

Visualização de Dados



Visualização de Dados

“Na ciência de dados, o processo de produção de dados em informação e conhecimento, a visualização é uma parte imprescindível. Sem ela, não se tem resultado, não se chega ao objetivo do trabalho, não se pode contar a história...não da pra demonstrar o que aconteceu”



Visualização de Dados

O conceito de visualização de dados é bem simples. Visualização de dados nada mais é que a representação gráfica, estática ou em movimento, da análise de dados e indicadores de natureza variada que podemos tirar inferências (Wikipedia, 2022).



Visualização de Dados

- **Gráficos estáticos:** É uma representação instantânea de um conjunto de dados.
- **Dashboards:** Conjunto de gráficos interativos e exibidos em conjunto.
- **Infográficos:** Gráficos mais ricos visualmente, porém não estão vinculados diretamente a uma fonte de dados.

Visualização de Dados

Objetivos

Comunicar (Explicativa)

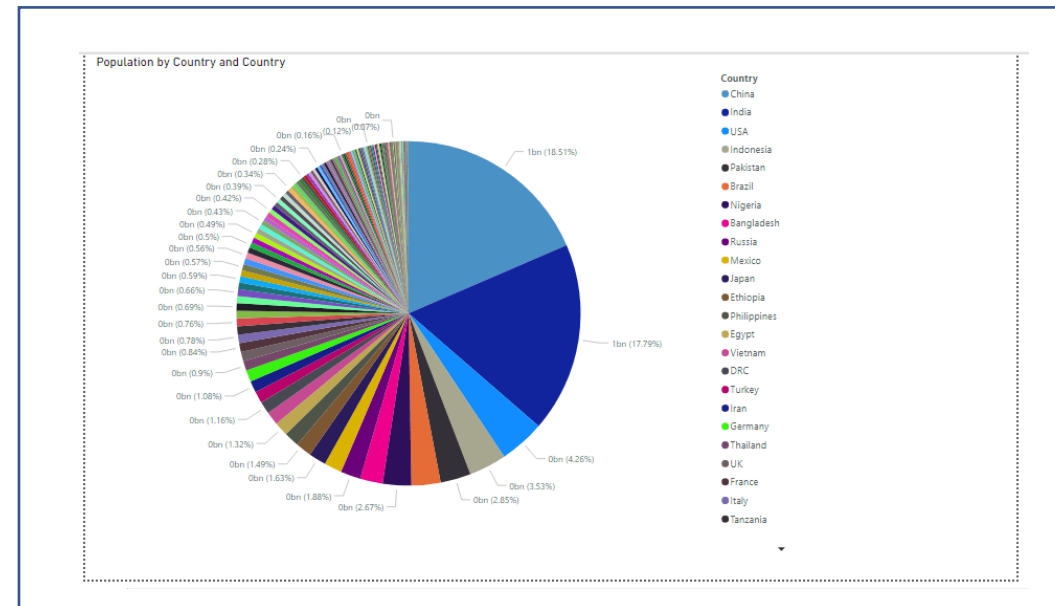
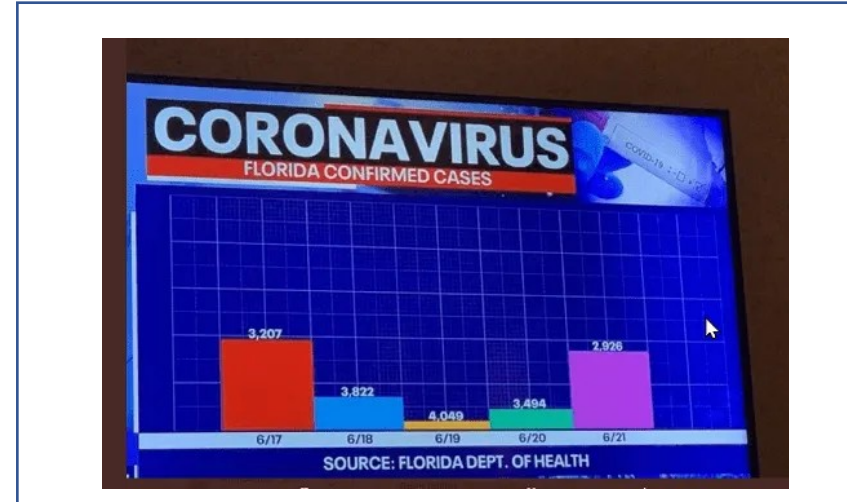
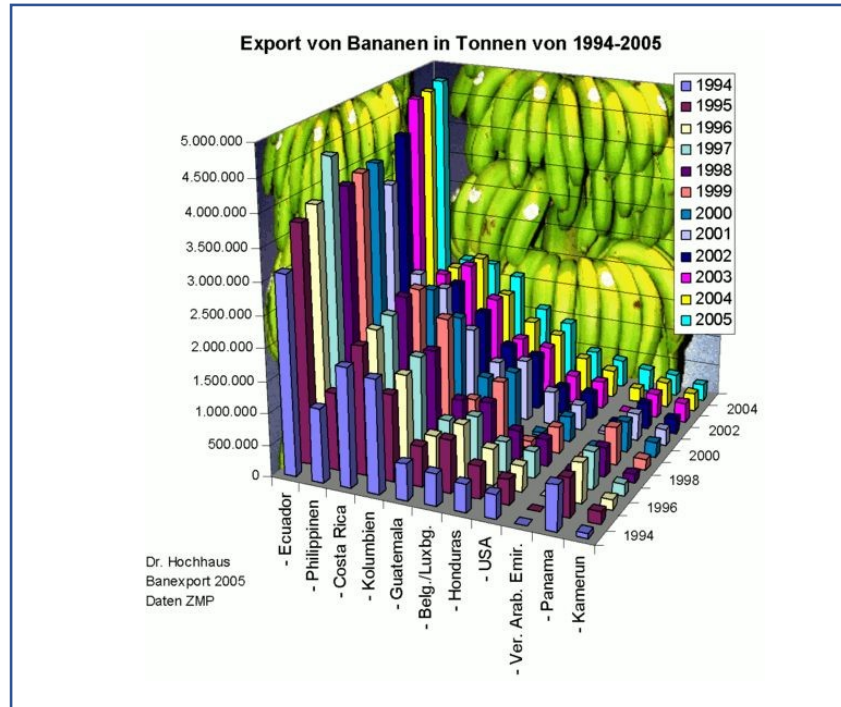
- Apresentar dados e ideias
- Explicar e informar
- Fornecer evidências e suporte
- Influenciar e persuadir

Analisar (Exploratória)

- Explorar os dados
- Avaliar uma situação
- Determinar como proceder
- Decidir o que fazer

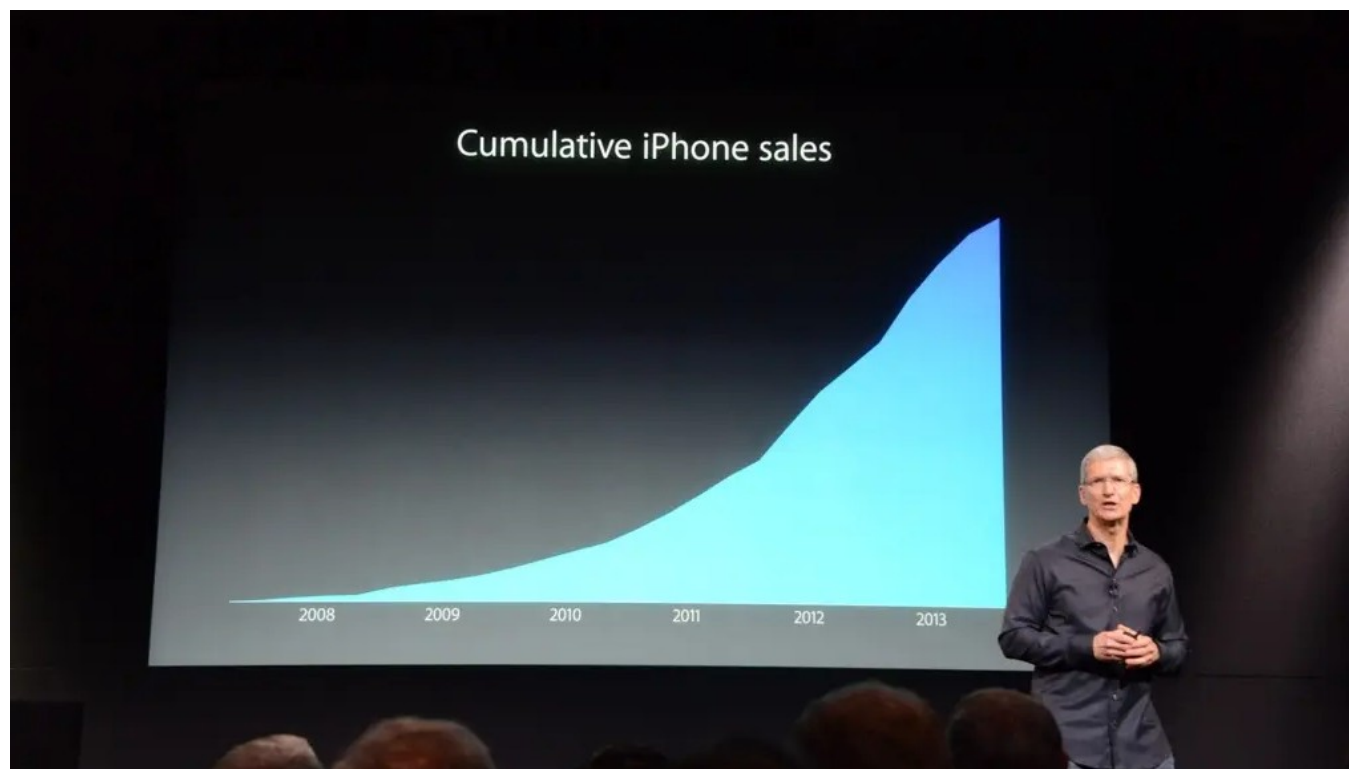
Visualização de Dados

Como não fazer gráficos!



Visualização

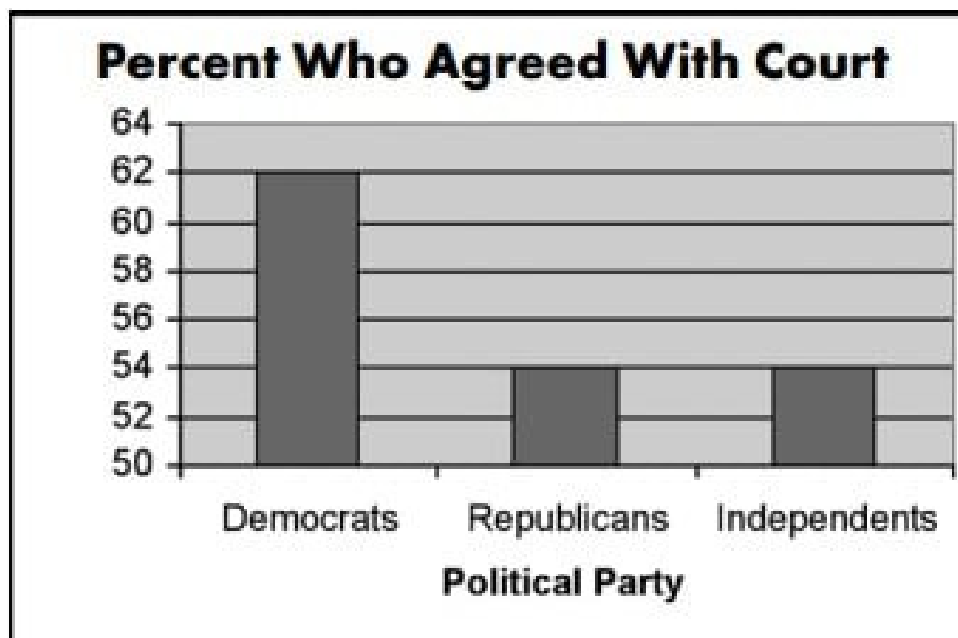
Em 2013, a Apple foi criticada pelo mesmo erro quando Tim Cook usou o gráfico específico para mostrar a crescente venda de iPads entre os anos de 2008-2013.



Fonte: Mail Online - How Apple exaggerated sales of its iPad: Chart shown at launch is 'misleading' because it fails to show recent dip in sales, 2013

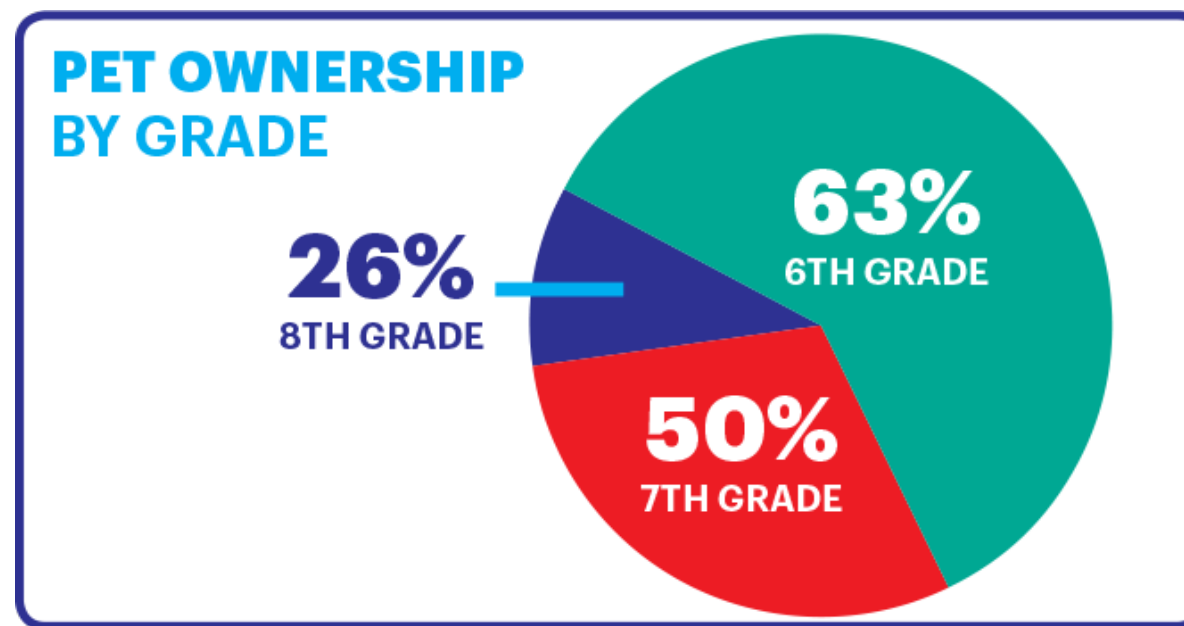
Visualização de Dados

Na imagem abaixo, uma olhada daria a impressão de que um número esmagador de democratas apoiou a decisão do tribunal do que os republicanos. No entanto, a diferença não é tão grande (democratas – 62% vs. republicanos – 54%) e só se faz aparecer mais começando o gráfico a partir de 50%, o que se chama truncar o gráfico.



Visualização de Dados

Os gráficos de pizza mostram partes de um todo em que o valor do todo soma 100%. No entanto, a soma das partes apresentadas na imagem abaixo soma 139%, o que é tecnicamente absurdo para ser apresentado por meio de um gráfico de pizza.



Visualização de Dados

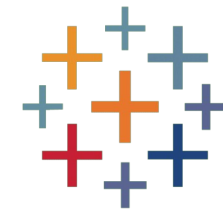
Tipos de gráficos:

- Linhas
- Setores
- Dispersão (2d, 3d)
- Barras
- Histograma
- Boxplot
- Headmap
- Mapas chloropleth
- Dispersão multivariada
- Linhas paralelas

Ferramentas para visualização de dados

The logo for matplotlib, featuring the word "matplotlib" in a blue sans-serif font. The "plot" part is in a lighter blue, and the "lib" part is in a darker blue. A small circular icon with a colorful pie chart is positioned between "plot" and "lib".The logo for seaborn, featuring a circular icon with a blue and white bar chart and a green line graph. To the right of the icon, the word "seaborn" is written in a dark blue sans-serif font.The logo for plotly, featuring a series of five vertical bars of increasing height, each with a different colored dot (pink, purple, blue, green, yellow) above it. To the right of the bars, the word "plotly" is written in a dark blue sans-serif font.

Power BI



+ a b | e a u

Existem diversas ferramentas poderosas para visualização de dados dentro do mercado, como por exemplo, Power BI e próprio Tableau, mas foco da aula será nas livrarias **open source** de visualização de dados com python (matplotlib, seaborn e plotly).

Visualização de Dados

❑ Matplotlib

Matplotlib é uma das principais bibliotecas de visualização de dados com Python, o matplotlib é abrangente para criar arquivos estáticos, animados, e visualizações interativas em Python. Matplotlib torna as coisas difíceis possíveis.



Visualização de Dados

❏ Matplotlib

Instalando matplotlib

```
>> pip install matplotlib
```

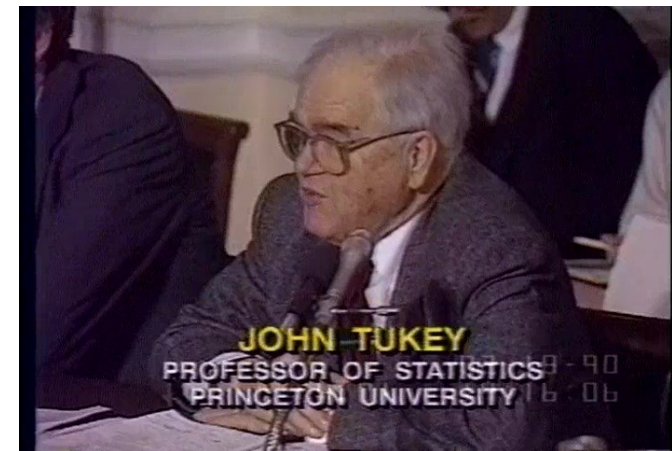
Importando matplotlib

```
import matplotlib.pyplot as plt
```

Para instalar o matplotlib de “**pip install matplotlib**” no terminal. Dentro do google colab o matplotlib já vem por padrão. Para se utilizar o próprio notebook como **backend** dos seus gráficos deve ser utilizado “**%matplotlib inline**” após a importação do matplotlib.

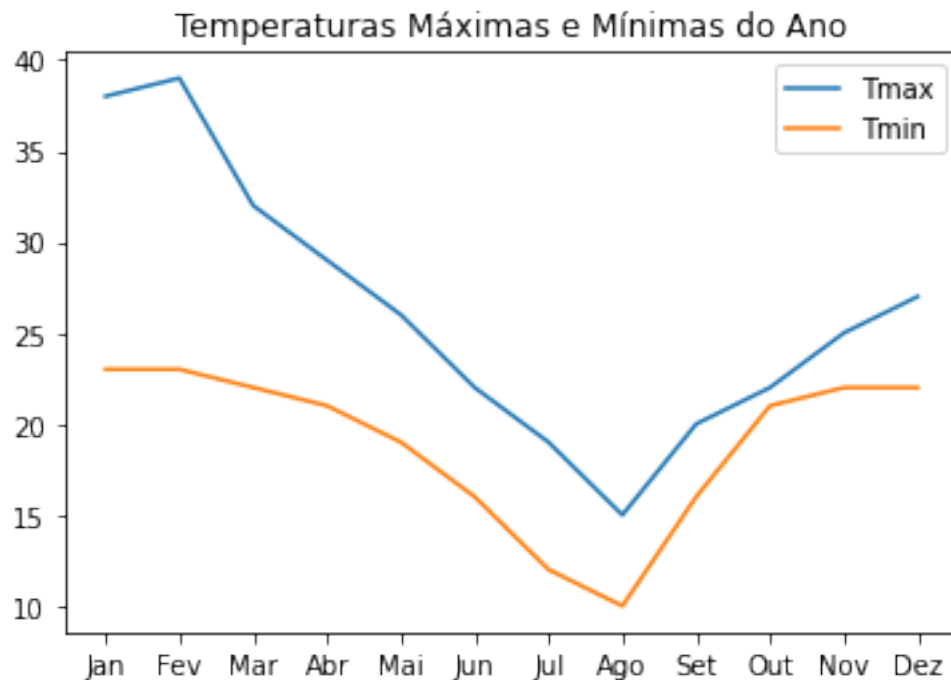
Visualização de Dados

Jonh Tokey “O maior valor de uma imagem é quando ela nos obriga a perceber o que nunca esperávamos ver. ”



Visualização de Dados

Gráfico de linhas: geralmente usado para representar tendências em séries temporais.



Matplotlib - Gráfico de linha Temperaturas

```
import matplotlib.pyplot as plt

meses = ['Jan', 'Fev', 'Mar', 'Abr',
         'Mai', 'Jun', 'Jul', 'Ago',
         'Set', 'Out', 'Nov', 'Dez']

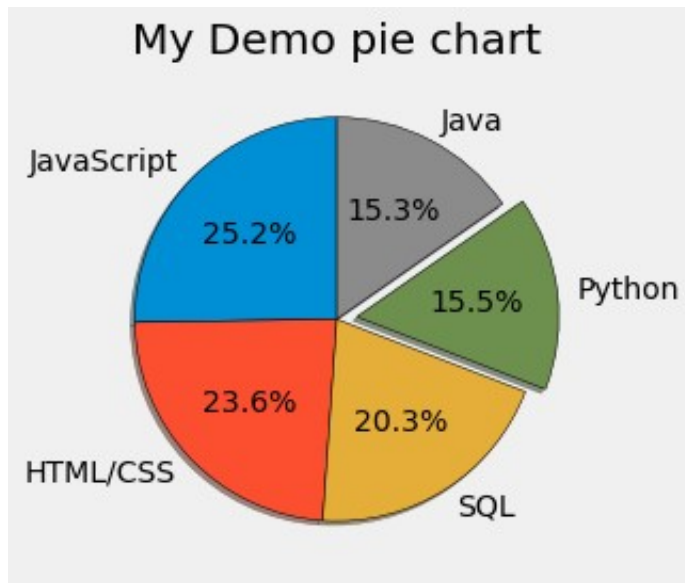
Tmax = [38, 39, 32, 29, 26, 22, 19, 15, 20, 22, 25, 27]
Tmin = [23, 23, 22, 21, 19, 16, 12, 10, 16, 21, 22, 22]

plt.title('Temperaturas Máximas e Mínimas do Ano')
plt.plot(meses, Tmax, meses, Tmin)
plt.legend(['Tmax', 'Tmin'])
plt.show()
```

Fonte: Autor próprio, 2022.

Visualização de Dados

Gráficos de setores ou de “pizza”: deve ser utilizado quando temos poucos setores, por exemplo, não mais de que 7 categorias e ainda existir uma diferença significativa entre os setores do gráfico.



Fonte: Habib - Exploring the Basics of Pie Chart in Matplotlib, 2020

Matplotlib - pie

```
plt.style.use('fivethirtyeight')

slices = [59219, 55466, 47544, 36443, 35917]

labels = ['JavaScript', 'HTML/CSS', 'SQL', 'Python', 'Java']

explode = [0,0,0,0.1,0]

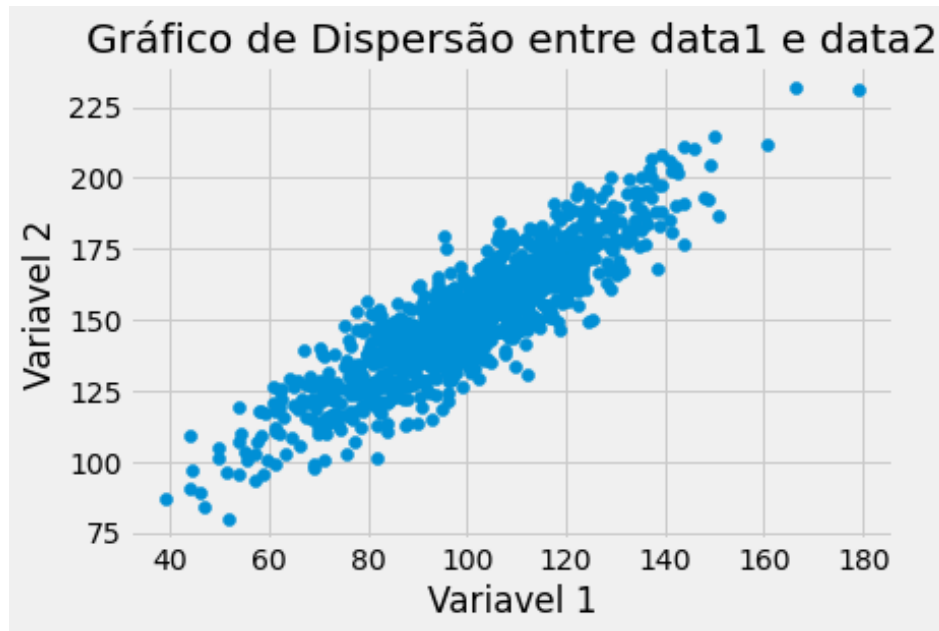
plt.pie(slices,
        labels=labels,
        explode=explode,
        shadow=True,
        startangle=90,
        autopct='%1.1f%%',
        wedgeprops= {'edgecolor':'black'})

plt.title('My Demo pie chart')
plt.tight_layout()
plt.show()
```

Fonte: Autor próprio, 2022.

Visualização de Dados

Gráfico de dispersão: usado para destacar os valores de duas ou até 3 variáveis diferentes como pontos em um gráfico plano.



Fonte: Habib - Exploring the Basics of Scatter Chart in Matplotlib, 2020

Matplotlib - Scatter

```
from numpy import std
from numpy import correlate
from numpy.random import randn
from numpy.random import seed
from matplotlib import pyplot

seed(1)

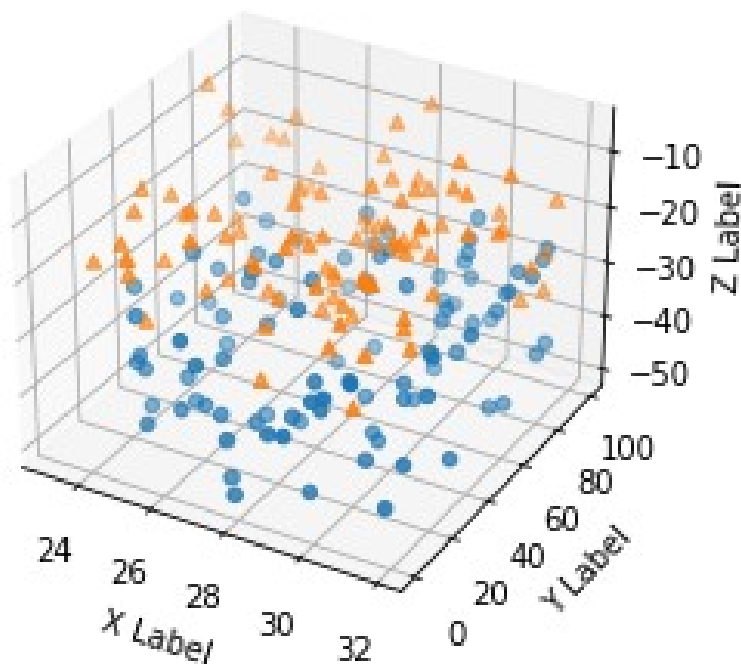
x = 20 * randn(1000) + 100
y = x + (10 * randn(1000) + 50)

pyplot.scatter(x, y)
pyplot.title('Gráfico de Dispersão entre data1 e data2')
pyplot.xlabel("Variavel 1")
pyplot.ylabel("Variavel 2")
pyplot.show()
```

Fonte: Autor próprio, 2022.

Visualização de Dados

Gráfico de dispersão: usado para destacar os valores de duas ou até 3 variáveis diferentes como pontos em um gráfico.



Matplotlib – Scatter 3d

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

# Fixing random state for reproducibility
np.random.seed(19680801)

def randrange(n, vmin, vmax):
    """
    Helper function to make an array of random numbers having
    shape (n, )
    with each number distributed Uniform(vmin, vmax).
    """
    return (vmax - vmin)*np.random.rand(n) + vmin

fig = plt.figure()
ax = fig.add_subplot(projection='3d')

n = 100

# For each set of style and range settings, plot n random
# points in the box
# defined by x in [23, 32], y in [0, 100], z in [zlow,
# zhigh].
for m, zlow, zhigh in [('o', -50, -25), ('^', -30, -5)]:
    xs = randrange(n, 23, 32)
    ys = randrange(n, 0, 100)
    zs = randrange(n, zlow, zhigh)
    ax.scatter(xs, ys, zs, marker=m)

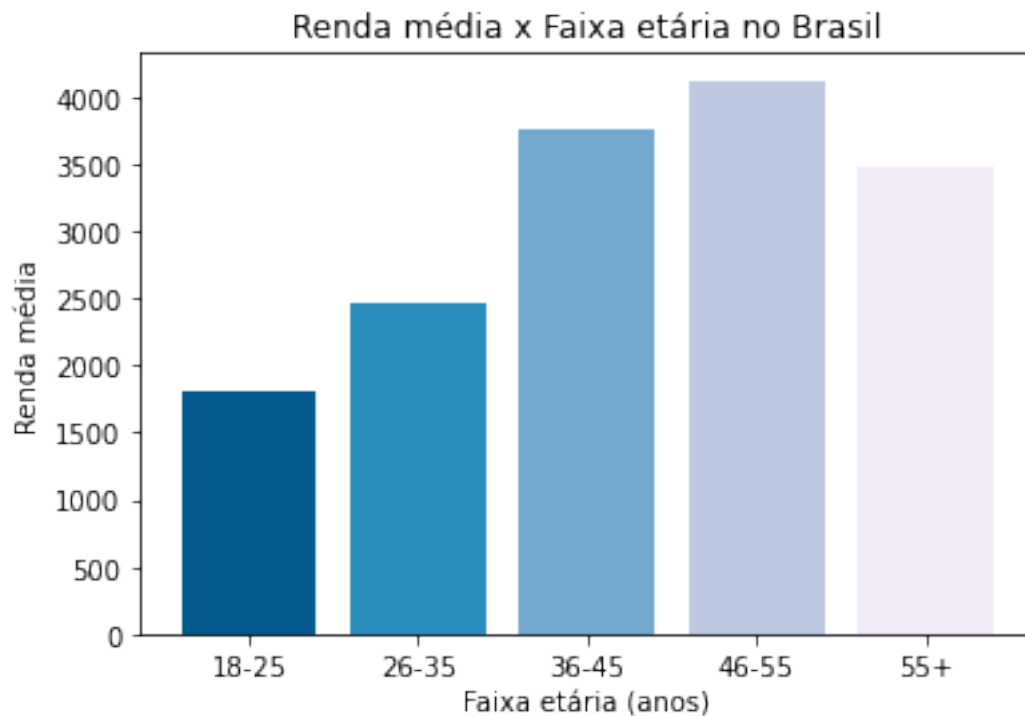
ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')

plt.show()
```

Visualização de Dados

Gráfico de Barras: Geralmente usado para comparar categorias e grupos.

Ótimo gráfico para ranking ou comparação de magnitude.



Fonte: Budkewics, M. – Medium - Gráficos-de-barras-com-matplotlib, 2018.

```
# Primeiramente importamos as libs
import numpy as np
import matplotlib.pyplot as plt

# Um gráfico com a renda média por faixa etária será nosso
Hello World, vamos definir alguns valores fictícios
faixaEtaria = ['18-25', '26-35', '36-45', '46-55', '55+']
renda = [1805.45, 2458.12, 3752.15, 4120.89, 3486.22]

plt.bar(faixaEtaria, renda, color = ["#045a8d", "#2b8cbe",
"#74a9cf", "#bdc9e1", "#f1eef7"])

# Aqui definimos as legendas de cada barra no eixo X
plt.xticks(faixaEtaria)

# A label para o eixo Y
plt.ylabel('Renda média')

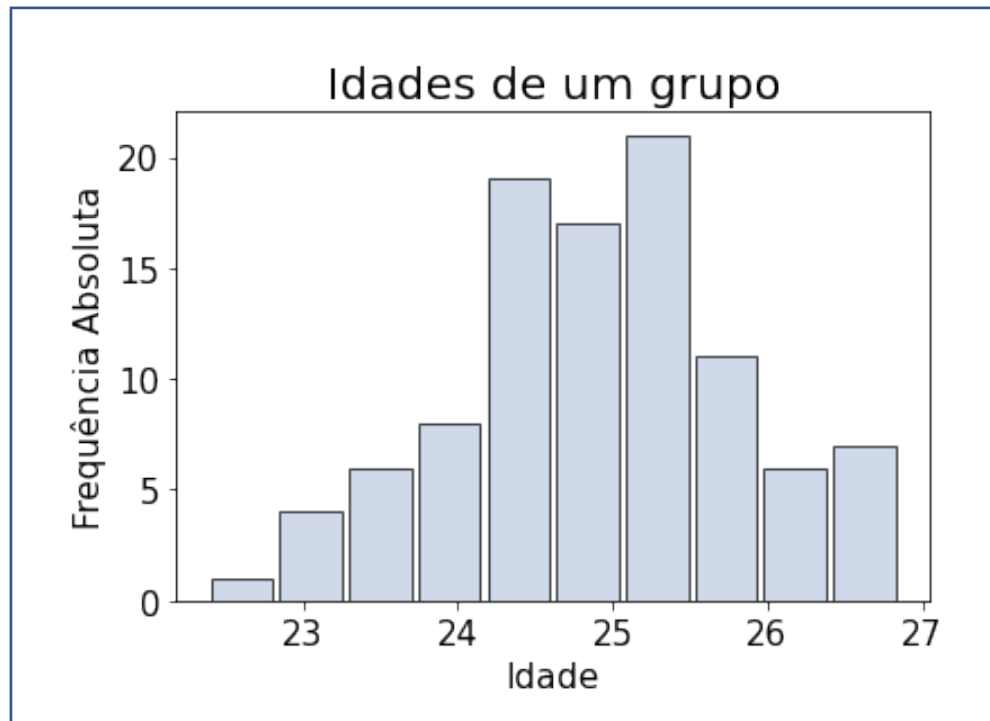
# A label para o eixo X
plt.xlabel('Faixa etária (anos)')

# O título do gráfico
plt.title('Renda média x Faixa etária no Brasil')
plt.show()
```

Fonte: Autor próprio, 2022.

Visualização de Dados

Histograma: O histograma, também conhecido como distribuição de frequências, é a representação gráfica em colunas ou em barras de um conjunto de dados previamente tabulado e dividido em classes uniformes ou não uniformes.



Fonte: Autor próprio, 2022.

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

# Definindo uma semente
np.random.seed(42)

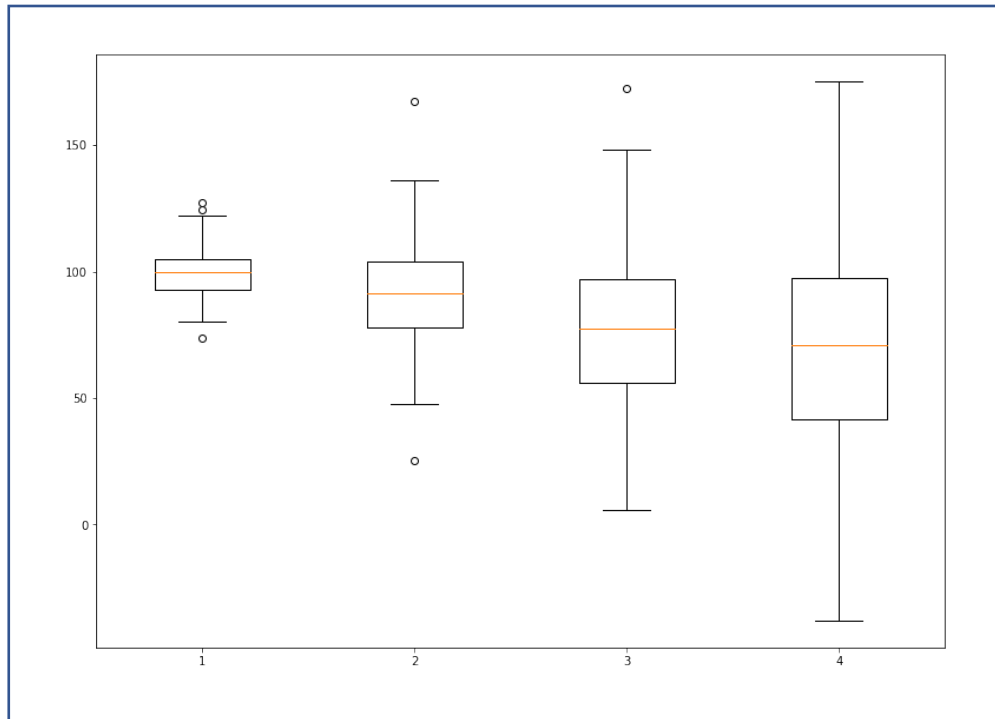
# Gerando 100 amostras aleatorias a partir de uma normal
idades = [np.random.normal(loc= 25, size = 100)]

# tituto para grafico
plt.title('Idades de um grupo', fontsize=20)
# titulo x axis
plt.xlabel('Idade', fontsize=15)
# titulo y axis
plt.ylabel('Frequência Absoluta', fontsize=15)
# fonte
plt.tick_params(labelsize=15)
# Configurando hist
plt.hist(idades,
        bins = 10,
        rwidth=0.9,
        color='#bdc9e1',
        alpha=0.7,
        edgecolor='black')

plt.show()
```

Visualização de Dados

Boxplot: Diagrama de caixa, diagrama de extremos e quartis, boxplot ou box plot é uma ferramenta gráfica para representar a variação de dados observados de uma variável numérica por meio de quartis.



Fonte: Autor próprio, 2022.

```
# Importando livrarias
import matplotlib.pyplot as plt
import numpy as np

# Definido random seed 42
np.random.seed(42)

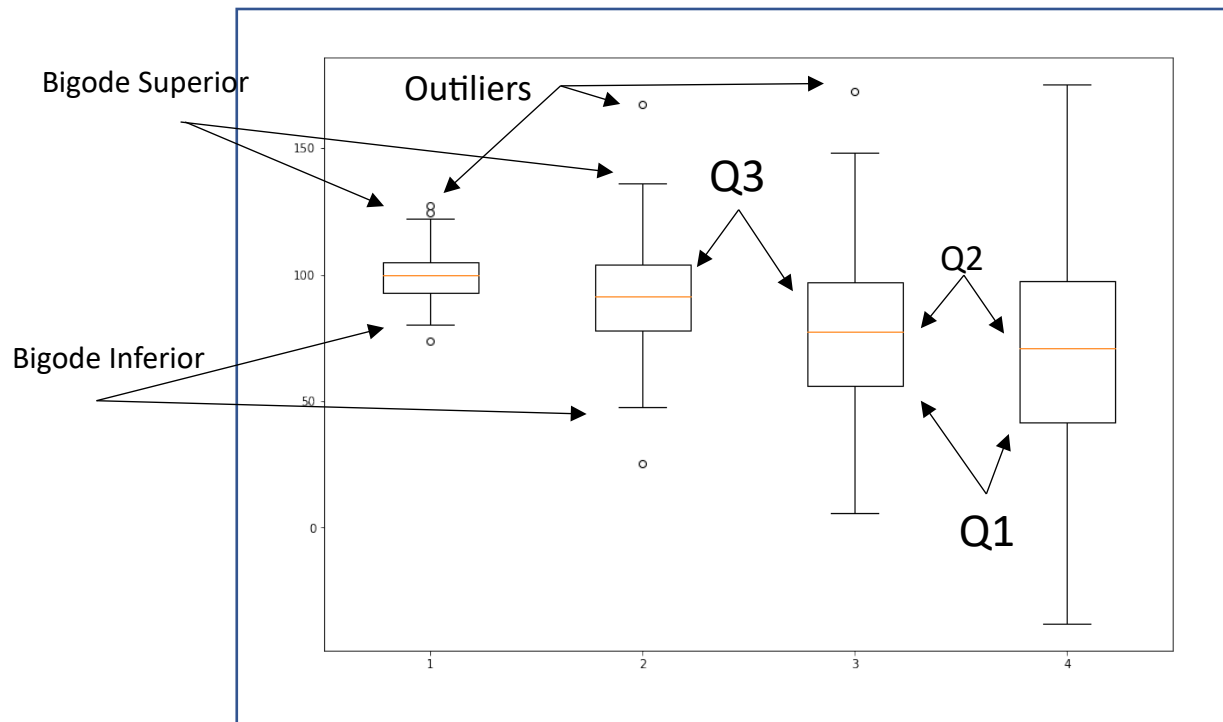
data_1 = np.random.normal(100, 10, 200)
data_2 = np.random.normal(90, 20, 200)
data_3 = np.random.normal(80, 30, 200)
data_4 = np.random.normal(70, 40, 200)
data = [data_1, data_2, data_3, data_4]

fig = plt.figure(figsize=(10, 7))
ax = fig.add_axes([0, 0, 1, 1])
bp = ax.boxplot(data)

plt.show()
```

Visualização de Dados

Boxplot: Diagrama de caixa, diagrama de extremos e quartis, boxplot ou box plot é uma ferramenta gráfica para representar a variação de dados observados de uma variável numérica por meio de quartis.



Fonte: Autor próprio, 2022.

```
# Importando livrarias
import matplotlib.pyplot as plt
import numpy as np

# Definido random seed 42
np.random.seed(42)

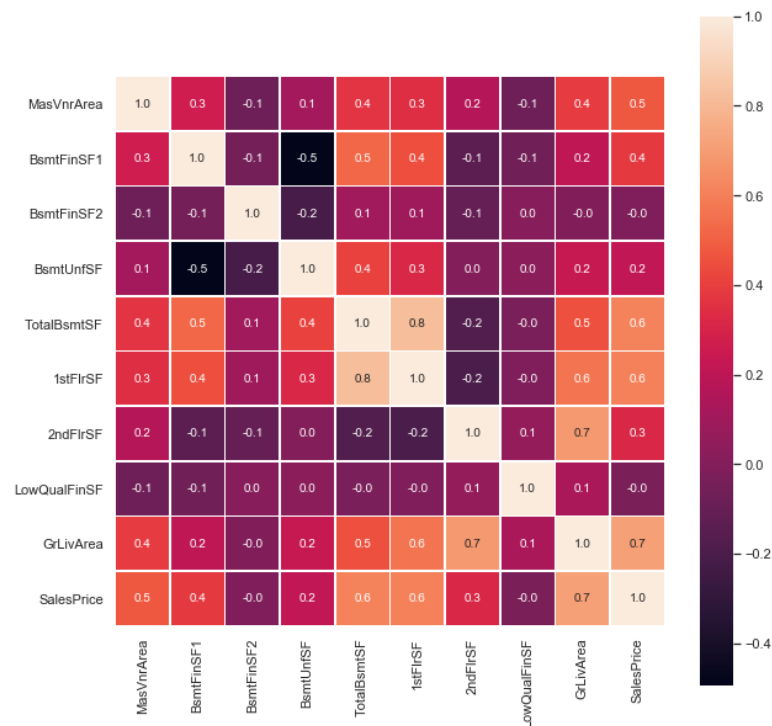
data_1 = np.random.normal(100, 10, 200)
data_2 = np.random.normal(90, 20, 200)
data_3 = np.random.normal(80, 30, 200)
data_4 = np.random.normal(70, 40, 200)
data = [data_1, data_2, data_3, data_4]

fig = plt.figure(figsize=(10, 7))
ax = fig.add_axes([0, 0, 1, 1])
bp = ax.boxplot(data)

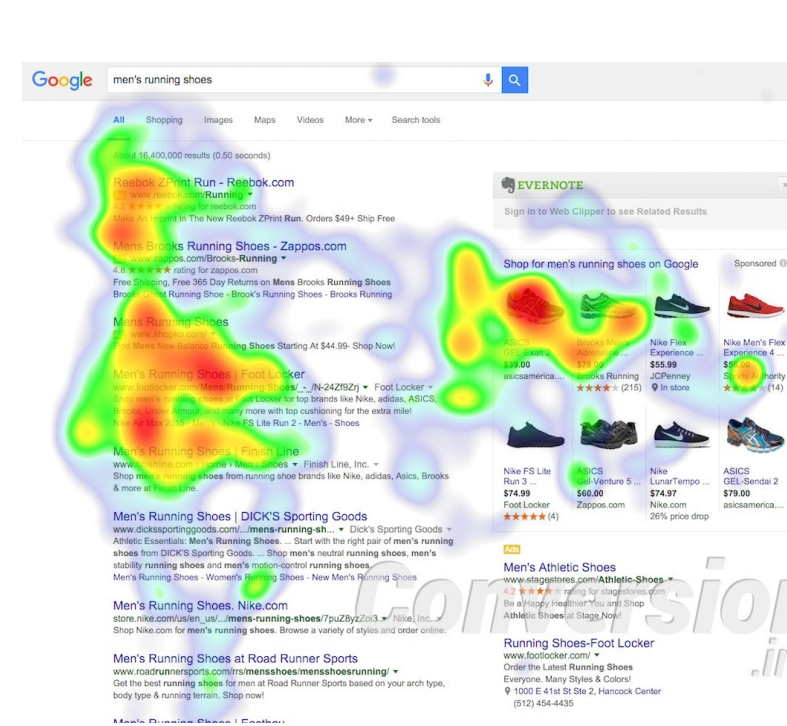
plt.show()
```


Visualização de Dados

Heatmap: O heatmap, ou mapa de calor, na tradução do inglês, é uma representação gráfica que mostra em quais pontos de um site, ou blog, houve maior atividade por parte do usuário. Essa presença é resultado de maiores interações com o mouse, considerando também cliques e rolagem da página.



Fonte: Machine Learning: Predicting House Prices, 2020.

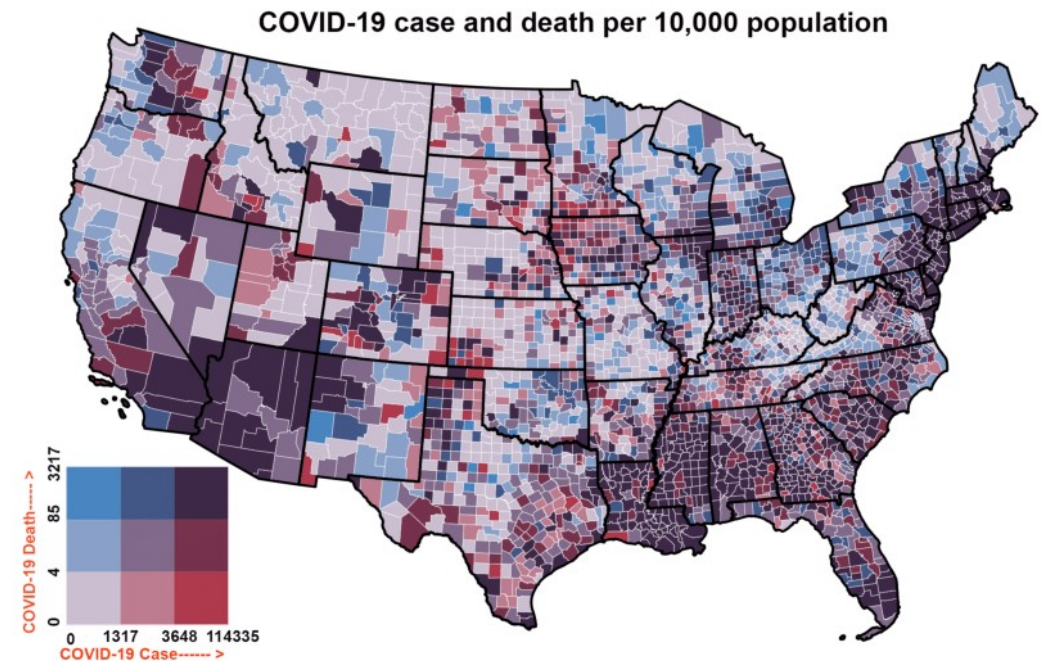


Fonte: F-Patterns No More: How People View Google & Bing Search Results, 2020.

Visualização de Dados

Mapas chloropleth: Um mapa coroplético representa normalmente uma superfície estatística por meio de áreas simbolizadas com cores, sombreamentos ou padrões de acordo com uma escala que representa a proporcionalidade da variável estatística em causa, como por exemplo a densidade populacional ou o rendimento per capita.

Bivariate choropleth map demonstrates the county wise distribution (per 10,000 population) of COVID-19 cases and deaths from 22 January to 26 July 2020.



Visualização de Dados

Dispersão pairplot: Ótimo para se checar se existe alguma separabilidade ou diferença estatística entre os atributos dos objetos estudados.

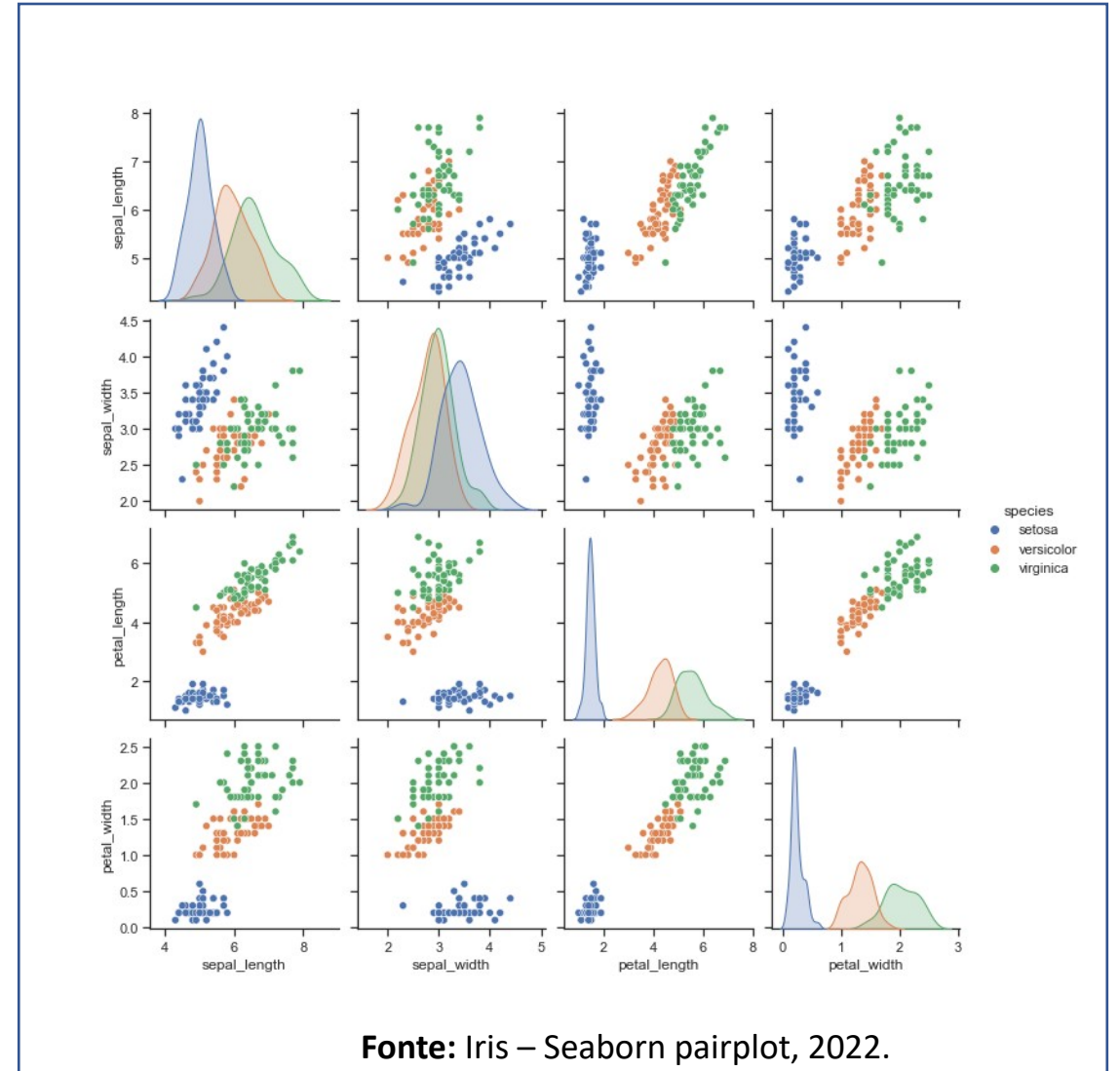
Seaborn pairplot

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.set(style="ticks", color_codes=True)
iris = sns.load_dataset("iris")
g = sns.pairplot(iris, diag_kind="kde", hue = "species")

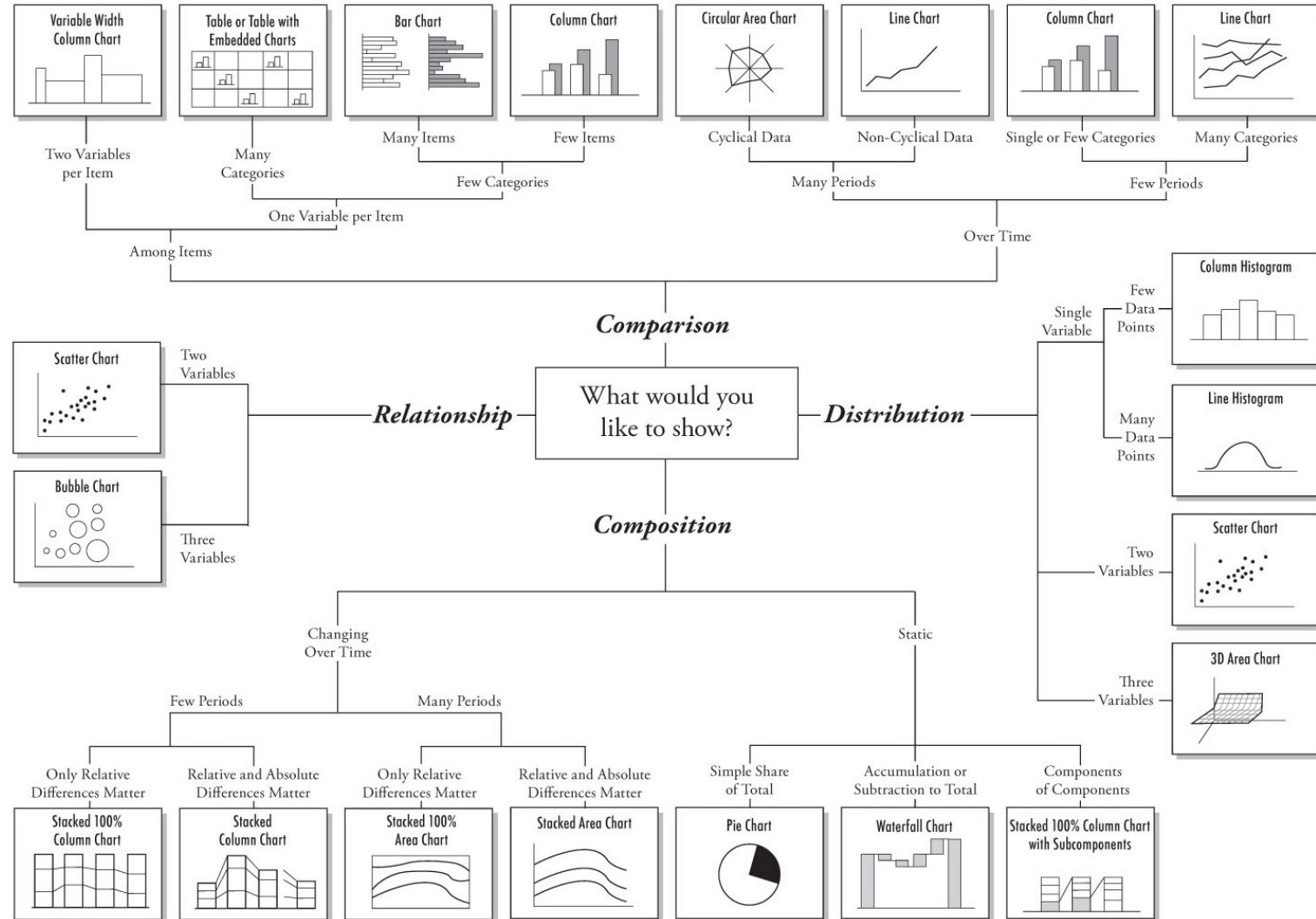
plt.show()
```

Fonte: Autor próprio, 2022.



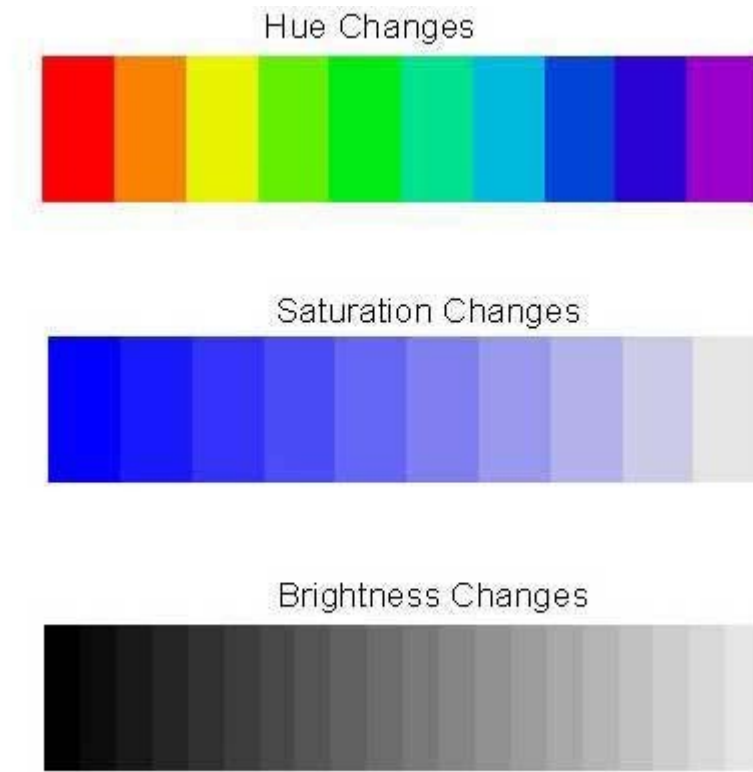
Visualização de Dados

Chart Suggestions—A Thought-Starter



❑ Usar as cores com sensatez

Cores para categorias: Não use mais de 5-8 cores de uma só vez. Dica: Cores para dados ordinais: Varie a luminância e a saturação.



Na dúvida, use o site “Color Brewer”.

Matplotlib



Google Colab

https://github.com/GabrielFonsecaNunes/data-science-with-python/blob/master/Aula%202/Notebooks/Notebook_1_Matplotlib_Plots_e_Graficos.ipynb

Visualização de Dados

❑ Seaborn

Seaborn é uma biblioteca de visualização de dados Python baseada em matplotlib. Ele fornece uma interface de alto nível para desenhar gráficos estatísticos atrativos e informativos.



seaborn

Visualização de Dados

❑ Seaborn

Instalando Seaborn

```
>> pip install seaborn
```

Importando Seaborn

```
import seaborn as sns
```

Para instalar o matplotlib de “**pip install seaborn**” no terminal. Dentro do google colab o seaborn já vem por padrão. Para se utilizar o próprio notebook como **backend** dos seus gráficos deve ser utilizado “**%matplotlib inline**” após a importação do matplotlib.

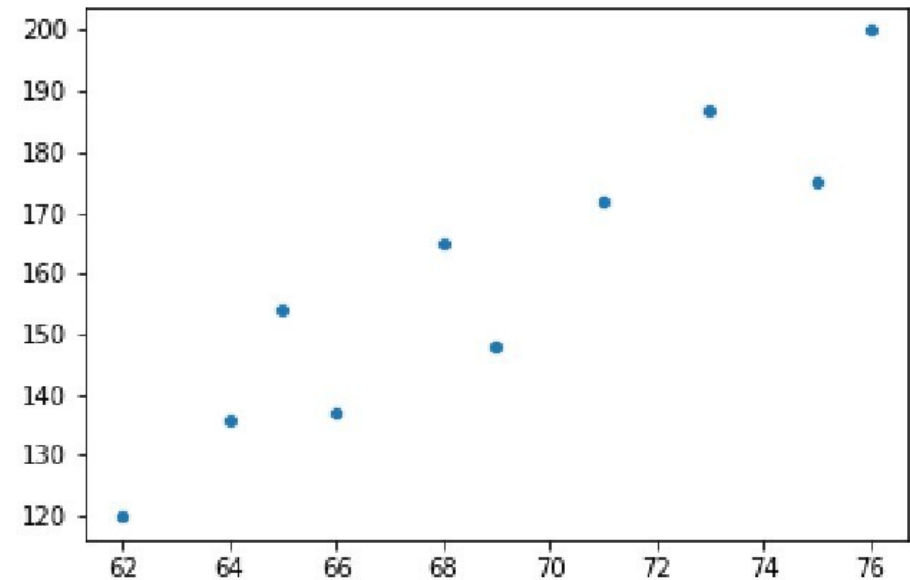
Visualização de Dados

❑ Seaborn

```
import seaborn as sns
import matplotlib.pyplot as plt

height = [62, 64, 69, 75, 66,
          68, 65, 71, 76, 73]
weight = [120, 136, 148, 175, 137,
          165, 154, 172, 200, 187]

sns.scatterplot(x=height, y=weight)
plt.show()
```



Visualização de Dados

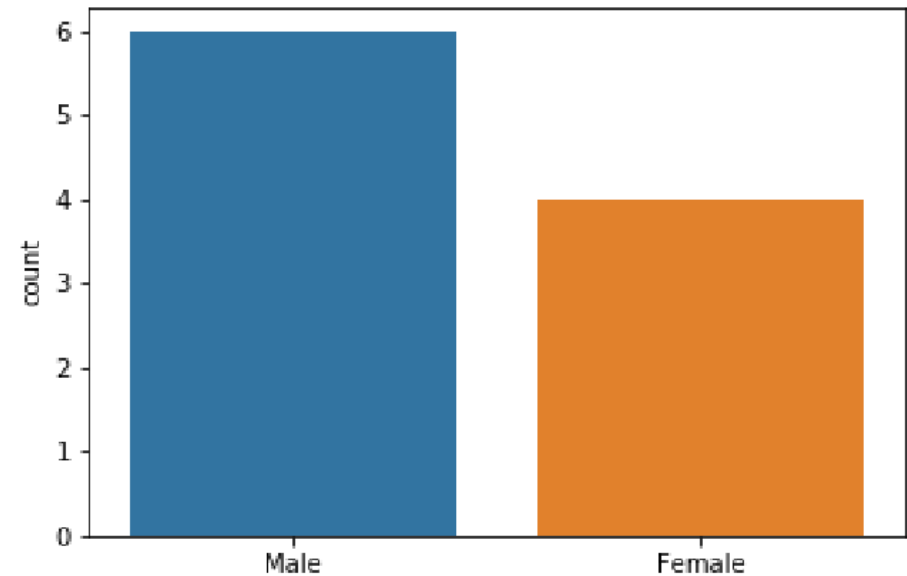
❑ Seaborn

```
import seaborn as sns
import matplotlib.pyplot as plt

gender = ["Female", "Female",
          "Female", "Female",
          "Male", "Male", "Male",
          "Male", "Male", "Male"]

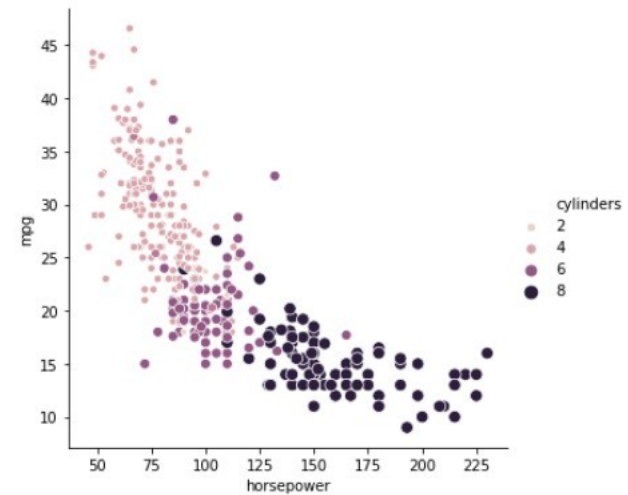
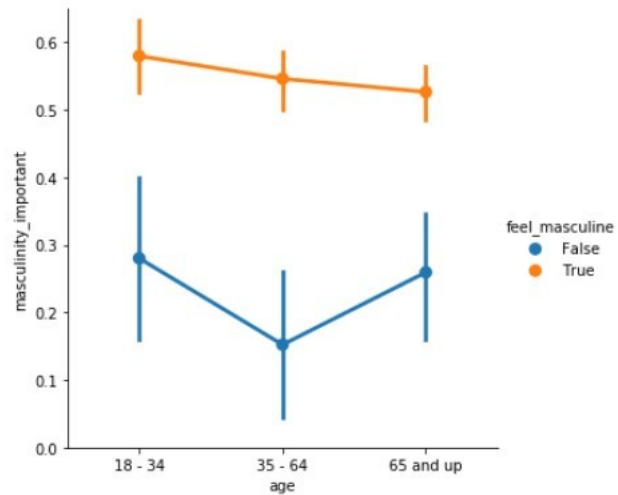
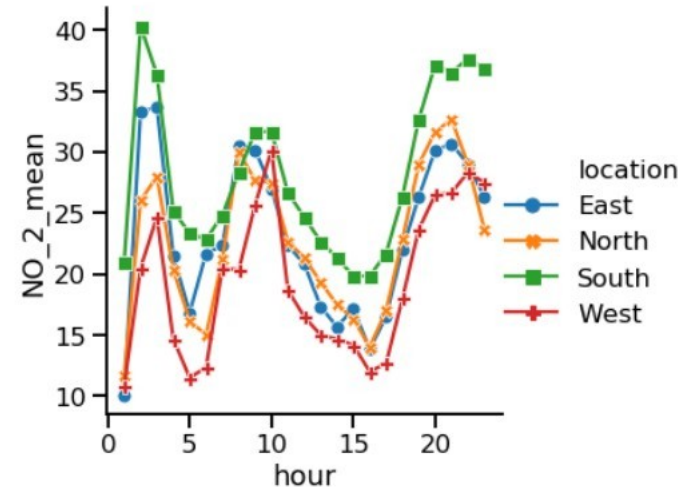
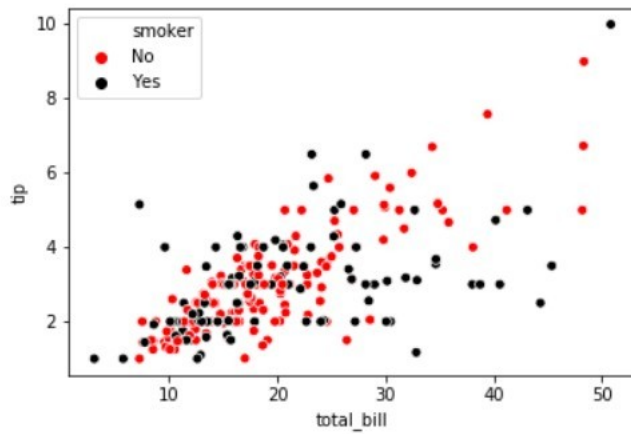
sns.countplot(x=gender)

plt.show()
```



Visualização de Dados

Seaborn

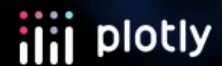


Seaborn



Google Colab

https://github.com/GabrielFonsecaNunes/data-science-with-python/blob/master/Aula%202/Notebooks/Notebook_2_Seaborn.ipynb



DASH ENTERPRISE ▾

DOCS ▾

GALLERIES

COMPANY ▾

PRICING


DEMO DASH

Low-Code Data Apps

Dash Enterprise is the premier platform for
building, scaling, and deploying data apps in Python.

LEARN ABOUT DASH

||| CISCO

 NVIDIA

intuit

 Shell

Colgate

AMGEN

Plotly



Google Colab

https://github.com/GabrielFonsecaNunes/data-science-with-python/blob/master/Aula%202/Notebooks/Notebook_3_plotly-tutorial-for-beginners.ipynb

Perguntas ?

Obrigado!

Referências Bibliográficas

Matplotlib Tutorial, 2022. Disponível em: <https://matplotlib.org/stable/tutorials/index.html>. Acesso em 15 jul. 2022.

User guide and tutorial - Seaborn tutorial, 2022. Disponível em: <https://seaborn.pydata.org/tutorial.html>. Acesso em 15 jul. 2022.

Disponível em: . Acesso em 15 jul. 2022.

Storytelling com dados: Um guia sobre visualização de dados para profissionais de negócios. Disponível em: <https://www.storytellingwithdata.com/books>. Acesso em 15 jul. 2022.

NYC Data Science Academy, Machine Learning: Predicting House Prices, 2020. Disponível em: <https://nycdatascience.com/blog/student-works/ames-housing-predicting-house-prices-with-machine-learning/>. Acesso em 15 jul. 2022.