_A_31-12-2022.CSV"  
file_2023 <-  
  "D:/Downloads/Dataset/2023/INMET_S_PR_A807_CURITIBA_01-01-2023_A_31-12-2023.CSV"  
  
# Carregar os datasets  
data_2022 <- read_delim(file_2022,  
  delim = ";",  
  skip = 8, # pular cabeçalho de 8 linhas  
  locale = locale(encoding = "ISO-8859-1", decimal_mark =  
    ","))  
  
print(head(data_2022))  
  
data_2023 <- read_delim(file_2023,  
  delim = ";",  
  skip = 8, # pular cabeçalho de 8 linhas  
  locale = locale(encoding = "ISO-8859-1", decimal_mark =  
    ","))  
  
# Verificar dimensões dos datasets  
dim_2022 <- dim(data_2022) # Número de linhas e colunas  
dim_2023 <- dim(data_2023)  
  
# Número total de células  
num_celulas_2022 <- dim_2022[1] * dim_2022[2]
```

```

num_celulas_2023 <- dim_2023[1] * dim_2023[2]

print(dim_2022)
print(dim_2023)
print(num_celulas_2022)
print(num_celulas_2023)

# -----
# 2. Seleção de colunas
# -----

# Seleção de colunas de interesse
colunas_inmet <- c(
  "Data",
  "Hora UTC",
  "PRECIPITAÇÃO TOTAL, HORÁRIO (mm)",
  "PRESSAO ATMOSFERICA AO NIVEL DA ESTACAO, HORARIA (mB)",
  "PRESSÃO ATMOSFERICA MAX.NA HORA ANT. (AUT) (mB)",
  "PRESSÃO ATMOSFERICA MIN. NA HORA ANT. (AUT) (mB)",
  "TEMPERATURA DO AR - BULBO SECO, HORARIA (C)",
  "TEMPERATURA MÁXIMA NA HORA ANT. (AUT) (C)",
  "TEMPERATURA MÍNIMA NA HORA ANT. (AUT) (C)",
  "VENTO, DIREÇÃO HORARIA (gr) ( (gr))",
  "VENTO, RAJADA MAXIMA (m/s)",
  "VENTO, VELOCIDADE HORARIA (m/s)"
)

data_2022_selected <- data_2022 %>% select(any_of(colunas_inmet))
data_2023_selected <- data_2023 %>% select(any_of(colunas_inmet))

print(head(data_2022_selected))

# Verificar dimensões após seleção
dim_selected_2022 <- dim(data_2022_selected)
dim_selected_2023 <- dim(data_2023_selected)

print(dim_selected_2022)
print(dim_selected_2023)

# -----
# 3. Filtragem de linhas
# -----

```

```

# Remover linhas com valores nulos
data_2022_clean <- data_2022_selected %>% drop_na()
data_2023_clean <- data_2023_selected %>% drop_na()

# Verificar dimensões após limpeza
dim_clean_2022 <- dim(data_2022_clean)
dim_clean_2023 <- dim(data_2023_clean)

print(dim_clean_2022)
print(dim_clean_2023)

print(head(data_2022_clean))

# x[!x %in% boxplot.stats(x)$out]

# boxplot()
#
# box22_3 <- boxplot(data_2022_clean[3], aes(y = colnames(data_2022_clean)[3],
#   group = year, color = factor(year))) +
#   labs(
#     title = "Média de PRECIPITAÇÃO TOTAL, HORÁRIO (mm)",
#     x = "Mês",
#     y = "Média de Precipitação (mm)",
#     color = "Ano"
#   ) +
#   theme_minimal() +
#   scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))

# -----
# Boxplot PRESSAO - 2022
# -----

# Seleção das variáveis de interesse
data_boxplot <- data_2022_clean %>%
  select(
    `PRESSAO ATMOSFERICA AO NIVEL DA ESTACAO, HORARIA (mB)`,
    `PRESSÃO ATMOSFERICA MAX.NA HORA ANT. (AUT) (mB)`,
    `PRESSÃO ATMOSFERICA MIN. NA HORA ANT. (AUT) (mB)`
  )

# Transformar os dados para o formato long (necessário para ggplot)

```

```

data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +
  scale_fill_manual(values = c("blue", "orange", "gray")) +
  labs(
    title = "Pressão Atmosférica (mB) - 2022",
    subtitle = "(a) Pressão Atmosférica (mB) - 2022",
    x = NULL,
    y = "Pressão (mB)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_blank(),
    axis.text.x = element_blank()
  )

# -----
# Boxplot PRESSAO - 2023
# -----

# Seleção das variáveis de interesse
data_boxplot <- data_2023_clean %>%
  select(
    `PRESSAO ATMOSFERICA AO NIVEL DA ESTACAO, HORARIA (mB)`,
    `PRESSÃO ATMOSFERICA MAX.NA HORA ANT. (AUT) (mB)`,
    `PRESSÃO ATMOSFERICA MIN. NA HORA ANT. (AUT) (mB)`
  )

# Transformar os dados para o formato long (necessário para ggplot)
data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )

```



```

)

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +
  scale_fill_manual(values = c("blue", "orange", "gray")) +
  labs(
    title = "Pressão Atmosférica (mB) - 2023",
    subtitle = "(a) Pressão Atmosférica (mB) - 2023",
    x = NULL,
    y = "Pressão (mB)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_blank(),
    axis.text.x = element_blank()
  )
)

# -----
# Boxplot TEMPERATURA - 2022
# -----

# Seleção das variáveis de interesse
data_boxplot <- data_2022_clean %>%
  select(
    `TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`,
    `TEMPERATURA MÁXIMA NA HORA ANT. (AUT) (C)`,
    `TEMPERATURA MÍNIMA NA HORA ANT. (AUT) (C)`
  )
)

# Transformar os dados para o formato long (necessário para ggplot)
data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )
)

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +

```

```

scale_fill_manual(values = c("blue", "orange", "gray")) +
labs(
  title = "Temperatura do Ar (C) - 2022",
  subtitle = "(a) Temperatura do Ar (C) - 2022",
  x = NULL,
  y = "Temperatura (C)"
) +
theme_minimal() +
theme(
  plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
  plot.subtitle = element_blank(),
  axis.text.x = element_blank()
)

# -----
# Boxplot TEMPERATURA - 2023
# -----

# Seleção das variáveis de interesse
data_boxplot <- data_2023_clean %>%
  select(
    `TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`,
    `TEMPERATURA MÁXIMA NA HORA ANT. (AUT) (C)`,
    `TEMPERATURA MÍNIMA NA HORA ANT. (AUT) (C)`
  )

# Transformar os dados para o formato long (necessário para ggplot)
data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +
  scale_fill_manual(values = c("blue", "orange", "gray")) +
  labs(
    title = "Temperatura do Ar (C) - 2023",
    subtitle = "(a) Temperatura do Ar (C) - 2023",

```

```

    x = NULL,
    y = "Temperatura (C)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_blank(),
    axis.text.x = element_blank()

  )

# -----
# Boxplot VENTO - 2022
# -----

# Seleção das variáveis de interesse
data_boxplot <- data_2022_clean %>%
  select(
    `VENTO, RAJADA MAXIMA (m/s)`,
    `VENTO, VELOCIDADE HORARIA (m/s)`
  )

# Transformar os dados para o formato long (necessário para ggplot)
data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +
  scale_fill_manual(values = c("blue", "orange", "gray")) +
  labs(
    title = "Vento (m/s) - 2022",
    subtitle = "(a) Vento (m/s) - 2022",
    x = NULL,
    y = "Vento (m/s)"
  ) +
  theme_minimal() +
  theme(

```

```

    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_blank(),
    axis.text.x = element_blank()

)

# -----
# Boxplot VENTO - 2023
# -----

# Seleção das variáveis de interesse
data_boxplot <- data_2023_clean %>%
  select(
    `VENTO, RAJADA MAXIMA (m/s)`,
    `VENTO, VELOCIDADE HORARIA (m/s)`
  )

# Transformar os dados para o formato long (necessário para ggplot)
data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +
  scale_fill_manual(values = c("blue", "orange", "gray")) +
  labs(
    title = "Vento (m/s) - 2023",
    subtitle = "(a) Vento (m/s) - 2023",
    x = NULL,
    y = "Vento (m/s)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_blank(),
    axis.text.x = element_blank()

  )

```

```

# -----
# Boxplot VENTO graus - 2022
# -----

# Seleção das variáveis de interesse
data_boxplot <- data_2022_clean %>%
  select(
    `VENTO, DIREÇÃO HORARIA (gr) ( (gr))`
  )

# Transformar os dados para o formato long (necessário para ggplot)
data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +
  scale_fill_manual(values = c("blue", "orange", "gray")) +
  labs(
    title = "Vento ( (gr)) - 2022",
    subtitle = "(a) Vento (m/s) - 2022",
    x = NULL,
    y = "Vento ( (gr))"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_blank(),
    axis.text.x = element_blank()
  )

# -----
# Boxplot VENTO graus - 2023
# -----

# Seleção das variáveis de interesse

```

```

data_boxplot <- data_2023_clean %>%
  select(
    `VENTO, DIREÇÃO HORARIA (gr)` (gr))`
  )

# Transformar os dados para o formato long (necessário para ggplot)
data_long <- data_boxplot %>%
  pivot_longer(
    cols = everything(),
    names_to = "Variável",
    values_to = "Valor"
  )

# Criar o gráfico de boxplot
ggplot(data_long, aes(x = Variável, y = Valor, fill = Variável)) +
  geom_boxplot(outlier.color = "black", outlier.size = 1.5) +
  scale_fill_manual(values = c("blue", "orange", "gray")) +
  labs(
    title = "Vento ( gr) - 2023",
    subtitle = "(a) Vento (m/s) - 2023",
    x = NULL,
    y = "Vento ( gr)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    plot.subtitle = element_blank(),
    axis.text.x = element_blank()
  )

# -----
# 4. Análise
# -----

# Função para calcular a moda
calc_mode <- function(x) {
  uniq_x <- unique(x)
  uniq_x[which.max(tabulate(match(x, uniq_x)))]
}

summary_stats <- function(x) {

```

```

x = c(mean(x),
      calc_mode(x),
      median(x),
      var(x),
      sd(x),
      diff(range(x)),
      IQR(x),
      max(x),
      min(x))
names(x) = c("Média", "Moda", "Mediana", "Variância", "Desvio Padrão",
            "Amplitude", "IQR", "Máximo", "Mínimo")
x
}

summary_mmm <- function(x) {
  x = c(max(x),
        min(x),
        mean(x))
  names(x) = c("Máximo", "Mínimo", "Média")
  x
}

stats_22 <- data_2022_clean %>%
  select_if(is.numeric) %>%
  apply(2, summary_stats) %>%
  round(digits=1) %>%
  t

#view(stats_22)
print(xtable(stats_22, type = "latex"), file = "stats22.tex")

stats_23 <- data_2023_clean %>%
  select_if(is.numeric) %>%
  apply(2, summary_stats) %>%
  round(digits=1) %>%
  t

view(stats_23)
print(xtable(stats_23, type = "latex"), file = "stats23.tex")

# 4.1. Seleção de linhas e cálculo de estatísticas

```

```
# Adicionar colunas de ano e mês
data_2022_clean <- data_2022_clean %>%
  mutate(
    year = year(`Data`),
    month = month(`Data`, label = TRUE, abbr = TRUE) # Nome dos meses abreviados
  )

data_2023_clean <- data_2023_clean %>%
  mutate(
    year = year(`Data`),
    month = month(`Data`, label = TRUE, abbr = TRUE) # Nome dos meses abreviados
  )

# Adicionando uma coluna para o trimestre
data_2022_clean <- data_2022_clean %>%
  mutate(Trimestre = paste0("Trimestre", quarter(Data)))

data_2023_clean <- data_2023_clean %>%
  mutate(Trimestre = paste0("Trimestre", quarter(Data)))

# Calculando os máximos, mínimos e médias por trimestre
resumo_2022_trimestral <- data_2022_clean %>%
  group_by(Trimestre) %>%
  summarise(
    Máximo = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Mínimo = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Média = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE)
  )

resumo_2022_total <- data_2022_clean %>%
  select(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)%>%
  summarise(
    Máximo = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Mínimo = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Média = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE)
  ) %>%
  mutate(Trimestre = "Total")

resumo_2022_trim_temp <- bind_rows(resumo_2022_trimestral, resumo_2022_total)

view(resumo_2022_trim_temp)
print(xtable(resumo_2022_trim_temp, type = "latex"), file = "trim22.tex")
```



```

resumo_2023_trimestral <- data_2023_clean %>%
  group_by(Trimestre) %>%
  summarise(
    Máximo = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Mínimo = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Média = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE)
  )

resumo_2023_total <- data_2023_clean %>%
  select(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)%>%
  summarise(
    Máximo = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Mínimo = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    Média = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE)
  ) %>%
  mutate(Trimestre = "Total")

resumo_2023_trim_temp <- bind_rows(resumo_2023_trimestral, resumo_2023_total)

view(resumo_2023_trim_temp)
print(xtable(resumo_2023_trim_temp, type = "latex"), file = "trim23.tex")

# Combinar os dois datasets
combined_data <- bind_rows(data_2022_clean, data_2023_clean)

# -----
# Calcular a média mensal de PRECIPITAÇÃO
# -----
monthly_precipitation <- combined_data %>%
  group_by(year, month) %>%
  summarise(
    mean_precipitation = mean(`PRECIPITAÇÃO TOTAL, HORÁRIO (mm)`, na.rm = TRUE)
  )

# Criar o gráfico com ggplot2
dev.new()

precip_mensal <- ggplot(monthly_precipitation, aes(x = month, y =
  mean_precipitation, group = year, color = factor(year))) +
  geom_line(size = 1) +
  geom_point(size = 2) +

```

```

labs(
  title = "Média de PRECIPITAÇÃO TOTAL, HORÁRIO (mm)",
  x = "Mês",
  y = "Média de Precipitação (mm)",
  color = "Ano"
) +
theme_minimal() +
scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))

precip_mensal %>% print

# -----
# Calcular a média horária de PRECIPITAÇÃO
# -----
hourly_precipitation <- combined_data %>%
  group_by(year, `Hora UTC`) %>%
  summarise(
    mean_precipitation = mean(`PRECIPITAÇÃO TOTAL, HORÁRIO (mm)`, na.rm = TRUE)
  )

# Criar o gráfico com ggplot2
dev.new()

precip_hora <- ggplot(hourly_precipitation, aes(x = `Hora UTC`, y =
  mean_precipitation, group = year, color = factor(year))) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(
    title = "Média de PRECIPITAÇÃO TOTAL, HORÁRIO (mm)",
    x = "Hora",
    y = "Média de Precipitação (mm)",
    color = "Ano"
  ) +
  theme_minimal() +
  scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))+
  theme(axis.text.x = element_text(angle = 90))

precip_hora %>% print

# -----
# Calcular a média mensal de PRESSÃO
# -----

```

```

monthly_pressure <- combined_data %>%
  group_by(year, month) %>%
  summarise(
    mean_pressure = mean(`PRESSAO ATMOSFERICA AO NIVEL DA ESTACAO, HORARIA
      (mB)`, na.rm = TRUE)
  )

# Criar o gráfico da média mensal
dev.new()

pressao_mensal <- ggplot(monthly_pressure, aes(x = month, y = mean_pressure,
  group = year, color = factor(year))) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(
    title = "Média de PRESSÃO ATMOSFÉRICA AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)",
    x = "Mês",
    y = "Média de Pressão (mB)",
    color = "Ano"
  ) +
  theme_minimal() +
  scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))

pressao_mensal %>% print

# -----
# Calcular a média horária de PRESSÃO
# -----
hourly_pressure <- combined_data %>%
  group_by(year, `Hora UTC`) %>%
  summarise(
    mean_pressure = mean(`PRESSAO ATMOSFERICA AO NIVEL DA ESTACAO, HORARIA
      (mB)`, na.rm = TRUE)
  )

# Criar o gráfico da média horária
dev.new()

pressao_hora <- ggplot(hourly_pressure, aes(x = `Hora UTC`, y = mean_pressure,
  group = year, color = factor(year))) +
  geom_line(size = 1) +
  geom_point(size = 2) +

```

```

labs(
  title = "Média de PRESSÃO ATMOSFÉRICA AO NÍVEL DA ESTAÇÃO, HORÁRIA (mB)",
  x = "Hora",
  y = "Média de Pressão (mB)",
  color = "Ano"
) +
theme_minimal() +
scale_color_manual(values = c("2022" = "blue", "2023" = "orange")) +
theme(axis.text.x = element_text(angle = 90))

pressao_hora %>% print

# -----
# Calcular a média mensal de TEMPERATURA
# -----
monthly_temp <- combined_data %>%
  group_by(year, month) %>%
  summarise(
    mean_temp = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm =
      TRUE)
  )

# Criar o gráfico com ggplot2
dev.new()

temp_mensal <- ggplot(monthly_temp, aes(x = month, y = mean_temp, group = year,
  color = factor(year))) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(
    title = "Média de TEMPERATURA DO AR - BULBO SECO, HORARIA (C)",
    x = "Mês",
    y = "Média de Precipitação (mm)",
    color = "Ano"
  ) +
  theme_minimal() +
  scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))

temp_mensal %>% print

# -----
# Calcular a média horária de TEMPERATURA

```

```

# -----
hourly_temp <- combined_data %>%
  group_by(year, `Hora UTC`) %>%
  summarise(
    mean_temp = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm =
      TRUE)
  )

# Criar o gráfico com ggplot2
dev.new()

temp_hora <- ggplot(hourly_temp, aes(x = `Hora UTC`, y = mean_temp, group =
  year, color = factor(year))) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  labs(
    title = "Média de TEMPERATURA DO AR - BULBO SECO, HORARIA (C)",
    x = "Hora",
    y = "Média de Precipitação (mm)",
    color = "Ano"
  ) +
  theme_minimal() +
  scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))+
  theme(axis.text.x = element_text(angle = 90))

temp_hora %>% print

# -----
# Calcular a média mensal de TEMPERATURA
# -----
monthly_vento <- combined_data %>%
  group_by(year, month) %>%
  summarise(
    mean_vento = mean(`VENTO, RAJADA MAXIMA (m/s)`, na.rm = TRUE)
  )

# Criar o gráfico com ggplot2
dev.new()

vento_mensal <- ggplot(monthly_vento, aes(x = month, y = mean_vento, group =
  year, color = factor(year))) +
  geom_line(size = 1) +

```

```

geom_point(size = 2) +
labs(
  title = "Média de VENTO, RAJADA MAXIMA (m/s)",
  x = "Mês",
  y = "Média de Precipitação (mm)",
  color = "Ano"
) +
theme_minimal() +
scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))

vento_mensal %>% print

# -----
# Calcular a média, máxima e mínima de temperatura para cada hora e ano
hourly_temperature <- combined_data %>%
  group_by(year, `Hora UTC`) %>% # Agrupar por ano e Hora UTC
  summarise(
    mean_temp = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm =
      TRUE),
    max_temp = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE),
    min_temp = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`, na.rm = TRUE)
  )

# Transformar os dados para o formato long (para facilitar o gráfico)
hourly_temperature_long <- hourly_temperature %>%
  pivot_longer(
    cols = c(mean_temp, max_temp, min_temp),
    names_to = "Metric",
    values_to = "Temperature"
  )

# Criar o gráfico de temperaturas horárias
ggplot(hourly_temperature_long, aes(x = `Hora UTC`, y = Temperature, group =
  interaction(year, Metric), color = year)) +
  geom_line(aes(linetype = Metric), size = 1) + # Diferenciar por tipo de linha
  (métrica)
labs(
  title = "Temperatura Máxima, Média e Mínima ao Longo do Dia",
  x = "Hora UTC",
  y = "Temperatura (C)",
  color = "Ano",
  linetype = "Métrica"
)

```

```

) +
theme_minimal()+
theme(axis.text.x = element_text(angle = 90)) #+
#scale_color_manual(values = c("2022" = "blue", "2023" = "orange"))

# 4.2. Estatísticas básicas de variáveis quantitativas

# Estatísticas básicas para 2022
# summary_2022 <- data_2022_clean %>%
#   summarise(
#     mean_temp = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     median_temp = median(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     sd_temp = sd(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     max_temp = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     min_temp = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)
#   )
#
# summary_2023 <- data_2023_clean %>%
#   summarise(
#     mean_temp = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     median_temp = median(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     sd_temp = sd(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     max_temp = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     min_temp = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)
#   )
#
# print(summary_2022)
# print(summary_2023)

# 4.3. Agrupamento temporal e cálculo de estatísticas

# Agrupar por mês e calcular estatísticas básicas de temperatura
# monthly_stats_2022 <- data_2022_clean %>%
#   group_by(month = lubridate::month(`Data`, label = TRUE)) %>%
#   summarise(
#     mean_temp = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     max_temp = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     min_temp = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)
#   )
#
# monthly_stats_2023 <- data_2023_clean %>%

```

```
# group_by(month = lubridate::month(`Data`, label = TRUE)) %>%
# summarise(
#   mean_temp = mean(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#   max_temp = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#   min_temp = min(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)
# )
#
# print(monthly_stats_2022)
# print(monthly_stats_2023)

# 4.4. Comparação entre anos

# Comparar as médias anuais de temperatura
# annual_stats <- data.frame(
#   year = c(2022, 2023),
#   mean_temp = c(
#     mean(data_2022_clean$`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`),
#     mean(data_2023_clean$`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)
#   )
# )
#
# print(annual_stats)

# # 4.5. Gráficos com ggplot2
#
# library(ggplot2)
#
# # Combinar dados de ambos os anos
# monthly_combined <- bind_rows(
#   mutate(monthly_stats_2022, year = 2022),
#   mutate(monthly_stats_2023, year = 2023)
# )
#
# dev.new()
#
# # Gráfico de linha
# monthly_combined %>% ggplot( aes(x = month, y = mean_temp, color =
#   factor(year))) +
#   geom_line(size = 10) +
#   geom_point() +
#   geom_line() +
#   labs(
```



```

#   title = "Temperatura Média Mensal - 2022 vs 2023",
#   x = "Mês",
#   y = "Temperatura Média (C)",
#   color = "Ano"
# ) +
#   theme_minimal()

# # 4.6. Mapas de calor
#
# # Adicionar colunas auxiliares para agrupamento
# data_2022_clean <- data_2022_clean %>%
#   mutate(
#     month = lubridate::month(`Data`, label = TRUE),
#     hour = lubridate::hour(`Data`)
#   )
#
# dev.new()
#
# # Criar mapa de calor
# heatmap_temp <- data_2022_clean %>%
#   group_by(month, hour) %>%
#   summarise(max_temp = max(`TEMPERATURA DO AR - BULBO SECO, HORARIA (C)`)
# )
#
# heatmap_temp %>% ggplot(aes(x = hour, y = month, fill = max_temp)) +
#   geom_tile() +
#   labs(
#     title = "Mapa de Calor: Temperatura Máxima (2022)",
#     x = "Hora",
#     y = "Mês",
#     fill = "Temp Máx (C)"
#   ) +
#   theme_minimal()

#print(heatmap_temp)

# -----
# Descartar
# -----

# my_skim <- skim_with(numeric =
#   sfl(mean,calc_mode,median,var,sd,range,IQR,max,min), append = FALSE)

```

```
# print(my_skim(data_2022_clean))
```
