

Avaliação 3 - Gráficos de indicadores

Raissa Bastos

UNIVERSIDADE FEDERAL DO ABC
Física Computacional

27 de junho de 2025

1 Tarefa 0

2 Tarefa 1

3 Tarefa 2

4 Tarefa 3

5 Tarefa 4

Organização inicial

```
import matplotlib.pyplot as plt
import numpy as np

# Carregar os dados dos três arquivos
eletricidade = pd.read_csv("eletricidade.csv")
combustiveis = pd.read_csv("combustiveis_limpos.csv")
agua = pd.read_csv("agua_potavel.csv")

# Filtrar apenas El Salvador
pais = "El Salvador"
eletricidade_sv = eletricidade[eletricidade["Entity"] == pais]
combustiveis_sv = combustiveis[combustiveis["Entity"] == pais]
agua_sv = agua[agua["Entity"] == pais]

# Renomear colunas para facilitar
eletricidade_sv = eletricidade_sv.rename(columns={"Access to electricity (% of population)": "Eletricidade (%)"})
combustiveis_sv = combustiveis_sv.rename(columns={
    "Proportion of population with primary reliance on clean fuels and technologies for cooking (%) - Residence area type: Total": "Combustíveis"
})
agua_sv = agua_sv.rename(columns={
    "Share of the population using at least basic drinking water services": "Água potável (%)"
})

# Unir os três DataFrames com base em 'Entity', 'Code' e 'Year'
df = eletricidade_sv.merge(combustiveis_sv, on=["Entity", "Code", "Year"])
df = df.merge(agua_sv, on=["Entity", "Code", "Year"])

# Mostrar os primeiros dados organizados
print(df)
```

Organizando os dataframes

Tarefa 1 - gráfico com linha temporal para os três indicadores

```
plt.figure(figsize=(12, 6))
plt.plot(df["Year"], df["Eletricidade (%)"], label="Eletricidade (%)")
plt.plot(df["Year"], df["Combustíveis limpos (%)"], label="Combustíveis limpos (%)")
plt.plot(df["Year"], df["Água potável (%)"], label="Água potável (%)")

plt.xlabel("Ano")
plt.ylabel("Porcentagem da população (%)")
plt.title("Indicadores de acesso ☐ El Salvador")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Plotando o Gráfico

Tarefa 1 - gráfico com linha temporal para os três indicadores

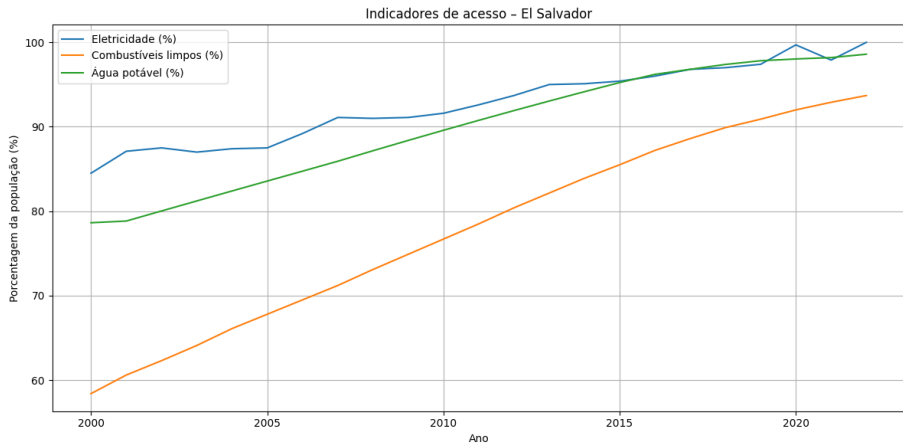


Gráfico resultante

Tarefa 2 - gráfico de dispersão com regressão linear para cada indicador

```
indicadores = [
    ("Eletricidade (%)", "Acesso à Eletricidade"),
    ("Combustíveis limpos (%)", "Acesso à Combustíveis Limpos"),
    ("Água potável (%)", "Acesso à Água Potável")
]

plt.figure(figsize=(18, 5))
for i, (coluna, titulo) in enumerate(indicadores, 1):
    plt.subplot(1, 3, i)

    # Dados x (ano) e y (indicador)
    x = df["Year"].values
    y = df[coluna].values

    # Regressão linear: coeficientes da reta (y = ax + b)
    coef = np.polyfit(x, y, 1)
    polinomio = np.poly1d(coef)

    # Gerar linha de tendência
    x_tend = np.linspace(x.min(), x.max(), 100)
    y_tend = polinomio(x_tend)

    # Plotar pontos e linha
    plt.scatter(x, y, color="blue", s=40, label="Dados reais")
    plt.plot(x_tend, y_tend, color="red", linewidth=2, label="Tendência linear")

    plt.title(titulo)
    plt.xlabel("Ano")
    plt.ylabel("% da População")
    plt.grid(True)
    plt.legend()

plt.suptitle("Tendência Temporal dos Indicadores - El Salvador", fontsize=16, y=1.05)
plt.tight_layout()
plt.show()
```

Plotando o Gráfico

Tarefa 2 - gráfico de dispersão com regressão linear para cada indicador

Tendência Temporal dos Indicadores - El Salvador

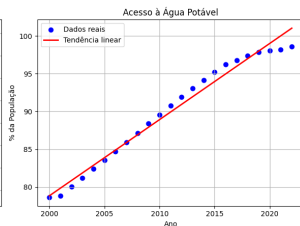
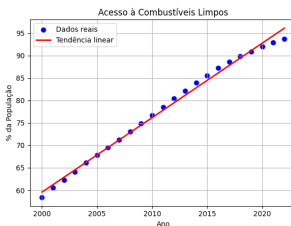
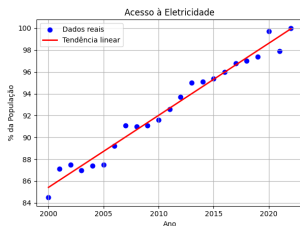


Gráfico Resultante

Tarefa 3 - heatmap dos indicadores a cada 5 anos

```
# Criar coluna de década - mudei para 5 anos para melhor visualização
df["Década"] = (df["Year"] // 5) * 5

# Agrupar por 5 anos e calcular a média dos indicadores
decadal = df.groupby("Década")[["Eletricidade (%)", "Combustíveis limpos (%)", "Água potável (%)"]].mean().round(1) #deixe o nome "decada" pela simplicidade

# Converter para matriz
dados = decadal.values
decadas = decadal.index.astype(int).tolist()
indicadores = decadal.columns.tolist()

# Plotar heatmap com matplotlib
plt.figure(figsize=(10, 6))
im = plt.imshow(dados, cmap="YlGnBu")

# Adicionar os valores nas células
for i in range(dados.shape[0]):
    for j in range(dados.shape[1]):
        texto = f"{dados[i, j]:.1f}" if not np.isnan(dados[i, j]) else ""
        plt.text(j, i, texto, ha='center', va='center', color='black')

# Configurar eixos
plt.xticks(ticks=np.arange(len(indicadores)), labels=indicadores, rotation=15)
plt.yticks(ticks=np.arange(len(decadas)), labels=decadas)
plt.xlabel("Indicadores")
plt.ylabel("Grupo de 5 anos")
plt.title("Média dos Indicadores a cada 5 anos - El Salvador")
plt.colorbar(im, label="% da População")
plt.tight_layout()
plt.show()
```

Plotando o Gráfico

Tarefa 3 - heatmap dos indicadores a cada 5 anos

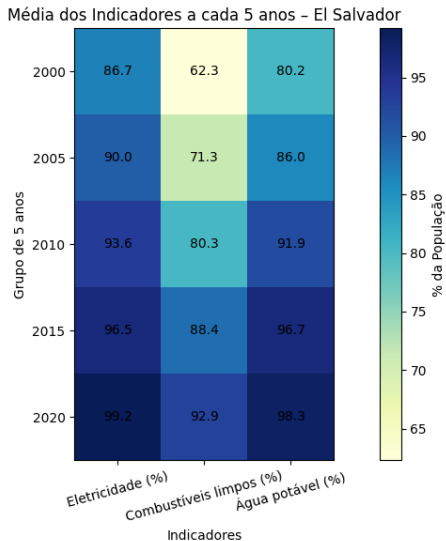


Gráfico Resultante

Tarefa 4 - gráfico de barras comparando os 3 indicadores em um único ano

```
ano = 2015
dados_ano = df[df["Year"] == ano][["Eletricidade (%)", "Combustíveis limpos (%)", "Água potável (%)"]].T
dados_ano.columns = [str(ano)]

# Plotar gráfico de barras
plt.figure(figsize=(6, 5))
dados_ano.plot(kind="bar", legend=False, color=[ "#2a9d8f"])
plt.title(f"Indicadores em El Salvador {ano}")
plt.ylabel("Porcentagem da população (%)")
plt.xticks(rotation=45)
plt.grid(axis='y')
plt.tight_layout()
plt.show()
```

Plotando o Gráfico

Tarefa 4 - gráfico de barras comparando os 3 indicadores em um único ano

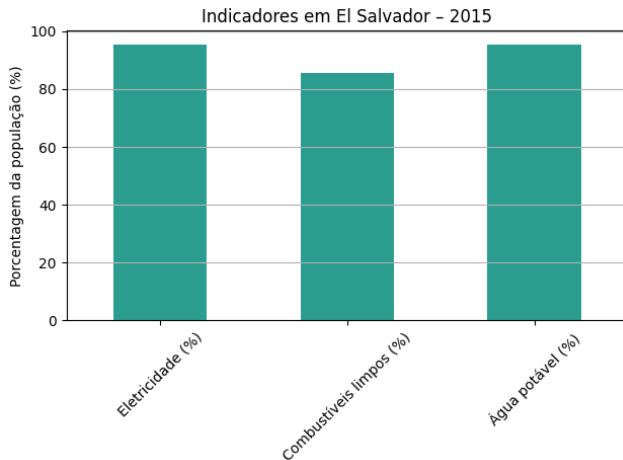


Gráfico Resultante