

**FACULDADE SENAC PERNAMBUCO**  
**CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE**  
**SISTEMAS**

**Caio Rodrigues**

**Gabriel Eugênio**

**Hesdras Francisco**

**Nicole Araujo**

**RELATÓRIO**  
**Opioid Overdose Deaths**

Recife,  
2024

## SUMÁRIO

<b>Introdução</b>	<b>3</b>
<b>Descrição do Dataset</b>	<b>3</b>
<b>Requisitos</b>	<b>5</b>
<b>Pacotes Necessários</b>	<b>5</b>
<b>Comando de Instalação</b>	<b>5</b>
<b>Análise Exploratória dos Dados</b>	<b>6</b>
• Estrutura do Dataset	6
• Remoção das linhas com valores ausentes ou suprimidos	6
• Análise Temporal das Mortes por Overdose de Opioides	6
• Análise de Correlação	7
◦ Principais Correlações Identificadas	7
• Visualização Interativa com Dash	8
• Dashboard	10
<b>Conclusão</b>	<b>12</b>

## Introdução

A dependência de opioides e as taxas de mortalidade relacionadas a esses medicamentos atingiram níveis epidêmicos nos Estados Unidos e em outros países. Com o aumento significativo dos casos de overdose envolvendo opioides, é essencial realizar novas pesquisas para entender melhor essa crise de saúde pública. Este relatório apresenta uma análise do conjunto de dados "Opioid Overdose Deaths", que inclui informações detalhadas sobre mortes por overdose, populações, taxas de mortalidade e prescrições de opioides entre os anos de 1999 e 2014. O dataset foi coletado a partir de fontes confiáveis, incluindo o CDC, e não contém valores nulos ou zerados. Ele fornece um panorama abrangente da epidemia de opioides nos EUA.

## Descrição do Dataset

O dataset contém as seguintes variáveis:

### State (Estado)

- Valores únicos: 51
- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)
- Mais Comum: Alabama (2%)

### Year (Ano)

- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)
- Média: 2001
- Desvio Padrão: 4.61
- Quartis: Min: 1999, 25%: 2003, 50%: 2007, 75%: 2011, Max: 2014

### Deaths (Mortes)

- Valores únicos: 497
- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)
- Mais Comum: Suprimido (2%)

### **Population (População)**

- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)
- Média: 5.87 milhões
- Desvio Padrão: 6.56 milhões
- Quartis: Min: 492 mil, 25%: 1.58 milhões, 50%: 4.12 milhões, 75%: 6.60 milhões, Max: 38.8 milhões

### **Crude Rate (Taxa Bruta)**

- Valores únicos: 164
- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)
- Mais Comum: Não Confiável (4%)

### **Crude Rate Lower 95% Confidence Interval (Intervalo de Confiança Inferior de 95% para a Taxa Bruta)**

- Valores únicos: 151
- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)
- Mais Comum: 4.5 (2%)

### **Crude Rate Upper 95% Confidence Interval (Intervalo de Confiança Superior de 95% para a Taxa Bruta)**

- Valores únicos: 172
- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)
- Mais Comum: 4.2 (2%)

### **Prescriptions Dispensed by US Retailers in that year (millions) (Prescrições Dispensadas por Varejistas nos EUA naquele ano, em milhões)**

- Válidos: 816 (100%)
- Incompatíveis: 0 (0%)
- Faltantes: 0 (0%)

## Requisitos

Para garantir que todos os scripts e notebooks funcionem corretamente, é necessário instalar os seguintes pacotes Python. Abaixo está a lista de pacotes necessários e o comando para instalá-los.

## Pacotes Necessários

**matplotlib:** Biblioteca para criação de gráficos estáticos, animados e interativos.

**seaborn:** Biblioteca baseada em matplotlib que fornece uma interface de alto nível para desenhar gráficos estatísticos atraentes e informativos.

**scikit-learn:** Biblioteca para aprendizado de máquina que fornece ferramentas simples e eficientes para análise de dados e mineração de dados.

**dash:** Biblioteca para construção de dashboards web interativos com Python.

**pandas:** Biblioteca para manipulação e análise de dados, oferecendo estruturas de dados e operações para manipular tabelas numéricas e séries temporais.

**numpy:** Biblioteca fundamental para computação científica em Python, fornecendo suporte para arrays e matrizes grandes multidimensionais, junto com uma grande coleção de funções matemáticas.

## Comando de Instalação

Para instalar todas essas bibliotecas, execute o seguinte comando no terminal:

```
pip install matplotlib seaborn scikit-learn dash pandas numpy
```

## Análise Exploratória dos Dados

- Estrutura do Dataset

```
# Lendo a base de dados utilizada.
data = pd.read_csv('Multiple Cause of Death 1999-2014 v1.1.csv',
sep=',')

# Visualizar dataframe
data.head()

# Informações das colunas
data.info()

# Total de registros
data.count()

# Total de nulos
data.isnull().sum()

# Total de NaN
data.isna().sum()

# Total de dados duplicatos
data.duplicated().sum()
```

- Remoção das linhas com valores ausentes ou suprimidos

```
# Remover linhas com valores ausentes ou suprimidos
drop_rows = data[ (data['Deaths'] == 'Suppressed') | (data['Deaths']
== 'Unreliable') | (data['Crude Rate'] == 'Suppressed') |
(data['Crude Rate'] == 'Unreliable') ].index
data.drop(drop_rows , inplace=True)
```

- Análise Temporal das Mortes por Overdose de Opioides

```
pivot_data = data.pivot_table(index='Year', columns='State',
values='Deaths', aggfunc='sum')

plt.figure(figsize=(14, 8))
pivot_data.plot(kind='bar', stacked=True, figsize=(14, 8))
plt.title('Total de Mortes por Estado e Ano')
plt.xlabel('Ano')
plt.ylabel('Total de Mortes')
plt.xticks(rotation=45)
plt.legend(loc='upper left', bbox_to_anchor=(1, 1))
plt.tight_layout()
```

- **Análise de Correlação**

```
# Selecionar as colunas relevantes para a correlação
correlation_data = data[['Deaths', 'Population', 'Crude Rate',
'Prescriptions Dispensed by US Retailers in that year (millions)']]

# Calcular a matriz de correlação
correlation_matrix = correlation_data.corr()

# Plotar o mapa de calor
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1,
vmax=1, linewidths=0.5, linecolor='black')
plt.title('Mapa de Calor das Correlações entre Variáveis')
plt.show()
```

- **Principais Correlações Identificadas**

- **Deaths & Population:** 0.84 (correlação forte)
  - O número de mortes tem uma forte relação com a população.
- **Deaths & Crude Rate:** 0.24 (correlação fraca)
  - O número de mortes tem uma relação fraca com a taxa bruta.
- **Deaths & Prescriptions Dispensed by US Retailers in that year (millions):** 0.27 (correlação fraca)
  - O número de mortes tem uma relação fraca com a quantidade de prescrições dispensadas por varejistas nos EUA naquele ano.
- **Population & Crude Rate:** -0.17 (correlação fraca negativa)
  - A população tem uma relação fraca e negativa com a taxa bruta.
- **Population & Prescriptions Dispensed by US Retailers in that year (millions):** 0.005 (correlação muito fraca)
  - A população tem uma relação muito fraca com a quantidade de prescrições dispensadas por varejistas nos EUA naquele ano.
- **Crude Rate & Prescriptions Dispensed by US Retailers in that year (millions):** 0.44 (correlação moderada)
  - A taxa bruta tem uma relação moderada com a quantidade de prescrições dispensadas por varejistas nos EUA naquele ano.

- Visualização Interativa com Dash

```
import dash
from dash import dcc, html
import pandas as pd
import plotly.express as px
import plotly.graph_objects as go

# Converter colunas relevantes para numérico
data[['Deaths', 'Population', 'Crude Rate', 'Prescriptions Dispensed
by US Retailers in that year (millions)']] = data[['Deaths',
'Population', 'Crude Rate', 'Prescriptions Dispensed by US Retailers
in that year (millions)']].apply(pd.to_numeric, errors='coerce')

# Filtrar os dados para o ano de 2014
data_2014 = data[data['Year'] == 2014]

# Calcular a matriz de correlação
correlation_data = data[['Deaths', 'Population', 'Crude Rate',
'Prescriptions Dispensed by US Retailers in that year (millions)']]
correlation_matrix = correlation_data.corr()

# Calcular o DataFrame agrupado
avg_prescrip = data.groupby('Year')[['Deaths', 'Prescriptions
Dispensed by US Retailers in that year
(millions)']].sum().sort_values(by='Year', ascending=False)

# Criar o gráfico de linha com Plotly
line_fig = go.Figure()

line_fig.add_trace(go.Scatter(
    x=avg_prescrip.index,
    y=avg_prescrip['Deaths'],
    mode='lines',
    name='Deaths',
    line=dict(width=5)
))

line_fig.add_trace(go.Scatter(
    x=avg_prescrip.index,
    y=avg_prescrip['Prescriptions Dispensed by US Retailers in that
year (millions)'],
    mode='lines',
    name='Prescriptions Dispensed by US Retailers in that year
(millions)',
    line=dict(width=5)
))
```



```

line_fig.update_layout(
    title='Número de Mortes Relacionadas a Opioides e Prescrição
(Milhões) ao Longo de 15 Anos',
    xaxis_title='Ano de 1994 a 2014',
    yaxis_title='Número de Mortes por Opioides e Prescrição de
Opioides',
    xaxis=dict(tickangle=75),
    width=800,
    height=800
)

# Inicializar a aplicação Dash
app = dash.Dash(__name__)

app.layout = html.Div([
    html.H1('Dashboard de Análise de Overdose por Opioides'),

    dcc.Tabs([
        dcc.Tab(label='Gráfico de Linha', children=[
            dcc.Graph(
                id='line-plot',
                figure=line_fig
            )
        ]),

        dcc.Tab(label='Gráfico de Dispersão', children=[
            dcc.Graph(
                id='scatter-plot',
                figure=px.scatter(data_2014, x='Prescriptions
Dispensed by US Retailers in that year (millions)', y='Deaths',
                                size='Population', color='State',
                                hover_name='State', log_x=True,
                                size_max=60,
                                title='Prescrições vs Mortes
(2014)')
            )
        ]),

        dcc.Tab(label='Gráfico de Barras', children=[
            dcc.Graph(
                id='bar-plot',
                figure=px.bar(data_2014, x='State', y='Deaths',
                              title='Mortes por Estado (2014)')
            )
        ]),
    ]),

```

```

dcc.Tab(label='Mapa de Calor das Correlações', children=[
    dcc.Graph(
        id='heatmap',
        figure=px.imshow(correlation_matrix,
                        labels=dict(x="Variáveis",
                                y="Variáveis", color="Correlação"),
                        x=correlation_matrix.columns,
                        y=correlation_matrix.columns,
                        title='Mapa de Calor das Correlações
entre Variáveis')
    ),
],
)
)

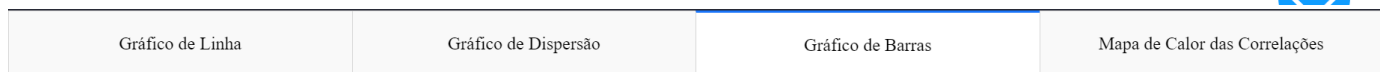
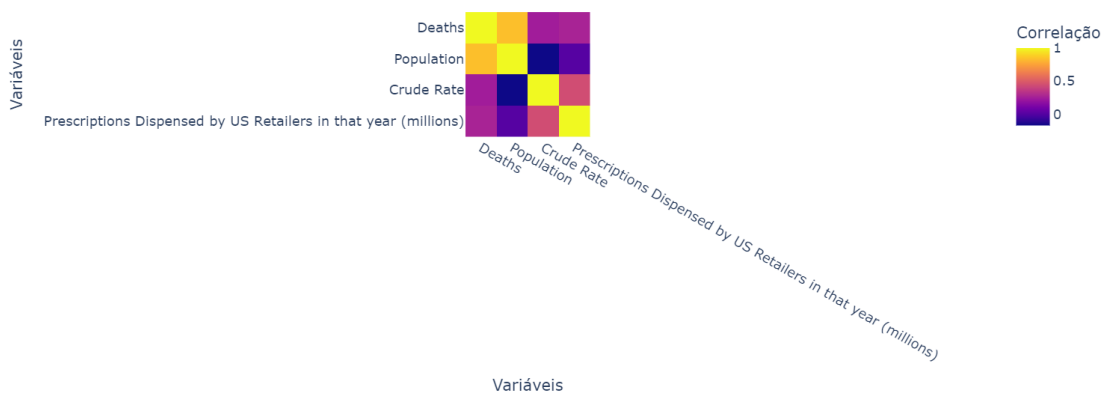
if __name__ == '__main__':
    app.run_server(debug=True)

```

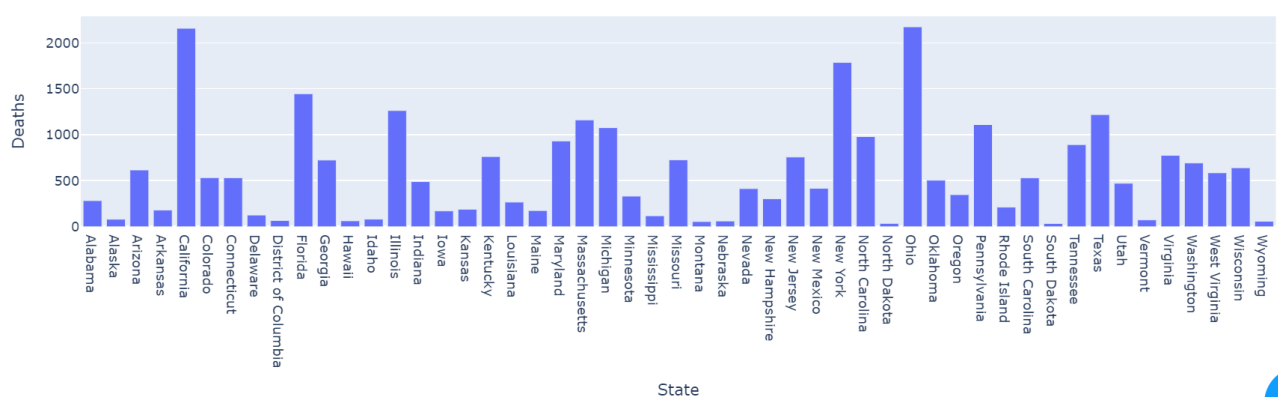
## • Dashboard



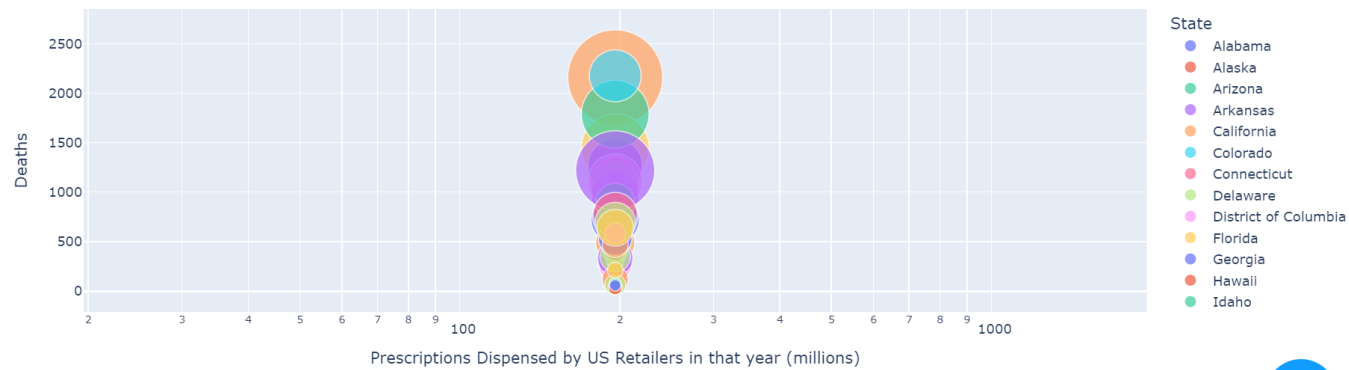
Mapa de Calor das Correlações entre Variáveis



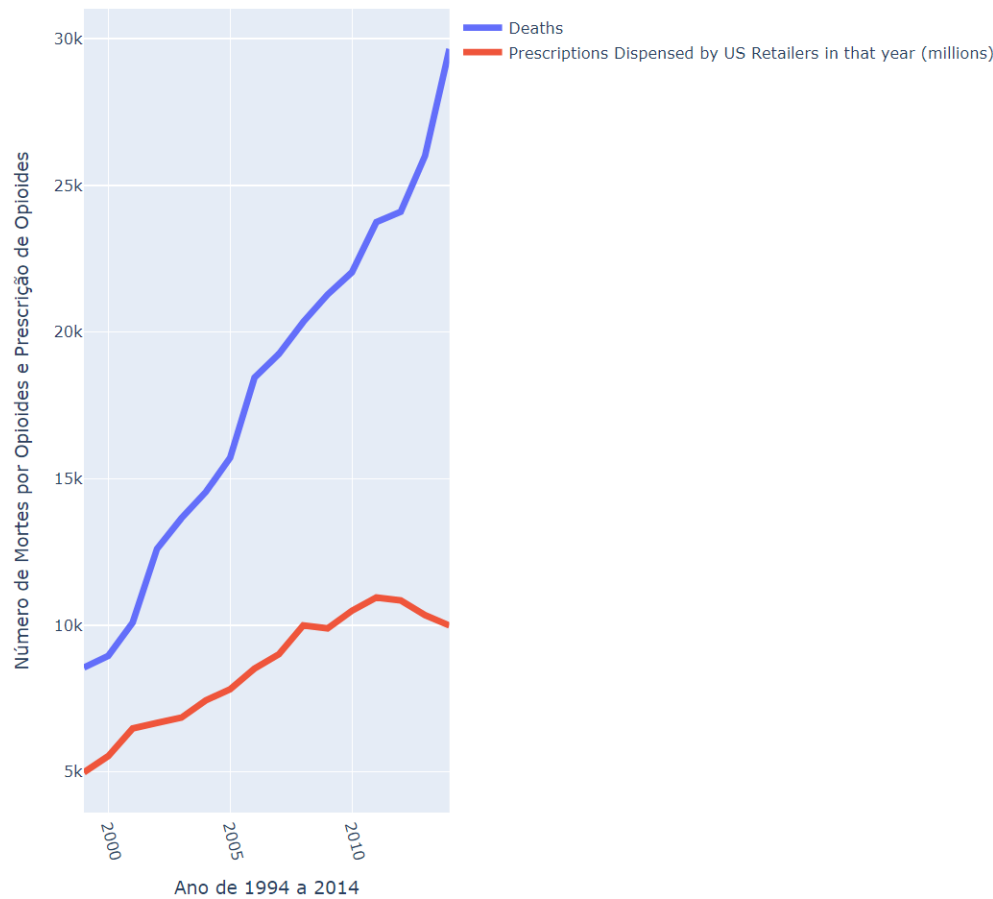
Mortes por Estado (2014)



Prescrições vs Mortes (2014)



Número de Mortes Relacionadas a Opioides e Prescrição (Milhões) ao Longo de 15 Anos



## **Conclusão**

A análise exploratória do conjunto de dados de mortes por overdose de opioides revelou importantes relações entre variáveis como mortes, população, taxa bruta e prescrições dispensadas por varejistas nos Estados Unidos. A visualização interativa fornecida pelo Dash oferece uma maneira eficiente de explorar essas correlações, permitindo uma melhor compreensão dos fatores que influenciam as mortes por overdose de opioides. Com base nesses insights, futuras intervenções podem ser direcionadas de forma mais eficaz para combater a crise de opioides e promover a saúde pública.