

Fundamentos de Python (UFCD 10809)

PPT02

Sandra Liliana Meira de Oliveira



Seaborn

<https://seaborn.pydata.org/>

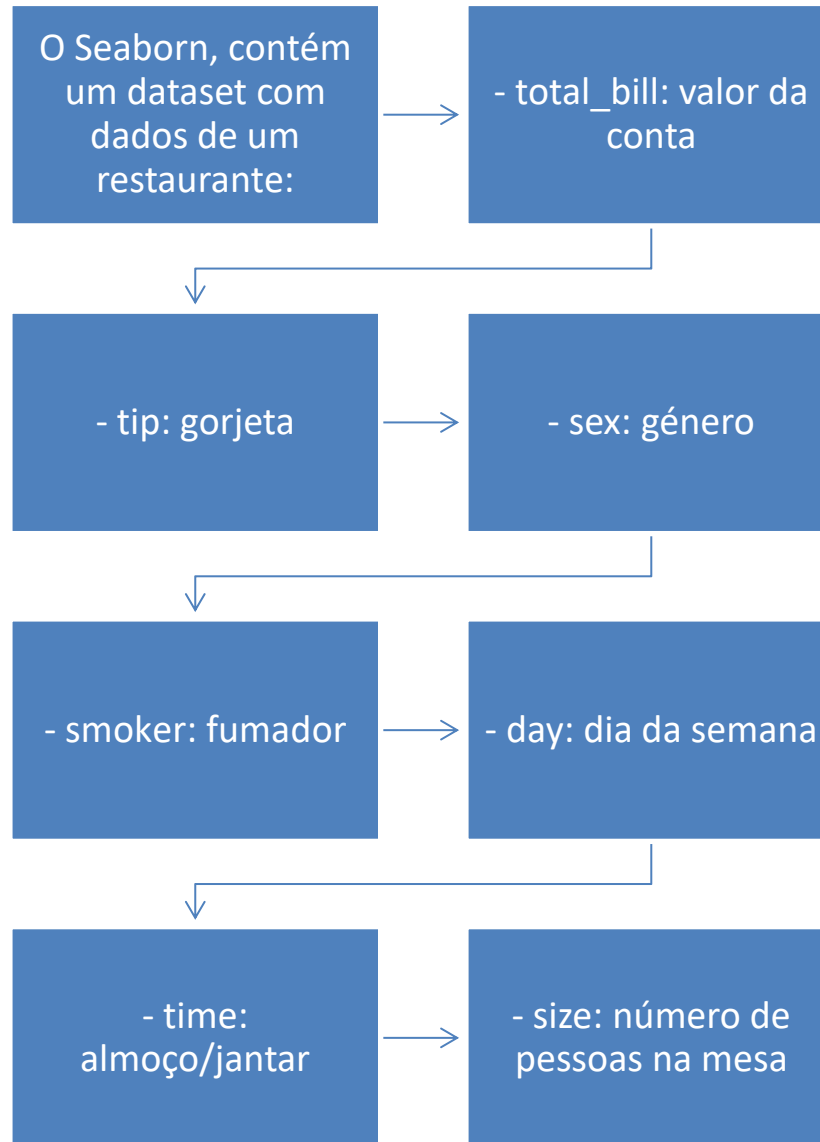
Introdução ao Seaborn

O Seaborn é uma biblioteca Python baseada no Matplotlib.

Permite criar gráficos estatísticos com menos código e com melhor aparência estética.

É altamente integrado com pandas DataFrames.

Dataset Utilizado: Tips



Funções Principais do Seaborn

- `sns.scatterplot()`: gráfico de dispersão
- `sns.lineplot()`: gráfico de linha
- `sns.barplot()`: médias por categoria
- `sns.countplot()`: contagem de ocorrências
- `sns.boxplot()`: mediana, quartis e outliers
- `sns.violinplot()`: distribuição e densidade
- `sns.histplot()`: distribuição de frequência
- `sns.heatmap()`: matriz de correlação
- `sns.pairplot()`: grelha de dispersões

A Reter

A maioria das funções de visualização do Seaborn utiliza o mesmo padrão de parâmetros:

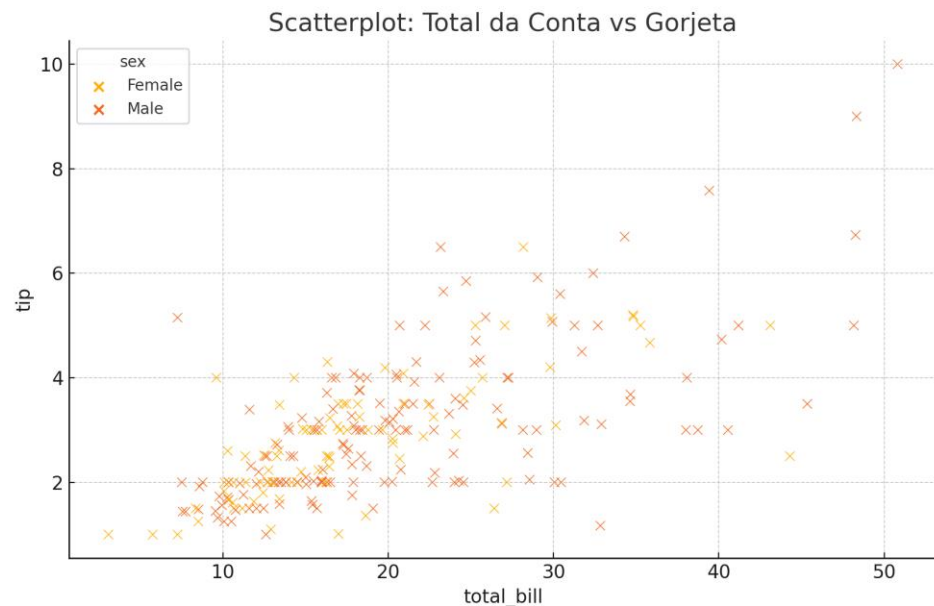
- **`data=` : define o DataFrame.**
- **`x=` e `y=` : mapeiam as variáveis para os eixos.**
- **`hue=`, `style=`, `size=` : permitem categorização visual adicional.**

Exemplo: scatterplot

```
sns.scatterplot(data=df, x="total_bill", y="tip", hue="sex")
```

- Gráfico de dispersão entre total_bill e tip
- hue='sex' cor por gênero
- Mostra relação entre duas variáveis numéricas

- data= define o dataset
- x= e y= para variáveis
- hue= cor por categoria
- style= define o marcador
- size= altera o tamanho dos pontos

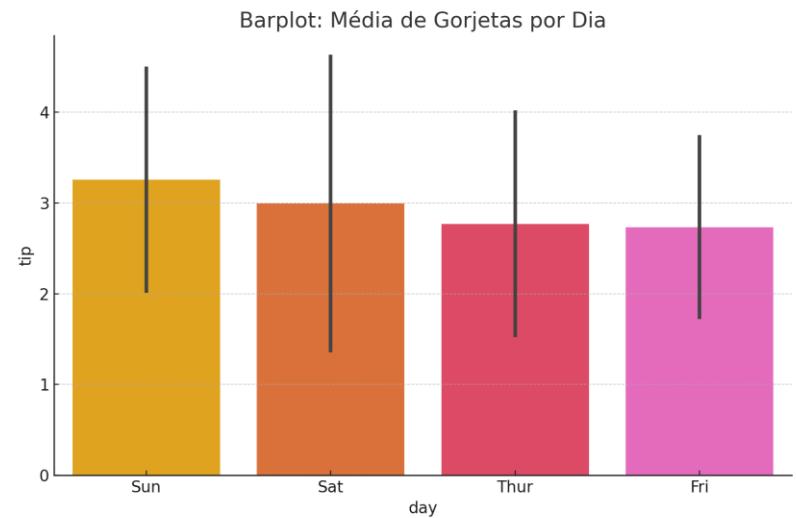
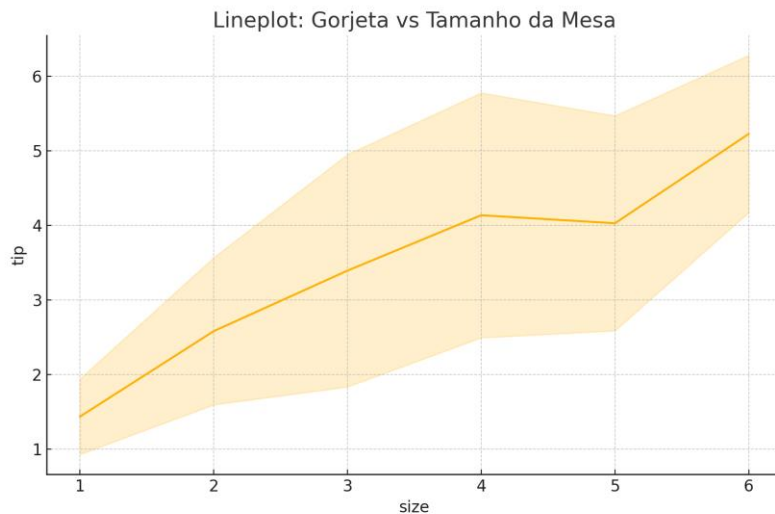


Statistical Operations

Funções como barplot e lineplot calculam estatísticas automáticas

Exemplo:

`sns.barplot(data=df, x="day", y="tip", ci="sd")` -> Média com barras de erro (desvio padrão)

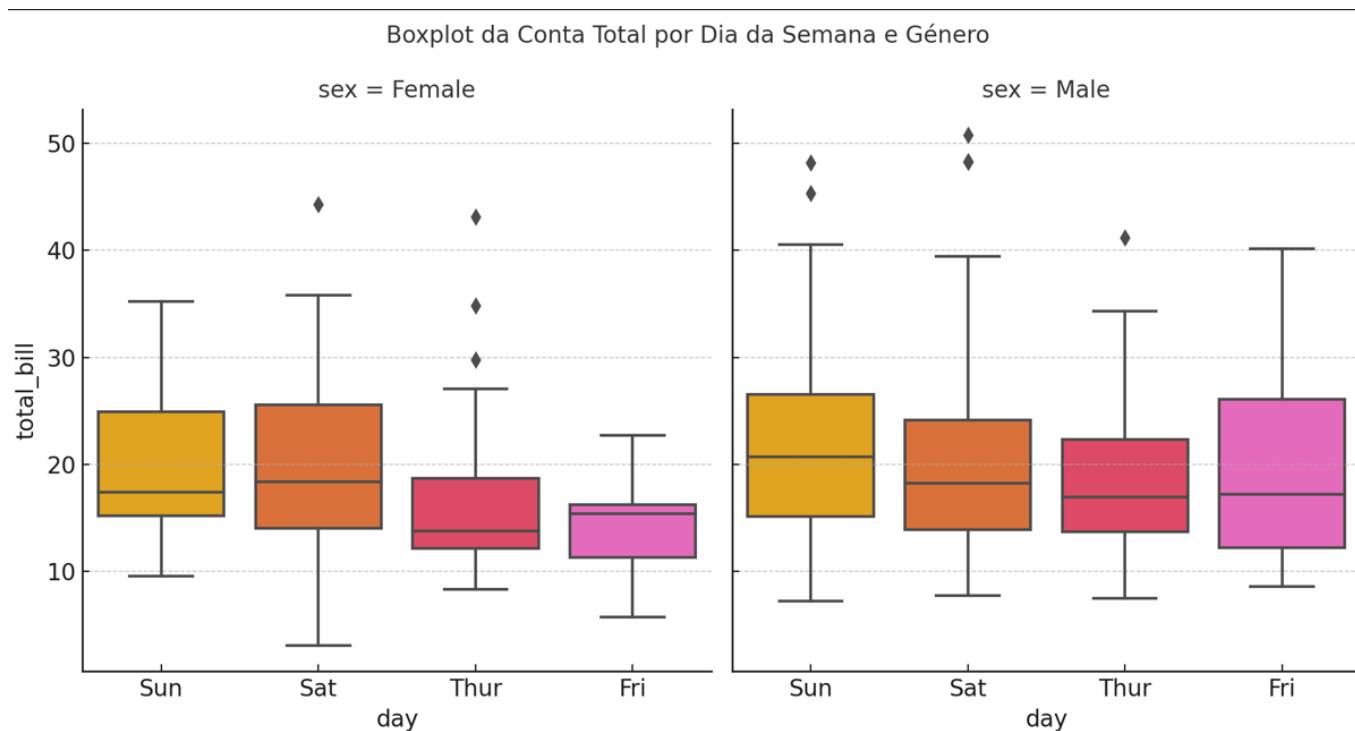


Multi-plot Grids

- FacetGrid, catplot, pairplot, jointplot
- Permitem criar grelhas com base em variáveis categóricas





Exemplo:

```
sns.catplot(data=df, x="day", y="total_bill", col="sex", kind="box")
```



Explicação gráfico anterior

Para cada **combinação de género e dia da semana**, o gráfico mostra:

-  **Linha no meio da caixa:** a mediana do valor total da conta;
-  **Caixa (box):** os quartis (25% e 75%);
-  **Linhas verticais (bigodes):** a dispersão dos dados dentro de 1.5x o intervalo interquartil;
-  **Pontos fora da caixa:** *outliers*, ou seja, contas muito mais altas do que o normal.

✅ O que podemos concluir visualmente?

Podemos comparar a distribuição dos valores da conta por dia entre géneros. Ver se há diferenças significativas entre homens e mulheres em termos de gastos em diferentes dias. Verificar a presença de outliers (ex: contas muito altas ao sábado).

Tema Avançado: Regressão Linear

```
sns.Implot(data=df, x="total_bill", y="tip", hue="sex")
```

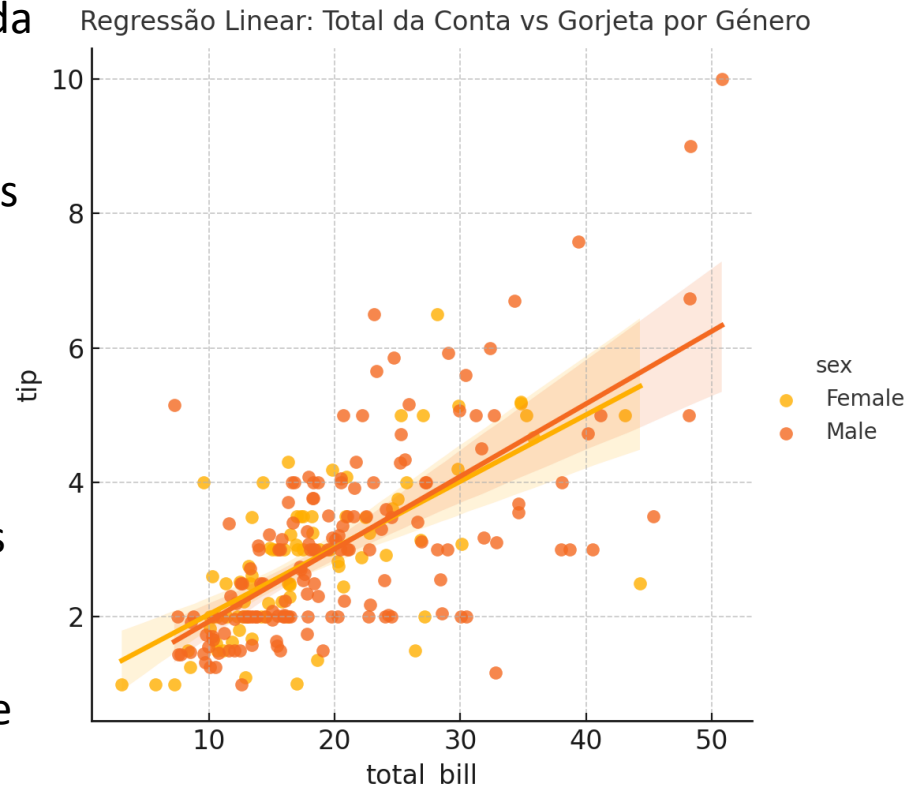
- Adiciona linha de regressão
- Permite observar tendência entre duas variáveis
- hue separa linhas por categoria

Este gráfico mostra a relação entre o valor da conta (total_bill) e a gorjeta (tip)

O Seaborn desenha automaticamente linhas de regressão linear para cada grupo de sex (homem/mulher), distinguido por cor com hue="sex".

Cada ponto representa uma observação. A linha mostra a tendência geral entre os dois valores.

A área sombreada representa o intervalo de confiança (por defeito 95%).

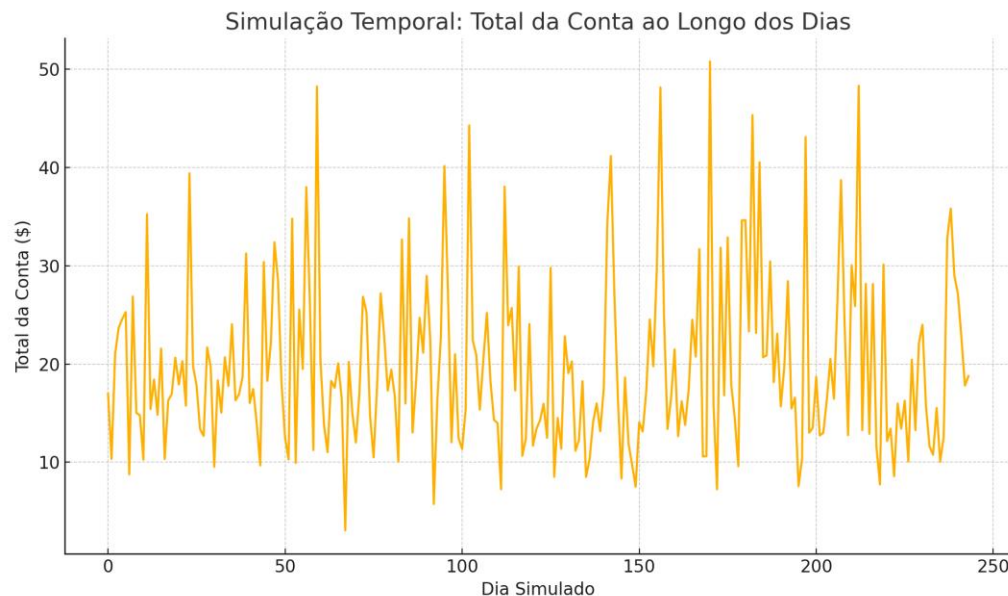


Tema Avançado: Visualização Temporal

```
df['dia_simulado'] = range(len(df))
```

```
sns.lineplot(data=df, x="dia_simulado", y="total_bill")
```

- Simula evolução temporal
- Mostra tendências ou padrões ao longo do tempo



Explicação gráfico anterior


-  Explicação: `df['dia_simulado'] = range(len(df))`: cria uma variável sequencial como se fosse "dias"
- `sns.lineplot(...)`: desenha a linha que conecta os valores de `total_bill` ao longo do tempo.
- Este tipo de gráfico é útil para observar tendências, padrões ou flutuações ao longo do tempo, mesmo que simulado.

Figure Aesthetics



- sns.set_theme()



- sns.set_style()



- sns.set_palette()



- sns.set_context()



Permitem personalizar aparência,
tamanho e cores dos gráficos.