desafio 2 python

Sophia ra277230

2025-08-21

```
#!pip install pandas
#!pip install numpy
#!pip install matplotlib
import pandas as pd  # Biblioteca para manipulação de dados
import numpy as np   # Biblioteca para operações numéricas
import calendar      # Biblioteca para operações com datas e calendários
import matplotlib.pyplot as plt  # Biblioteca para criação de gráficos
from matplotlib.colors import LinearSegmentedColormap  # Para criar gradiente de cor personalizado

#Função para processar chunks (equivale ao getStats no R)
def process_chunk(chunk):
    # Filtra os dados para manter apenas as companhias AA, DL, UA, US e remove valores nulos em 'AIRLIN
    chunk = chunk[chunk['AIRLINE'].isin(['AA', 'DL', 'UA', 'US']) & chunk['AIRLINE'].notna()]

    #Cria uma nova coluna indicando se o voo teve atraso superior a 10 minutos (1 = sim, 0 = não)
    chunk['voos_atrasados'] = (chunk['ARRIVAL_DELAY'].astype(float) > 10).astype(int)

    #Agrupa os dados por DIA, MES e AIRLINE
    #Soma os voos atrasados e conta o total de voos
    grouped = chunk.groupby(['DAY', 'MONTH', 'AIRLINE']).agg(
        voos_atrasados=('voos_atrasados', 'sum'),
        total_voos=('ARRIVAL_DELAY', 'count')
    ).reset_index()

    return grouped  #Retorna o DataFrame processado

#Lendo o CSV em chunks e processando (leitura do banco de dadis flights)
file_path = r"C:\Users\Shophia\Documents\me315\flights.csv.zip"  # Caminho para o arquivo ZIP contendo

chunks = []           #Lista que armazenará os chunks processados
chunk_size = 100_000  #Define o tamanho de cada chunk (100 mil linhas)

#Lê o arquivo CSV em partes (chunks)
for chunk in pd.read_csv(file_path,
                         usecols=['AIRLINE', 'ARRIVAL_DELAY', 'DAY', 'MONTH'],  #Colunas de interesse
                         dtype={'AIRLINE': str, 'ARRIVAL_DELAY': str, 'DAY': int, 'MONTH': int},  # Tip
                         chunksize=chunk_size):
    processed = process_chunk(chunk)  # Processa o chunk
    chunks.append(processed)          # Adiciona o resultado à lista
```

```
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
```

```
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
```

```
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
```

```
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
```

```
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
```

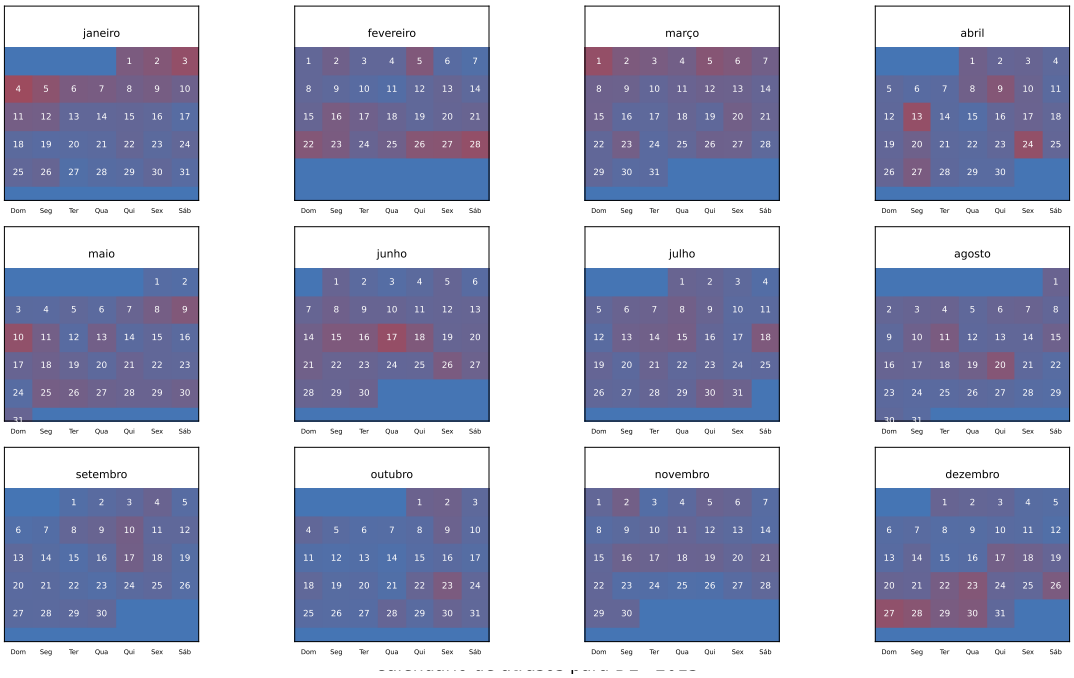
```
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
```

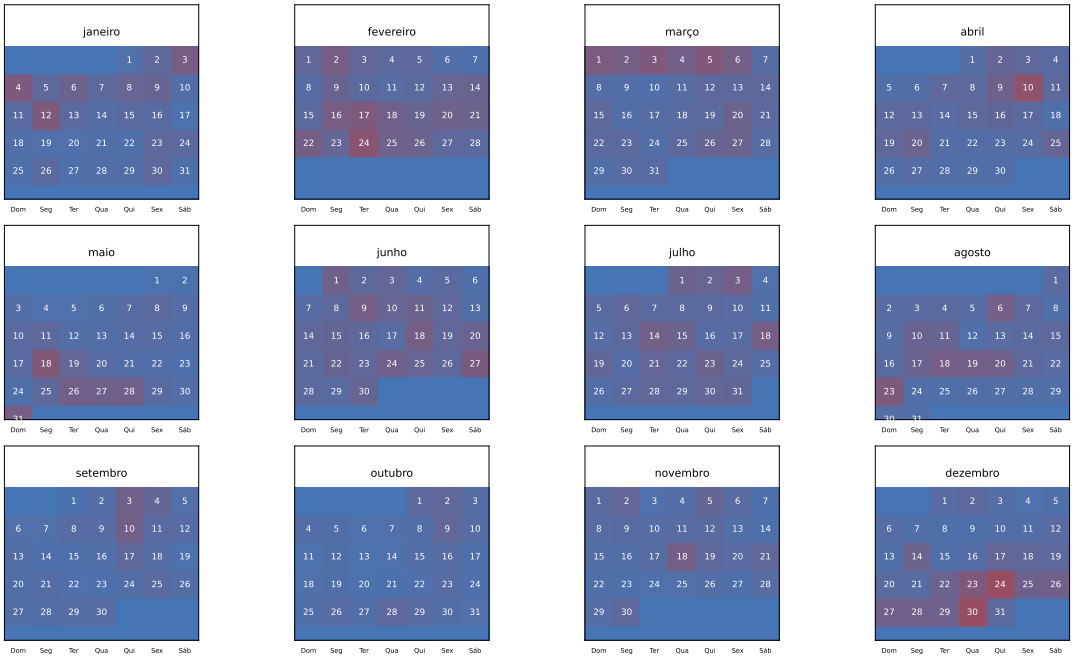
```
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
## <string>:8: SettingWithCopyWarning:
## A value is trying to be set on a copy of a slice from a DataFrame.
## Try using .loc[row_indexer,col_indexer] = value instead
##
## See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
```

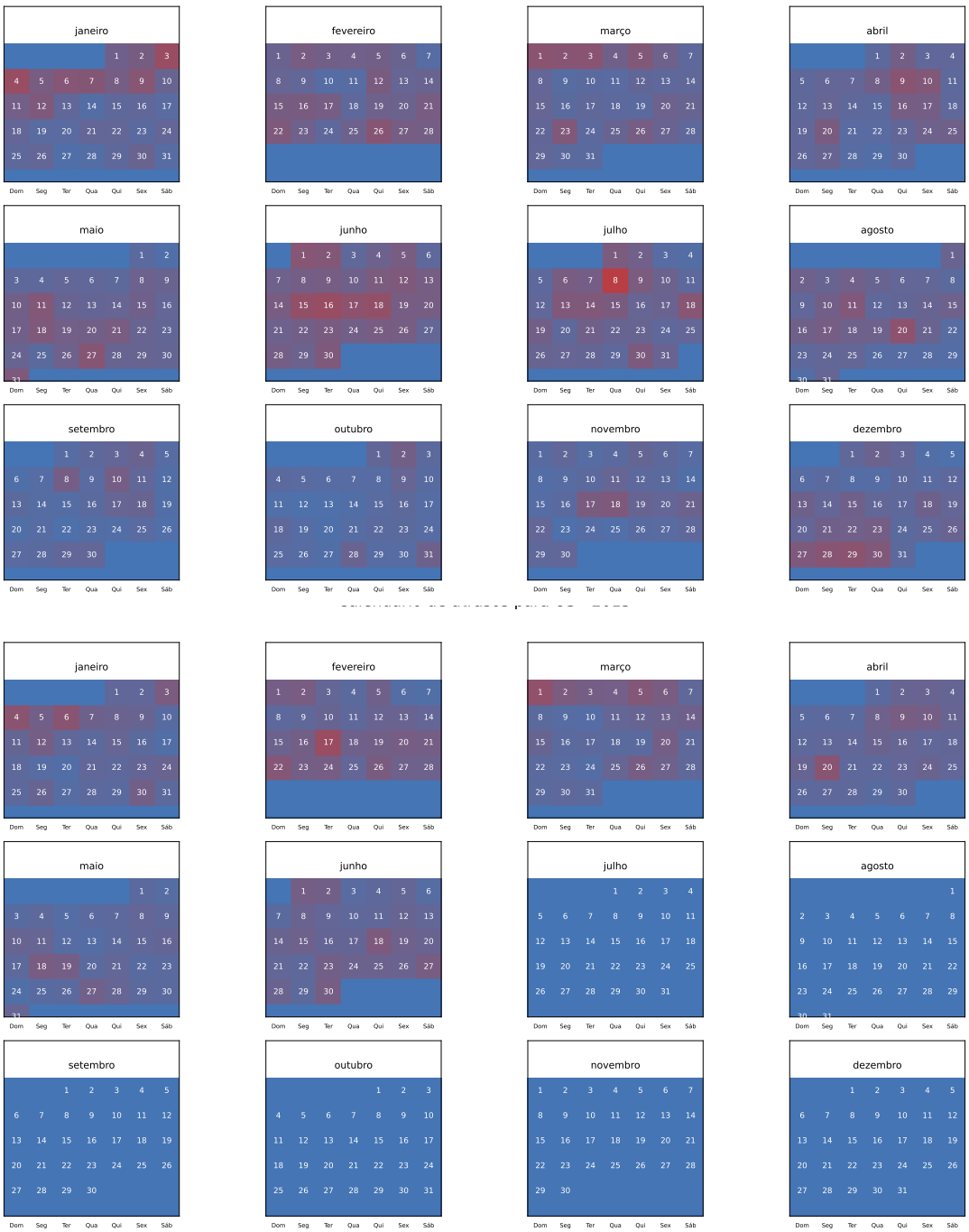
```
#Concatena todos os chunks processados em um único DataFrame
df_stats = pd.concat(chunks)

#Agrega os dados consolidados de todos os chunks
df_final = df_stats.groupby(['AIRLINE', 'DAY', 'MONTH']).agg(
    voos_atrasados=('voos_atrasados', 'sum'),  # Soma total de voos atrasados por dia/mês/companhia
    total_voos=('total_voos', 'sum')           # Soma total de voos
).reset_index()

# Calcula a porcentagem de voos atrasados
df_final['Perc'] = df_final['voos_atrasados'] / df_final['total_voos']

# Cria uma coluna de data completa (ano fixo 2015) para facilitar a plotagem
ano = 2015
df_final['Data'] = pd.to_datetime(dict(year=ano, month=df_final['MONTH'], day=df_final['DAY']))

#Função para desenhar um mês em formato de calendário com heatmap
def desenhar_mes(ax, ano, mes, datas, valores, cmap):


    ax.text(3, -1.0, calendar.month_name[mes], fontsize=10, ha='center', va='center')  # Título do mês
    ax.set_ylim(2, -2)

    cal = calendar.Calendar(firstweekday=6)  # Cria calendário começando no domingo

    # Obtém todos os dias do mês (com zeros onde não há dias do mês)
    dias = list(cal.itermonthdays(ano, mes))
    while len(dias) < 6*7:  # Garante que a matriz tenha 6 semanas
        dias.append(0)
    matriz = np.array(dias).reshape((6,7))  # Reshape para 6 linhas (semanas) x 7 colunas (dias)
```

```
    cores = np.zeros_like(matriz, dtype=float)  # Matriz para armazenar os valores de cor (porcentagem

    # Preenche a matriz de cores com os valores correspondentes ao dia
    for r in range(6):
        for c in range(7):
            dia = matriz[r,c]
            if dia != 0:  # Ignora os zeros (dias fora do mês)
                data_atual = pd.Timestamp(year=ano, month=mes, day=dia)
                valor = valores.get(data_atual, 0)  # Obtém o valor de atraso (ou 0 se não existir)
                cores[r,c] = valor

    # Plota o heatmap no eixo fornecido
    im = ax.imshow(cores, cmap=cmap, vmin=0, vmax=1)

    # Escreve os números dos dias sobre o heatmap
    for r in range(6):
        for c in range(7):
            dia = matriz[r,c]
            if dia != 0:
                ax.text(c, r, str(dia), ha='center', va='center', color='white', fontsize=8)

    # Configurações dos ticks e rótulos dos dias da semana
    ax.set_xticks(np.arange(7))
    ax.set_xticklabels(['Dom', 'Seg', 'Ter', 'Qua', 'Qui', 'Sex', 'Sáb'], fontsize=6)
    ax.set_yticks(np.arange(6))
    ax.set_yticklabels([])
    ax.tick_params(left=False, bottom=False)
    ax.set_xticks([], minor=True)
    ax.set_yticks([], minor=True)

    return im  # Retorna o objeto da imagem do heatmap

#Função principal para plotar calendário por companhia aérea
def calendario_por_companhia(df, companhia, ano=2015):
    # Filtra os dados da companhia aérea selecionada
    df_cia = df[df['AIRLINE'] == companhia]

    # Cria um dicionário mapeando Data -> Porcentagem de voos atrasados
    valores = pd.Series(df_cia.Perc.values, index=df_cia.Data).to_dict()

    # Cria uma figura com 12 subplots (3 linhas x 4 colunas, um para cada mês)
    fig, axs = plt.subplots(3, 4, figsize=(16,9))
    fig.suptitle(f'Calendário de atrasos para {companhia} - {ano}', fontsize=16, y=1.02)
    plt.subplots_adjust(hspace=0.5)  # Espaçamento entre os subplots

    # Cria colormap personalizado de azul (#4575b4) até vermelho (#d73027)
    cores_personalizadas = LinearSegmentedColormap.from_list("custom", ["#4575b4", "#d73027"])
    cmap = cores_personalizadas  # Aplica o colormap customizado ao gráfico

    # Desenha o calendário de cada mês
    for i in range(12):
        mes = i + 1
        ax = axs[i // 4, i % 4]  # Acessa o subplot correspondente
```

```
        desenhar_mes(ax, ano, mes, df_cia['Data'], valores, cmap)  # Chama função para desenhar o mês

    plt.tight_layout()  # Ajusta o layout dos subplots
    plt.show()          # Exibe o gráfico

# Calendário para cada companhia aérea
for cia in ['AA', 'DL', 'UA', 'US']: #separação por cia
    calendario_por_companhia(df_final, cia, ano=ano) #printando os calendarios por cia
```





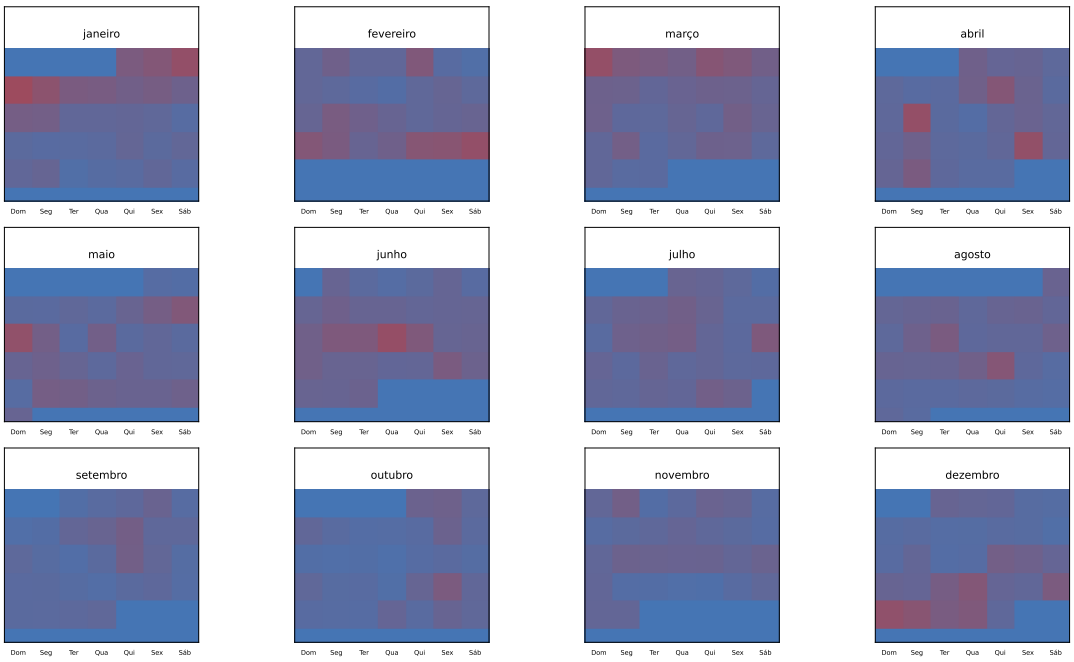


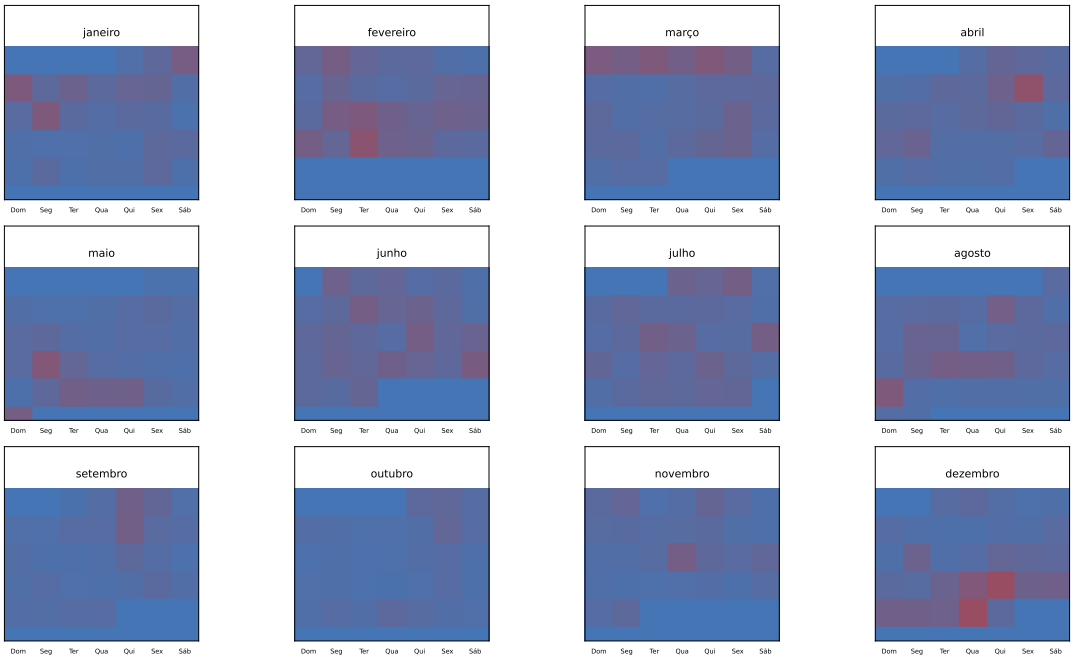
calendario sem os dias

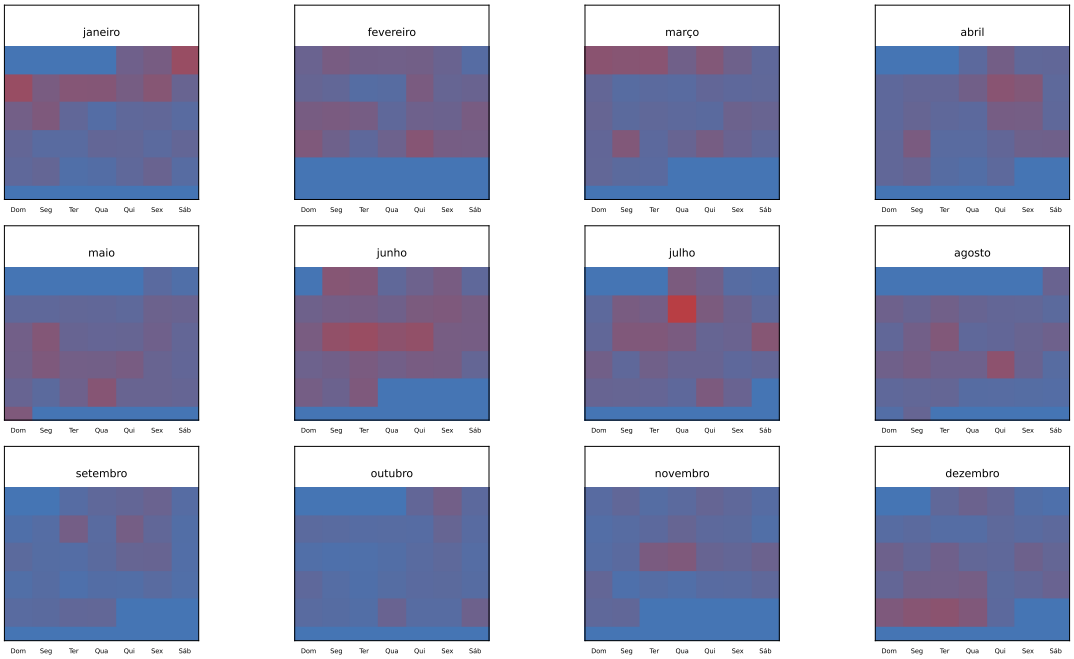
```
def desenhar_mes(ax, ano, mes, datas, valores, cmap):


    ax.text(3, -1.0, calendar.month_name[mes], fontsize=10, ha='center', va='center')  # Título do mês
    ax.set_ylim(2, -2)

    cal = calendar.Calendar(firstweekday=6)  # Cria calendário começando no domingo
```

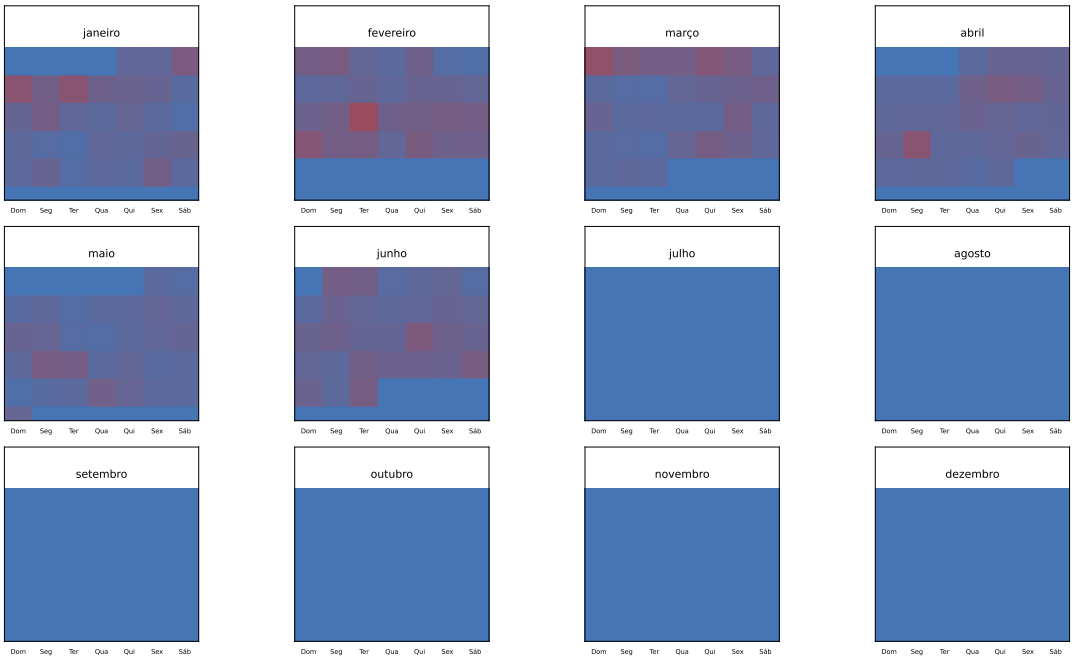
```
    # Obtém todos os dias do mês (com zeros onde não há dias do mês)
    dias = list(cal.itermonthdays(ano, mes))
    while len(dias) < 6*7:  # Garante que a matriz tenha 6 semanas
        dias.append(0)
    matriz = np.array(dias).reshape((6,7))  # Reshape para 6 linhas (semanas) x 7 colunas (dias)

    cores = np.zeros_like(matriz, dtype=float)  # Matriz para armazenar os valores de cor (porcentagem

    # Preenche a matriz de cores com os valores correspondentes ao dia
    for r in range(6):
        for c in range(7):
            dia = matriz[r,c]
            if dia != 0:  # Ignora os zeros (dias fora do mês)
                data_atual = pd.Timestamp(year=ano, month=mes, day=dia)
                valor = valores.get(data_atual, 0)  # Obtém o valor de atraso (ou 0 se não existir)
                cores[r,c] = valor

    # Plota o heatmap no eixo fornecido
    im = ax.imshow(cores, cmap=cmap, vmin=0, vmax=1)

    # Configurações dos ticks e rótulos dos dias da semana
    ax.set_xticks(np.arange(7))
    ax.set_xticklabels(['Dom', 'Seg', 'Ter', 'Qua', 'Qui', 'Sex', 'Sáb'], fontsize=6)
    ax.set_yticks(np.arange(6))
    ax.set_yticklabels([])
    ax.tick_params(left=False, bottom=False)
    ax.set_xticks([], minor=True)
    ax.set_yticks([], minor=True)

    return im  # Retorna o objeto da imagem do heatmap

#Função principal para plotar calendário por companhia aérea
def calendario_por_companhia(df, companhia, ano=2015):
    # Filtra os dados da companhia aérea selecionada
    df_cia = df[df['AIRLINE'] == companhia]

    # Cria um dicionário mapeando Data -> Porcentagem de voos atrasados
    valores = pd.Series(df_cia.Perc.values, index=df_cia.Data).to_dict()

    # Cria uma figura com 12 subplots (3 linhas x 4 colunas, um para cada mês)
    fig, axs = plt.subplots(3, 4, figsize=(16,9))
    fig.suptitle(f'Calendário de atrasos para {companhia} - {ano}', fontsize=16, y=1.02)
    plt.subplots_adjust(hspace=0.5)  # Espaçamento entre os subplots

    # Cria colormap personalizado de azul (#4575b4) até vermelho (#d73027)
    cores_personalizadas = LinearSegmentedColormap.from_list("custom", ["#4575b4", "#d73027"])
    cmap = cores_personalizadas  # Aplica o colormap customizado ao gráfico

    # Desenha o calendário de cada mês
    for i in range(12):
        mes = i + 1
        ax = axs[i // 4, i % 4]  # Acessa o subplot correspondente
        desenhar_mes(ax, ano, mes, df_cia['Data'], valores, cmap)  # Chama função para desenhar o mês
```

```
    plt.tight_layout()  # Ajusta o layout dos subplots
    plt.show()          # Exibe o gráfico

# Calendário para cada companhia aérea
for cia in ['AA', 'DL', 'UA', 'US']: #separação por cia
    calendario_por_companhia(df_final, cia, ano=ano) #printando
```






janeiro
fevereiro
março
abril
maio
junho
julho
agosto
setembro
outubro
novembro
dezembro
Dom Seg Ter Qua Qui Sex Sáb


janeiro
fevereiro
março
abril
maio
junho
julho
agosto
setembro
outubro
novembro
dezembro
Dom Seg Ter Qua Qui Sex Sáb