

PYTHON

Professor: Eduardo Inocencio

GRÁFICOS EM PYTHON?

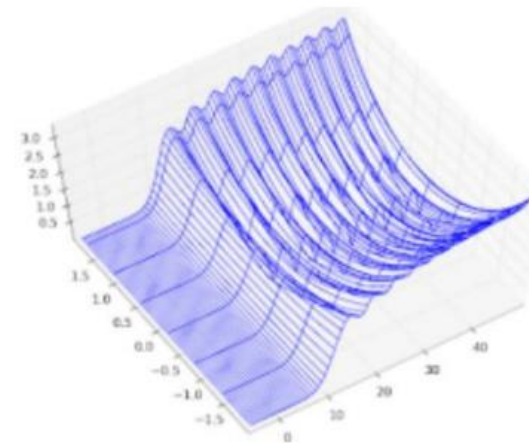
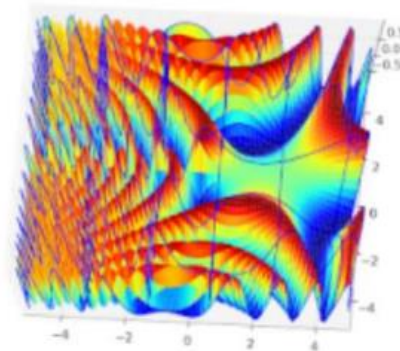
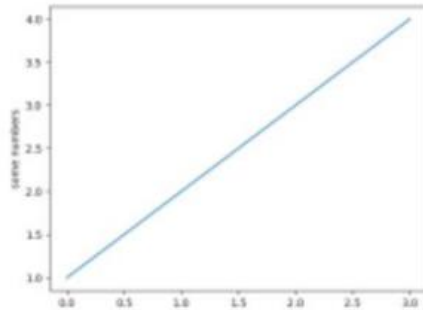
Existem hoje diversas bibliotecas em Python para criação de gráficos. Dentre as várias possibilidades, exploraremos aqui as duas mais comuns: Matplotlib e Seaborn. Em cada biblioteca, considerando os principais tipos de dados, iremos apresentar alguns dos tipos de gráficos mais comuns.

matplotlib

seaborn

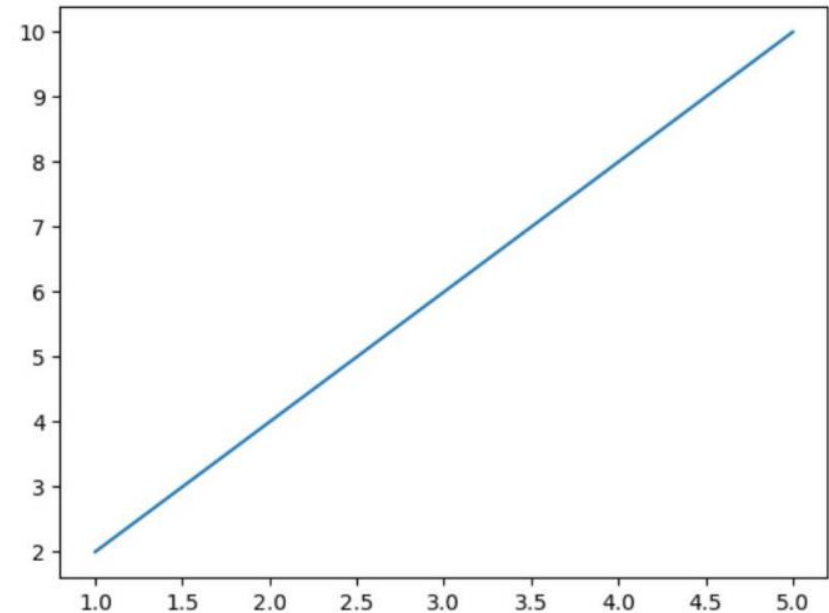
MATPLOTLIB — PRIMEIROS PASSOS

Matplotlib é a principal biblioteca para geração de gráficos em Python. A biblioteca possui uma infinidade de possibilidades, sendo possível criarmos desde gráficos simples até outros mais elaborados.



IMPORTANDO A BIBLIOTECA & CRIANDO O PRIMEIRO GRÁFICO

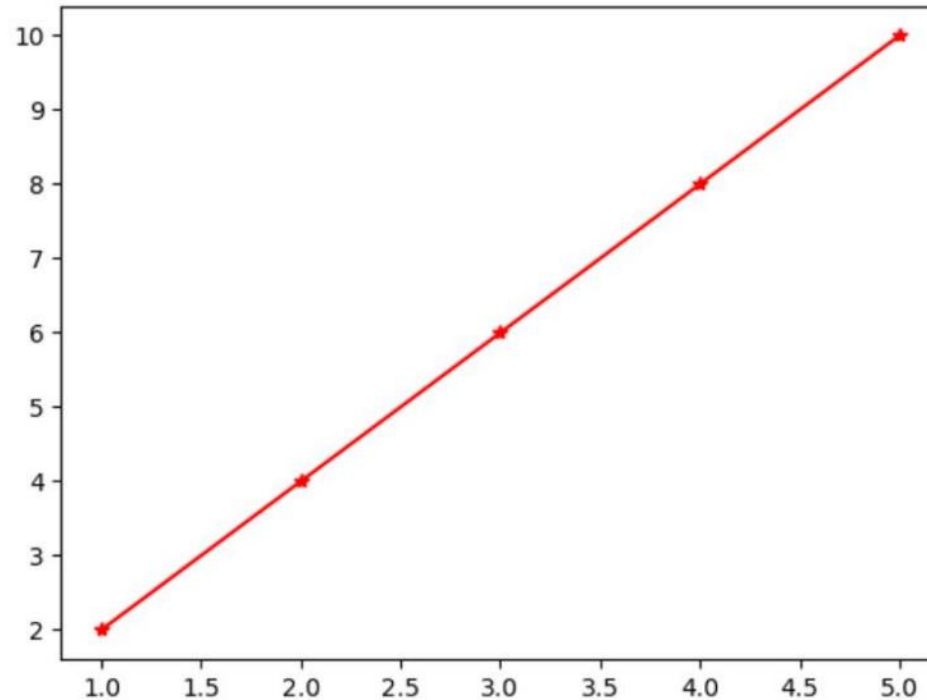
```
import matplotlib.pyplot as plt
import numpy as np
x = np.array ([1,2,3,4,5])
y = np.array ([2,4,6,8,10]) plt.plot(x,y)
```



MODIFICANDO ALGUNS PARÂMETROS

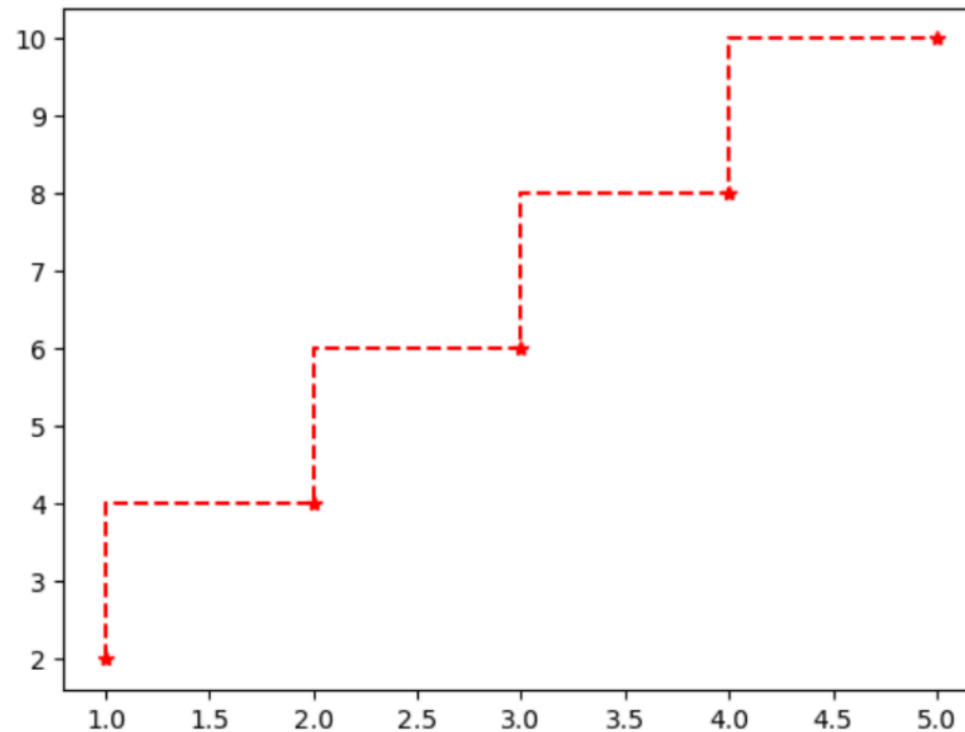
MODIFICANDO O TIPO DE MARCADOR

```
plt.plot (x,y, marker = '*', c = 'red')
```



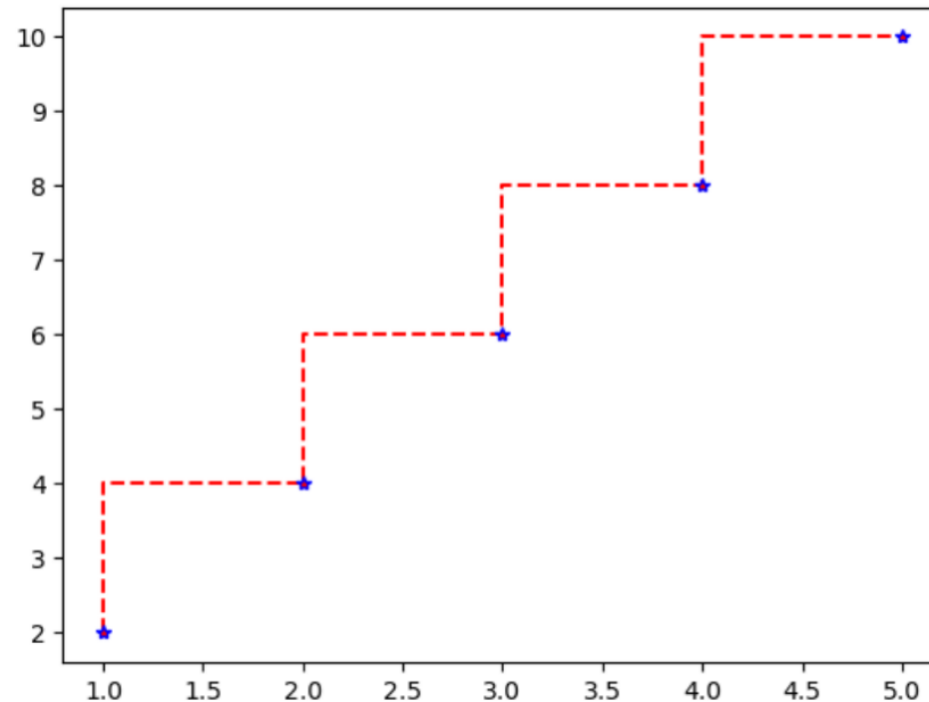
MUDANDO COMO OS PONTOS SÃO LIGADOS E O TIPO DE LINHA

```
plt.plot (x,y, marker = '*', c = 'red', ds = 'steps', ls = '--')
```



ALTERANDO AS CORES DOS MARCADORES

```
plt.plot (x,y, marker = '*', c = 'red', ds = 'steps', ls = '--', mec = 'blue')
```



ADICIONANDO OUTROS ELEMENTOS

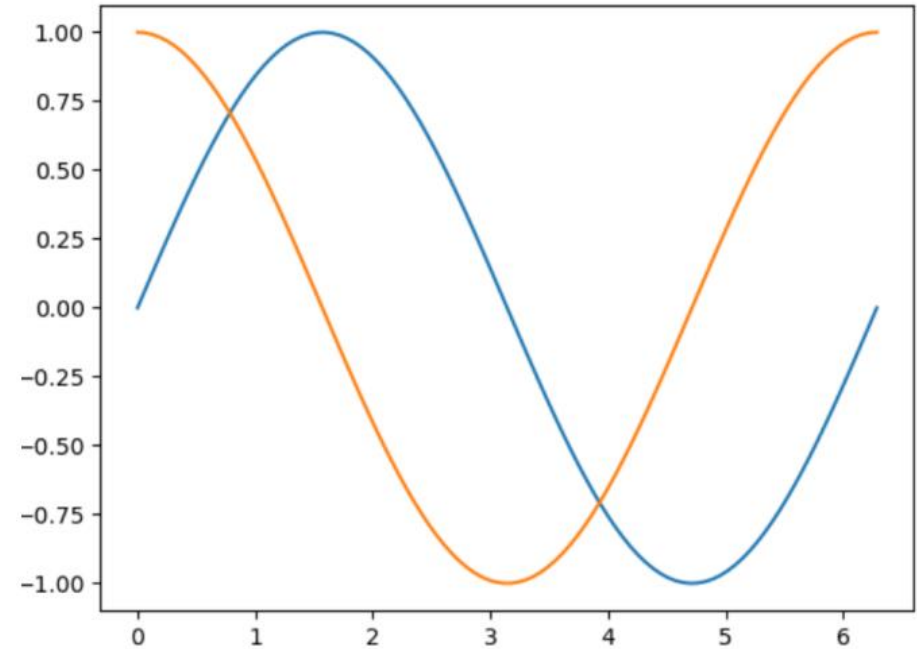
```
plt.plot(x,y)
plt.xlabel('Valores de X', fontsize = 12)
plt.ylabel('Valores de Y', fontsize = 12)
plt.title ('Título do gráfico', fontsize = 15, c='purple')
plt.xticks(fontsize = 8, c = 'blue')
plt.yticks(fontsize = 8, c = 'orange')
```



MÚTIPLAS CURVAS NO MECANISMO GRÁFICO

Neste exemplo, vamos plotar duas curvas em um único gráfico.

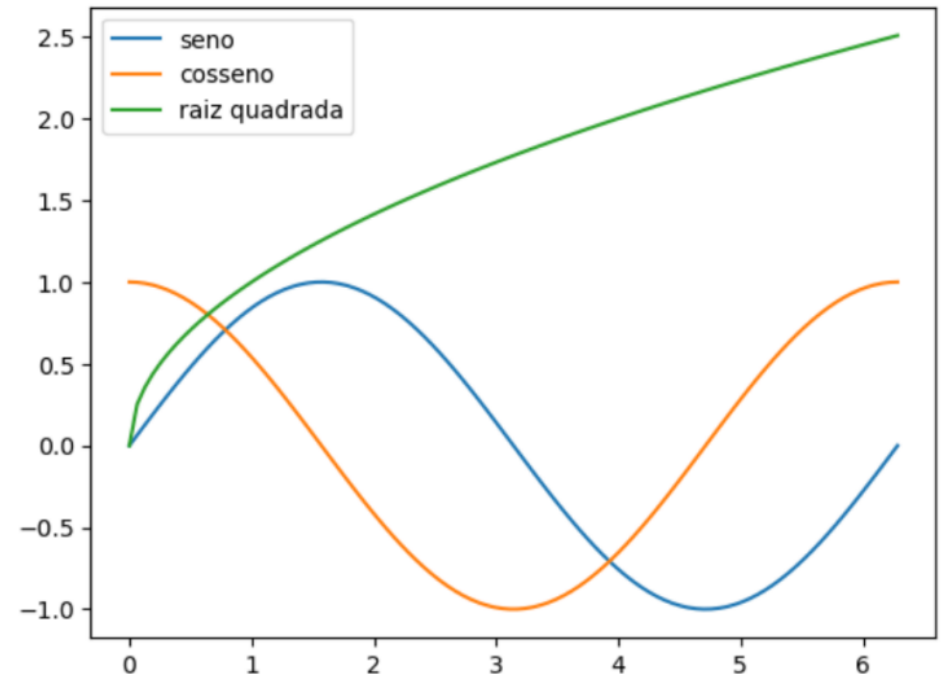
```
x = np.linspace (0,2*np.pi,100)
y = np.sin(x)
z = np.cos(x)
plt.plot (x,y)
plt.plot (x,z)
```



MÚTIPLAS CURVAS NO MECANISMO GRÁFICO

Neste exemplo, vamos plotar duas curvas em um único gráfico.

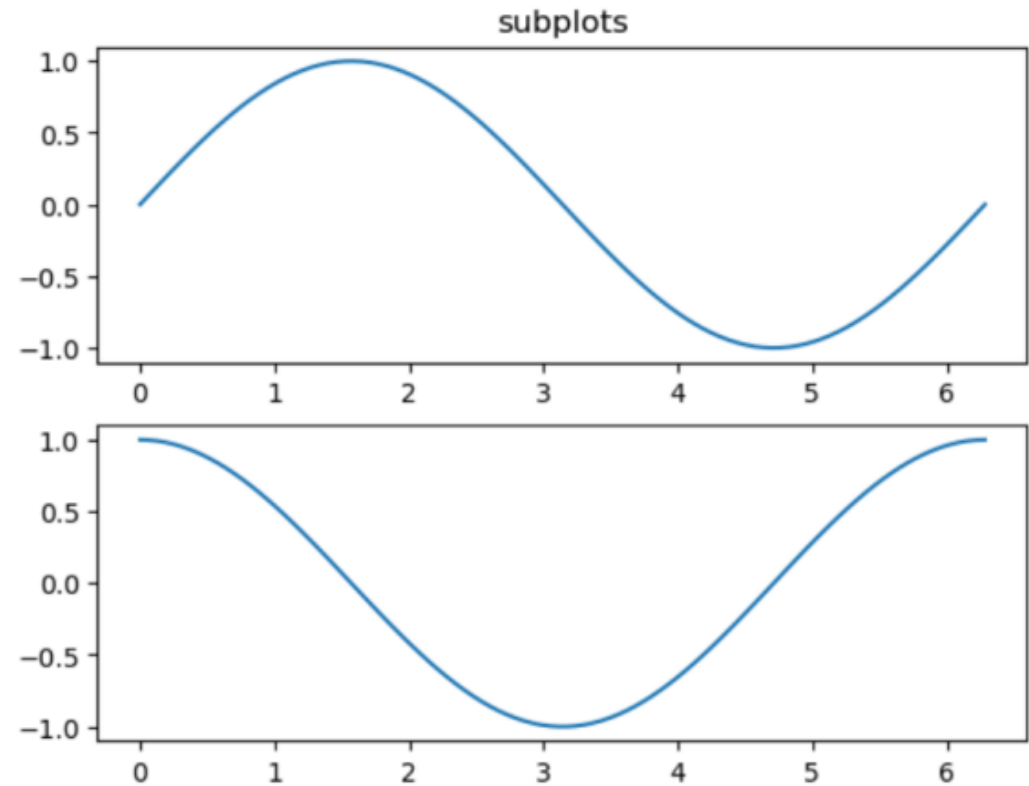
```
plt.plot(x,y, label = 'seno')  
plt.plot (x,z,label='cosseno')  
plt.plot (x,np.sqrt(x), label='raiz quadrada')  
plt.legend()
```



MÚLTIPLOS SUBGRÁFICOS

Dois gráficos na horizontal (linha)

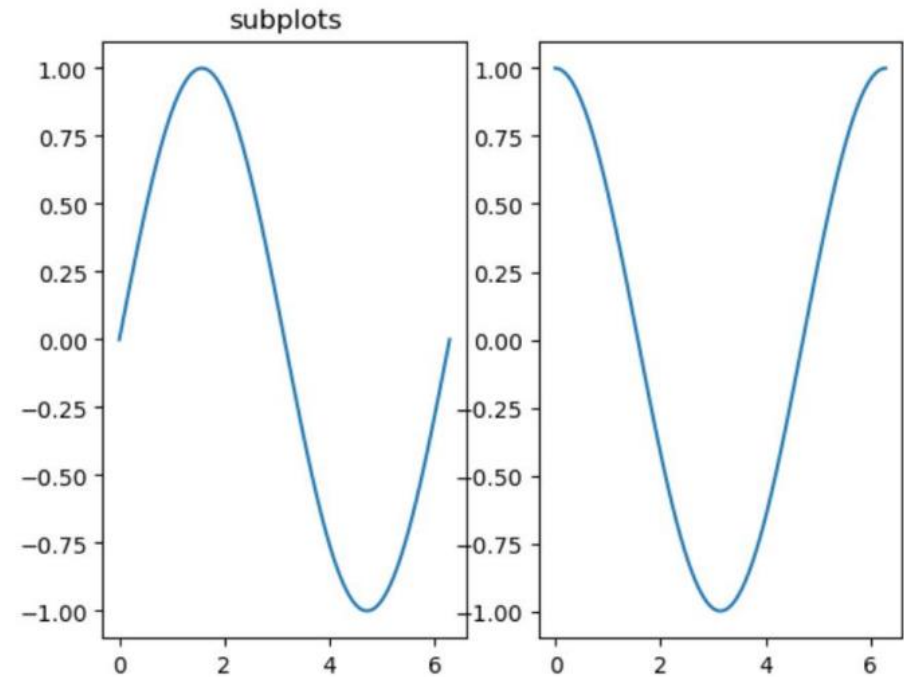
```
plt.subplot(2,1,1)
plt.plot(x,y)
plt.title('subplots')
plt.subplot(2,1,2)
plt.plot(x,z)
```



MÚLTIPLOS SUBGRÁFICOS

Dois gráficos na vertical (coluna)

```
plt.subplot(1,2,1)
plt.plot(x,y)
plt.title('subplots')
plt.subplot(1,2,2)
plt.plot(x,z)
```



OUTROS GRÁFICOS

Existem diversos tipos de gráficos que podem ser feitos com a biblioteca Matplotlib. Alguns serão explorados a seguir (tanto em Matplotlib, quanto em Seaborn), mas dentre as possibilidades, podemos citar:

- Gráficos lineares;
- Gráficos de dispersão;
- Gráficos de barras;
- Histogramas;
- Boxplots;
- Imagens;
- Tridimensionais;
- Gráficos de fluxo de campos;
- Gráficos polares.

EXERCÍCIO - VISUALIZAÇÃO DE DADOS - MATPLOTLIB

1. Crie dois arrays `x` e `y` com as listas `[-3,-2.5,-2,-1,0,1,2,2.5,3]` e `[-27,-15.62,-8,-1,0,1,8,15.62,27]`.
2. Use a função `plot()` e faça um gráfico de `x` e `y`.
3. Use a função `xlabel()` para adicionar o nome “Valor de X” ao eixo `x`.
4. Deixe com fonte de tamanho 15.
5. Use a função `ylabel()` para adicionar o nome “Valor de Y” ao eixo `y`.
6. Deixe com fonte de tamanho 15. 5.
7. Use a função `legend()` para adicionar o nome “Função Cúbica” ao gráfico.
8. Deixe com fonte de tamanho 20.
9. Rode novamente seu gráfico, mas mude agora a linha para a cor magenta.

SEABORN

Professor: Eduardo Inocencio

GRÁFICOS MAIS ELABORADOS

Seaborn é uma biblioteca baseada em Matplotlib que traz uma interface mais atraente para os gráficos, além de facilitar a modificação de parâmetros default do Matplotlib e permite a utilização de dataframes.

```
import seaborn as sns
```

DATASETS EMBUTIDOS

O Seaborn possui diversos dados embutidos que podem ser utilizados para estudar a própria biblioteca. Para sabermos quais os datasets estão disponíveis, podemos utilizar a função `get_dataset_names()`.

```
sns.get_dataset_names()
```

```
['anagrams',  
'anscombe',  
'attention',  
'brain_networks',  
'car_crashes',  
'diamonds',  
'dots',  
'dowjones',  
'exercise',  
'flights',  
'fmri',  
'geyser',  
'glue',  
'healthexp',  
'iris',  
'mpg',  
'penguins',  
'planets',  
'seaice',  
'taxis',  
'tips',  
'titanic']
```

CRIANDO UM DATAFRAME COM O DATASET ÍRIS.

```
iris = sns.load_dataset('iris')  
iris.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

CRIANDO UM DATAFRAME COM O DATASET PLANETS.

```
planetas = sns.load_dataset('planets')  
planetas.head()
```

	method	number	orbital_period	mass	distance	year
0	Radial Velocity	1	269.300	7.10	77.40	2006
1	Radial Velocity	1	874.774	2.21	56.95	2008
2	Radial Velocity	1	763.000	2.60	19.84	2011
3	Radial Velocity	1	326.030	19.40	110.62	2007
4	Radial Velocity	1	516.220	10.50	119.47	2009

AJUSTANDO OS GRÁFICOS

○ Seaborn possui a função `set ()` que permite ajustar estilos específicos para os gráficos, sem a necessidade de ajustar manualmente todos os argumentos.

```
plt.plot (x,y)
```

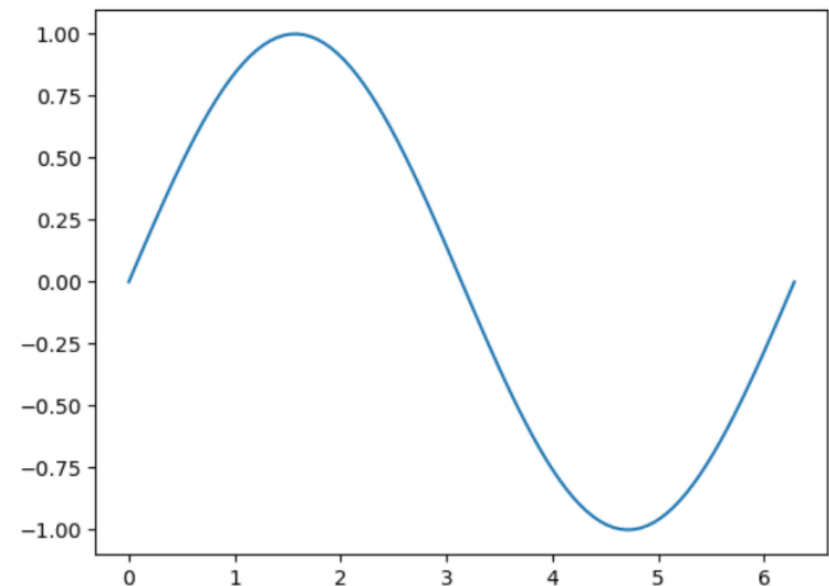
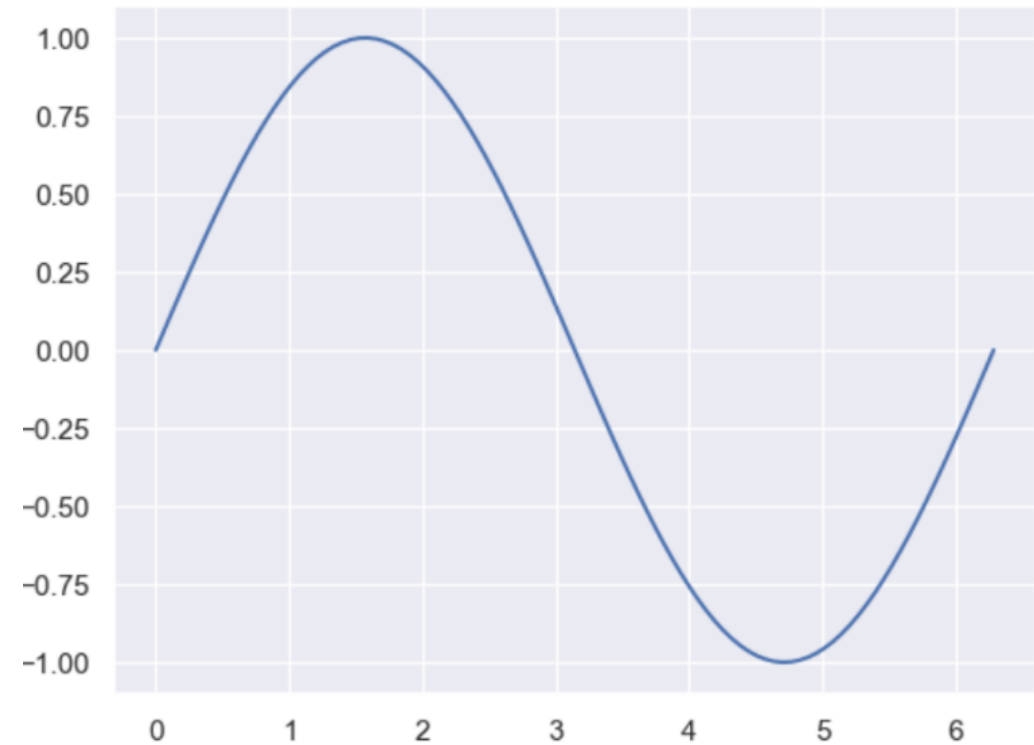


GRÁFICO COM A CONFIGURAÇÃO AJUSTADA COM A FUNÇÃO SET ().

```
sns.set()  
plt.plot (x,y)
```



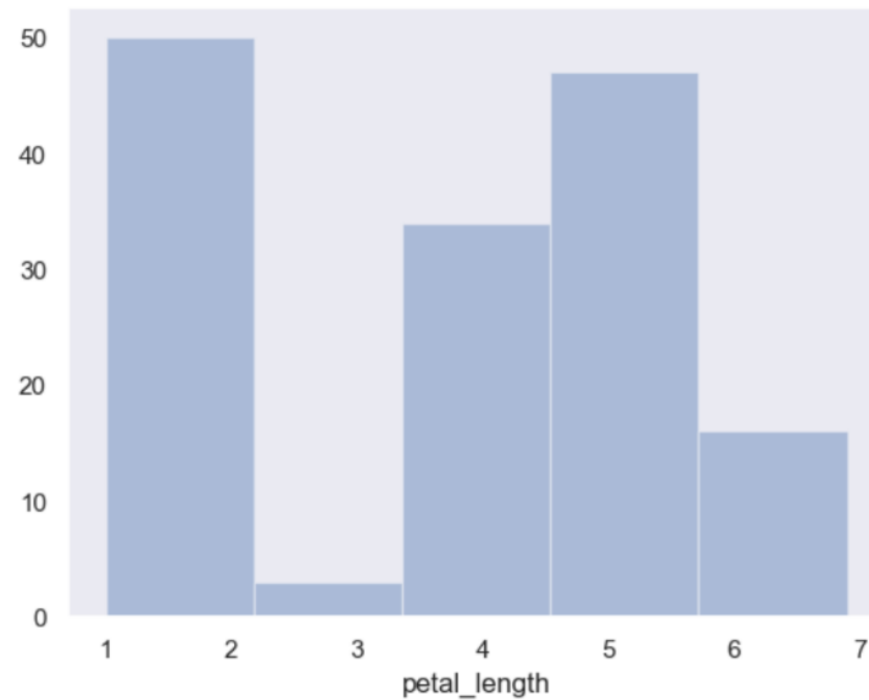
GRÁFICOS ESTATÍSTICOS

Outra vantagem do Seborn é a possibilidade de criar gráficos estatísticos como, por exemplo, gráficos que estimem a função de densidade de probabilidade. Entre as possibilidades de gráficos estatísticos, podemos criar gráficos estatísticos com as funções `distplot()` – univariada, `jointplot()` – multivariada.

Variando os parâmetros dentro dessas funções, podemos gerar visualização de histogramas, gráficos de hexágonos, dispersão e funções de distribuição.

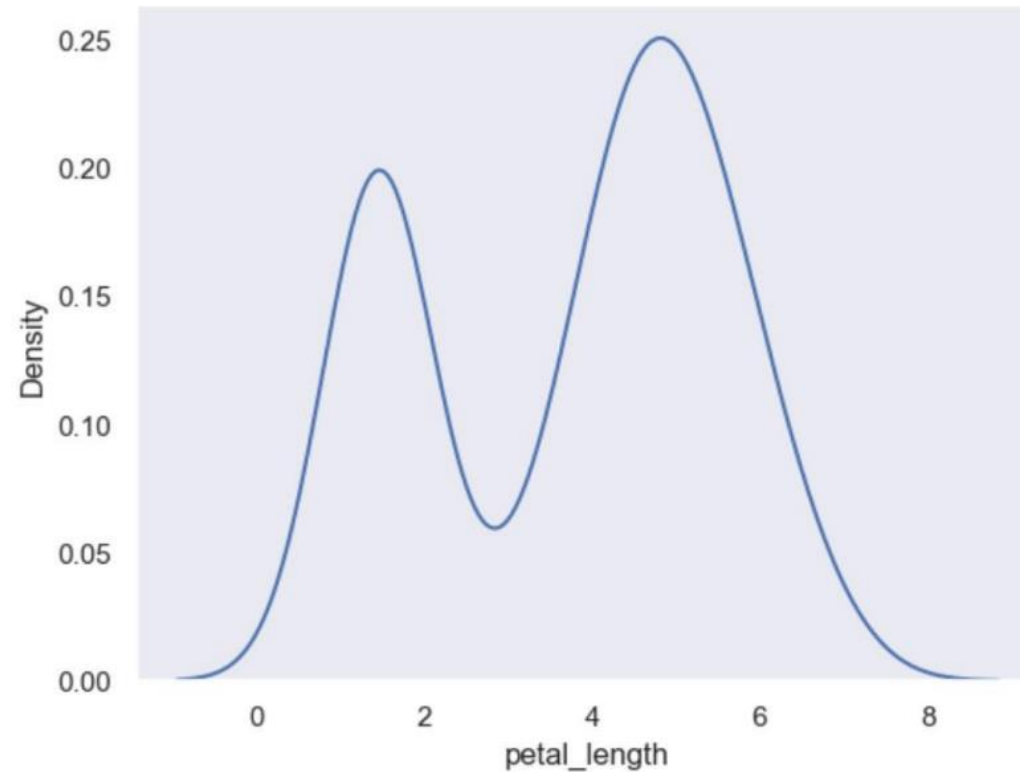
DISTPLOT (UNIVARIADA)

```
sns.distplot(iris['petal_length'], kde=False)
```



DISTPLOT (UNIVARIADA)

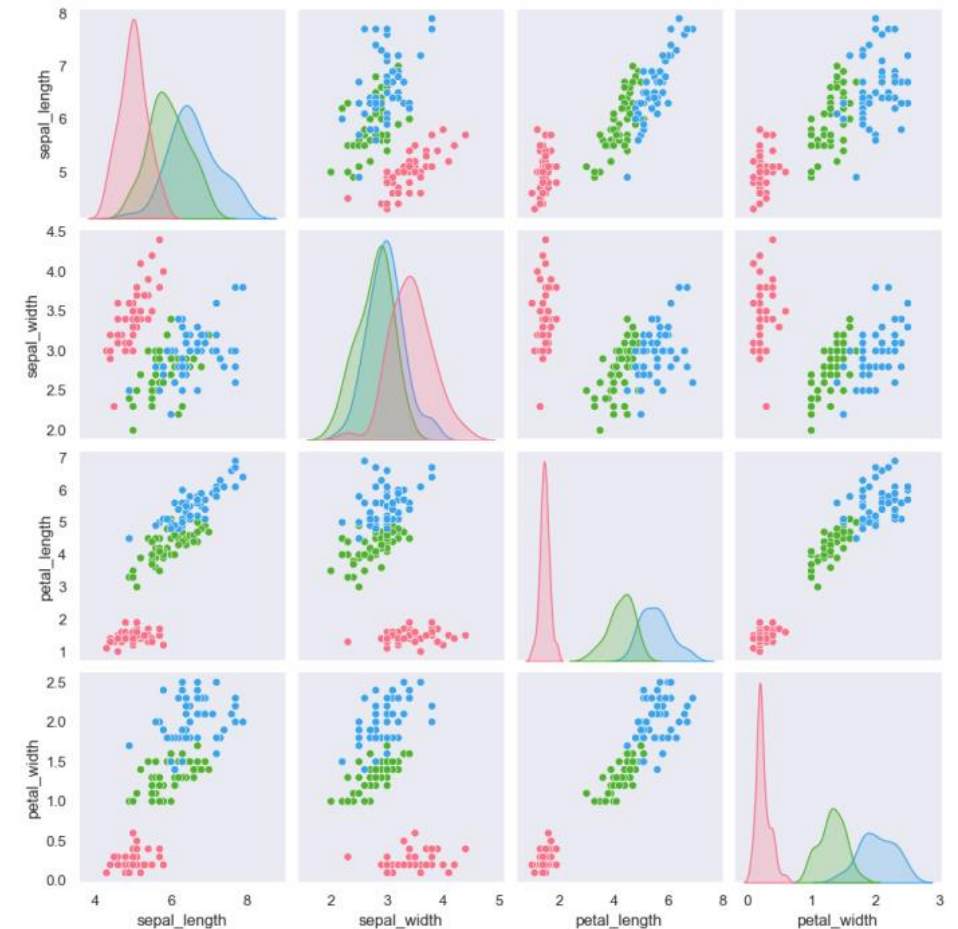
```
sns.distplot(iris['petal_length'], hist=False)
```



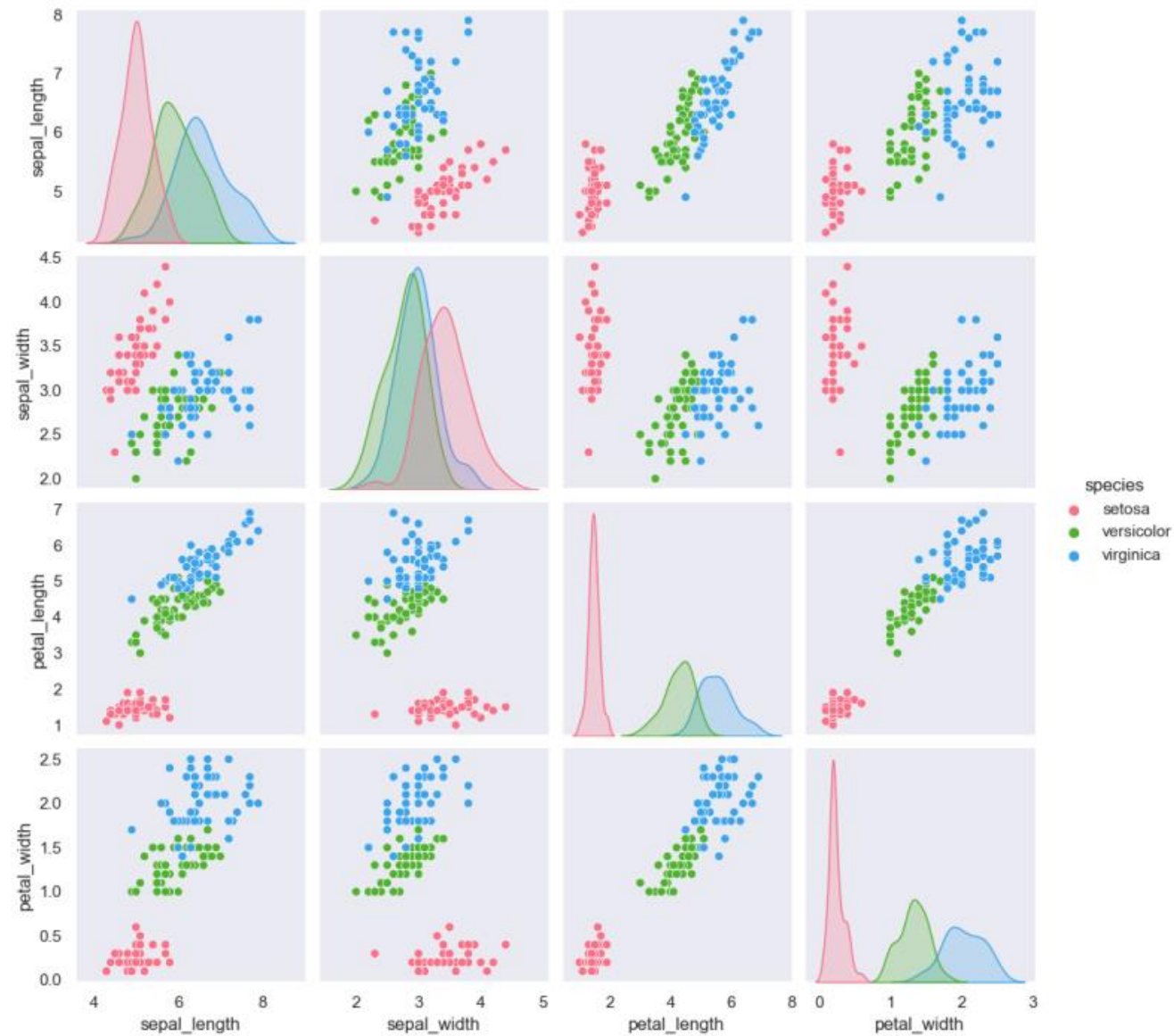
PAIRPLOT (BIVARIADA)

A função `pairplot()` é outra função muito útil quando usamos os gráficos como recurso exploratório.

Essa função exibe a relação 2 a 2 entre as variáveis, de modo que facilite a busca por correlações entre as variáveis e um conjunto de dados. Geralmente exibe dados numéricos, embora dados categóricos possam ser usados para discriminar os dados.



```
sns.pairplot(iris,hue='species',diag_kind='kde',kind='scatter',palette='hus_l')
plt.savefig('pairplot_iris.png')
```



SEABORN — DADOS CATEGÓRICOS

Os gráficos vistos até o momento são apenas com dados numéricos. E os dados categóricos? Como visualizá-los?

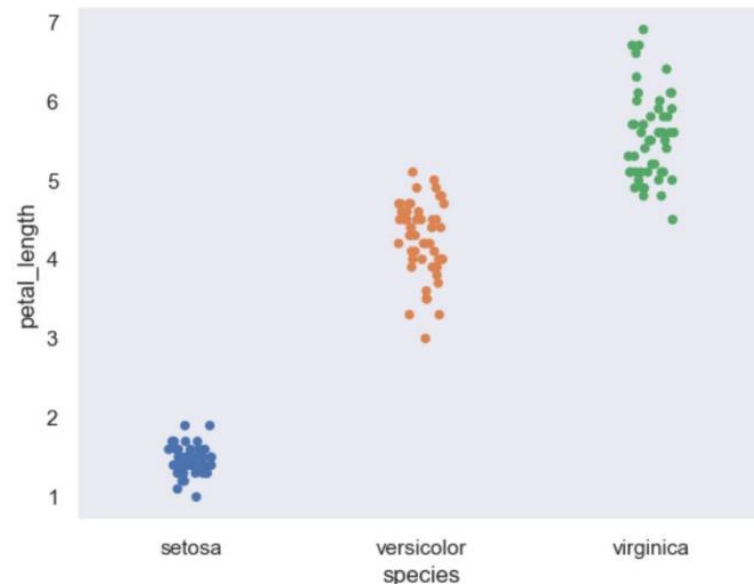
○ Seaborn possui diversos gráficos que ajudam na exploração de dados categóricos

- Striplot
- Swarmplot
- Boxplot
- Violinplot
- Barplot
- Pointplot
- Etc. :

SEABORN — DADOS CATEGÓRICOS

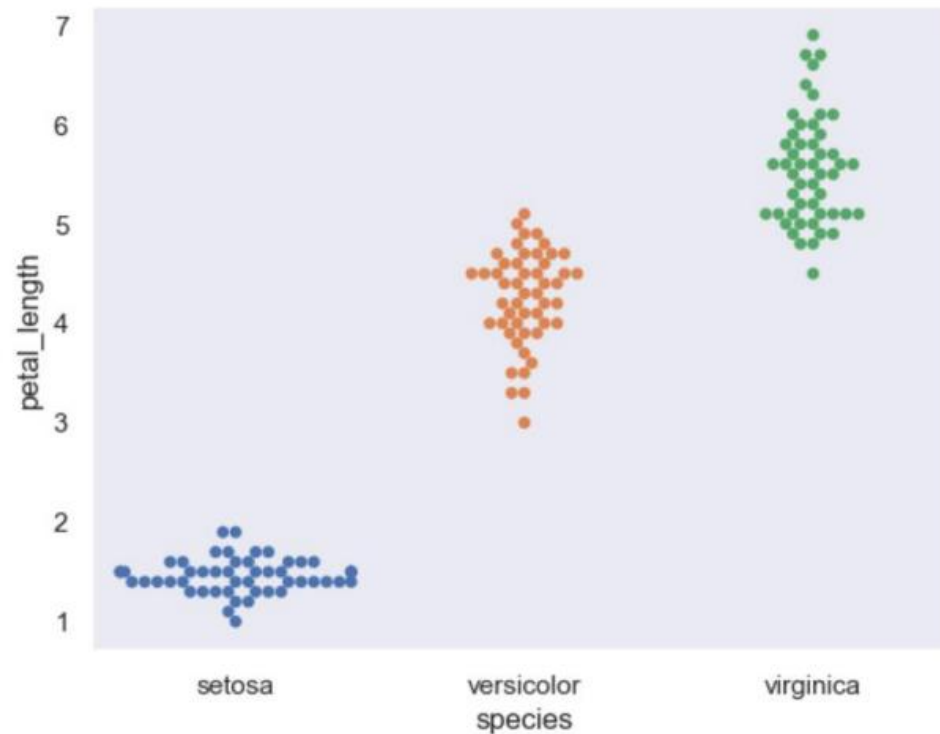
Os gráficos `stripplot()` e `swarmplot()`, são gráficos em que analisamos a dispersão de uma variável numérica e outra categórica.

```
sns.stripplot(x='species', y='petal_length', data=iris)
```



SEABORN — DADOS CATEGÓRICOS

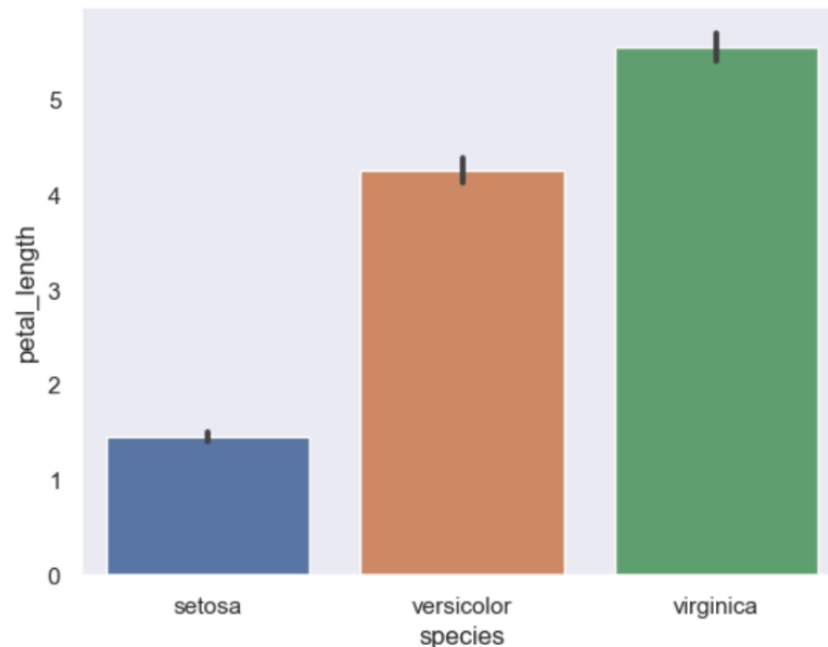
```
sns.swarmplot(x='species', y='petal_length', data=iris)
```



SEABORN – DADOS CATEGÓRICOS

Os gráficos `barplot()` e `boxplot()`, são gráficos e que analisamos a relação entre dados contínuos e categóricos.

```
sns.barplot(x='species', y='petal_length', data=iris)
```



EXERCÍCIO - VISUALIZAÇÃO DE DADOS - SEABORN

1. Importe a biblioteca Seaborn.
2. Use a função `load_dataset()` e coloque o dataset `exercises` num dataframe chamado `exercicios`.
3. Use a função `set_style()` e mude para 'dark'.
4. Faça o histograma da variável `pulse` usando a função `distplot()` e o argumento `kde=False`.
5. Agora faça a distribuição de densidade da mesma variável, mas utilizando o argumento `hist=False`.
6. Use a função `swarplot()` e analise a relação entre as variáveis `kind` (x) e `pulse` (y).

EXERCÍCIO 02 - VISUALIZAÇÃO DE DADOS

1. Utilizando os datasets do Seaborn, crie o dataframe voo com o datasets flights.
2. Faça um gráfico de barras com as variáveis year e passengers.
3. Que informação podemos extrair desse gráfico?
4. Use a função boxplot() e verifique a variável passengers.
5. O que podemos depreender desse gráfico?
6. Qual a diferença entre os resultados obtidos com o gráfico de barras e o boxplot?
7. Faça um gráfico de barras com as variáveis month e passengers.
8. Qual conclusão podemos extrair desse gráfico?