

# Mapas em Dashboards Web com Plotly em Python

## Principais Tipos de Mapas no Plotly

Os principais gráficos de mapas disponíveis no Plotly para Python são:

### 1. **Choropleth**

Exibe valores numéricos agregados por região (países, estados, municípios) preenchendo áreas geográficas com cores graduais.

### 2. **Scattermapbox**

Plota pontos geolocalizados sobre o mapa do Mapbox, ideal para marcadores com coordenadas latitude/longitude.

### 3. **Densitymapbox**

Representa densidade de pontos usando gradientes de cor sobre o fundo Mapbox.

### 4. **Scattergeo**

Similar ao Scattermapbox, mas utiliza o provedor GeoJSON interno, sem requerer chave Mapbox.

### 5. **Choroplethmapbox**

Combina o Choropleth com o provedor Mapbox, permitindo estilização avançada de mapas e tamanhos dinâmicos.

## 1. Configuração Básica

Antes de tudo, instale as bibliotecas necessárias:

```
pip install plotly dash
```

Importe no seu script Python:

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

## 2. Exemplo de Dashboard em Dash (HTML)

A seguir um exemplo completo que demonstra o uso de diferentes mapas:

```
# app.py
import dash
from dash import html, dcc
import plotly.express as px
```

```

import pandas as pd

# 1. Dados de exemplo: países e valores fictícios
df = pd.DataFrame({
    "country": ["Brazil", "Argentina", "Colombia", "Chile", "Peru"],
    "value": [100, 80, 120, 60, 90],
    "lat": [-14.2350, -38.4161, 4.5709, -35.6751, -9.1900],
    "lon": [-51.9253, -63.6167, -74.2973, -71.5430, -75.0152]
})

# 2. Mapa Choropleth global
fig_choro = px.choropleth(
    df,
    locations="country",
    locationmode="country names",
    color="value",
    color_continuous_scale="Viridis",
    title="Choropleth por País"
)

# 3. Scattermapbox
px.set_mapbox_access_token("YOUR_MAPBOX_TOKEN")
fig_scatter = px.scatter_mapbox(
    df,
    lat="lat",
    lon="lon",
    size="value",
    color="value",
    color_continuous_scale="Plasma",
    zoom=2,
    title="Pontos Geolocalizados"
)

# 4. Densitymapbox
fig_density = px.density_mapbox(
    df,
    lat="lat",
    lon="lon",
    z="value",
    radius=20,
    center=dict(lat=-15, lon=-60),
    zoom=3,
    title="Densidade de Pontos"
)

# 5. Layout do Dash
app = dash.Dash(__name__)
app.layout = html.Div([
    html.H1("Dashboard de Mapas com Plotly"),
    dcc.Graph(figure=fig_choro),
    dcc.Graph(figure=fig_scatter),
    dcc.Graph(figure=fig_density)
])

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

```

## Como usar

1. Substitua "YOUR\_MAPBOX\_TOKEN" por sua chave do Mapbox (gratuita no site Mapbox).
2. Execute `python app.py` para iniciar o servidor local.
3. Abra `http://127.0.0.1:8050/` no navegador para visualizar o dashboard.

## 3. Customizações e Dicas

- **Temas e Cores**

Use `color_continuous_scale` para definir paletas personalizadas (e.g., "Viridis", "Cividis", "Inferno").

- **Filtros Interativos**

Integre componentes Dash como `dcc.Dropdown` e callbacks para filtrar dados em tempo real.

- **GeoJSON Customizado**

Para áreas específicas (municípios do Brasil), carregue um arquivo GeoJSON e use `px.choropleth(geojson=...)`.

- **Responsividade HTML**

Envolva gráficos em containers flexíveis e ajuste `style={'width': '100%', 'height': '100%'}` para layouts responsivos.

## 4. Exemplo com GeoJSON (Municípios)

```
import json

with open("municipios_sp.geojson", "r", encoding="utf-8") as f:
    geojson = json.load(f)

df_mun = pd.DataFrame({
    "municipio": ["São Paulo", "Campinas", "Santos"],
    "value": [200, 150, 100]
})

fig_mun = px.choropleth_mapbox(
    df_mun,
    geojson=geojson,
    featureidkey="properties.NM_MUNICIP",
    locations="municipio",
    color="value",
    center={"lat": -23.55, "lon": -46.63},
    zoom=7,
    mapbox_style="carto-positron",
    title="Valores por Município de SP"
)
```

Insira `fig_mun` em um `dcc.Graph` para exibir.

## Conclusão

Plotly em Python oferece diversos tipos de mapas para dashboards web responsivos. Use

Choropleth para agregados regionais, Scattermapbox para pontos, Densitymapbox para calor de densidade e combine com Dash para interatividade e HTML limpo.