Unit - 3
Data Visualization

Objective

- Data Visualization
- Plotting and Visualization
- Figures and Subplot
- Colors, markers and Line styles
- Different types of plots
- Text and Annotation
- 3D plotting
- Data Visualizing with Numpy Library

What is Data Visualization?

Data visualization is the process of translating data into a chart, graph, or other visual component.

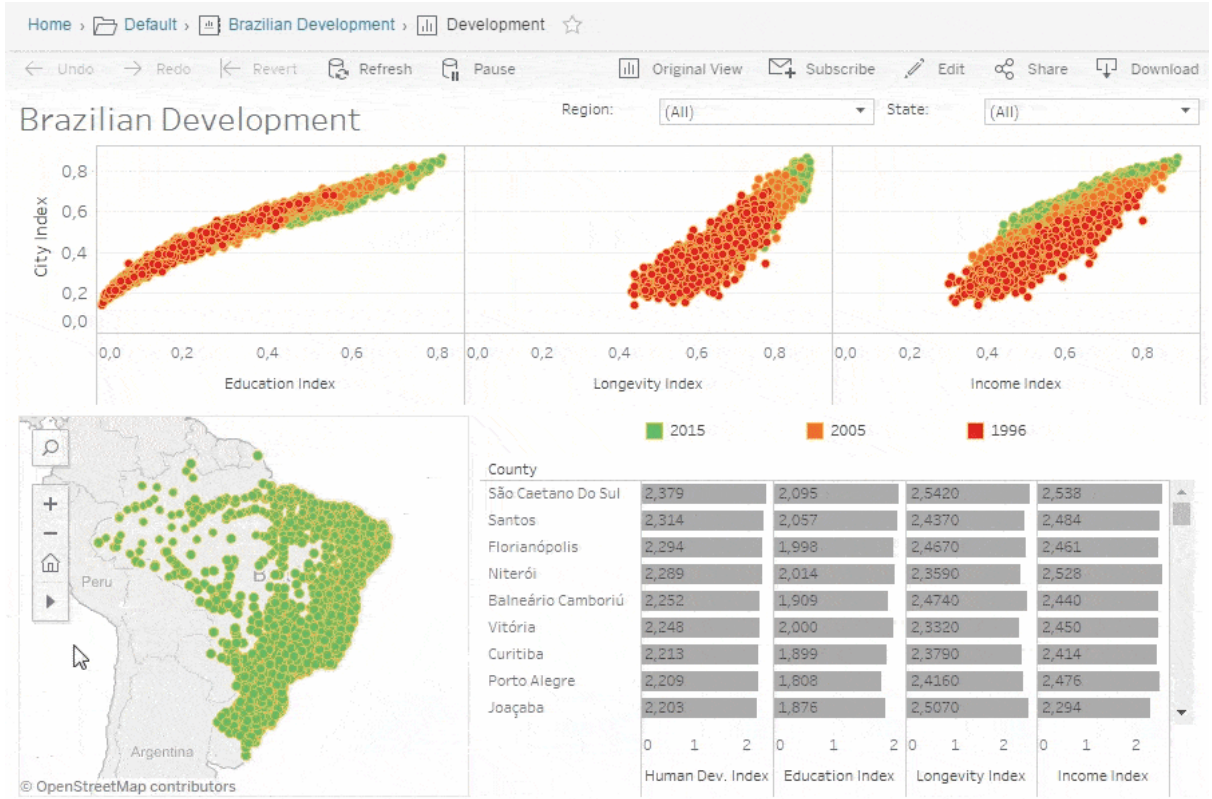


Image: sample Data visualization

Reference: https://cdnl.tblsft.com/sites/default/files/pages/_data_visualization_definition.gif

Families of Visualizations

Chart

Geovisualization

Tables

https://www.tableau.com/learn/articles/data-visualization/glossary

Plotting and Visualization

Plotting is a chart or map showing the movements or progress of an object.

Popular plotting libraries:

Matplotlib

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.

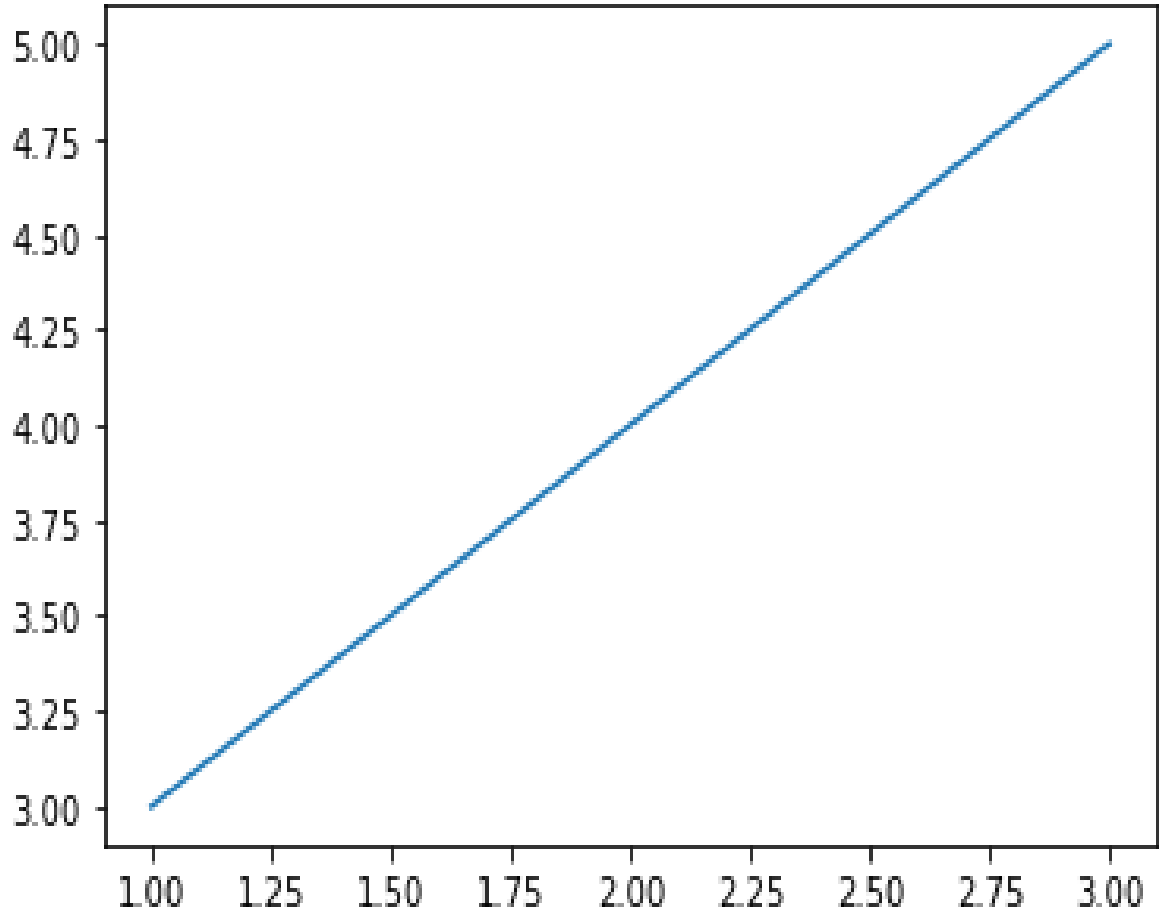
Matplotlib installation

Using pip	• *pip install matplotlib*
Using conda	• *Conda install matplotlib*

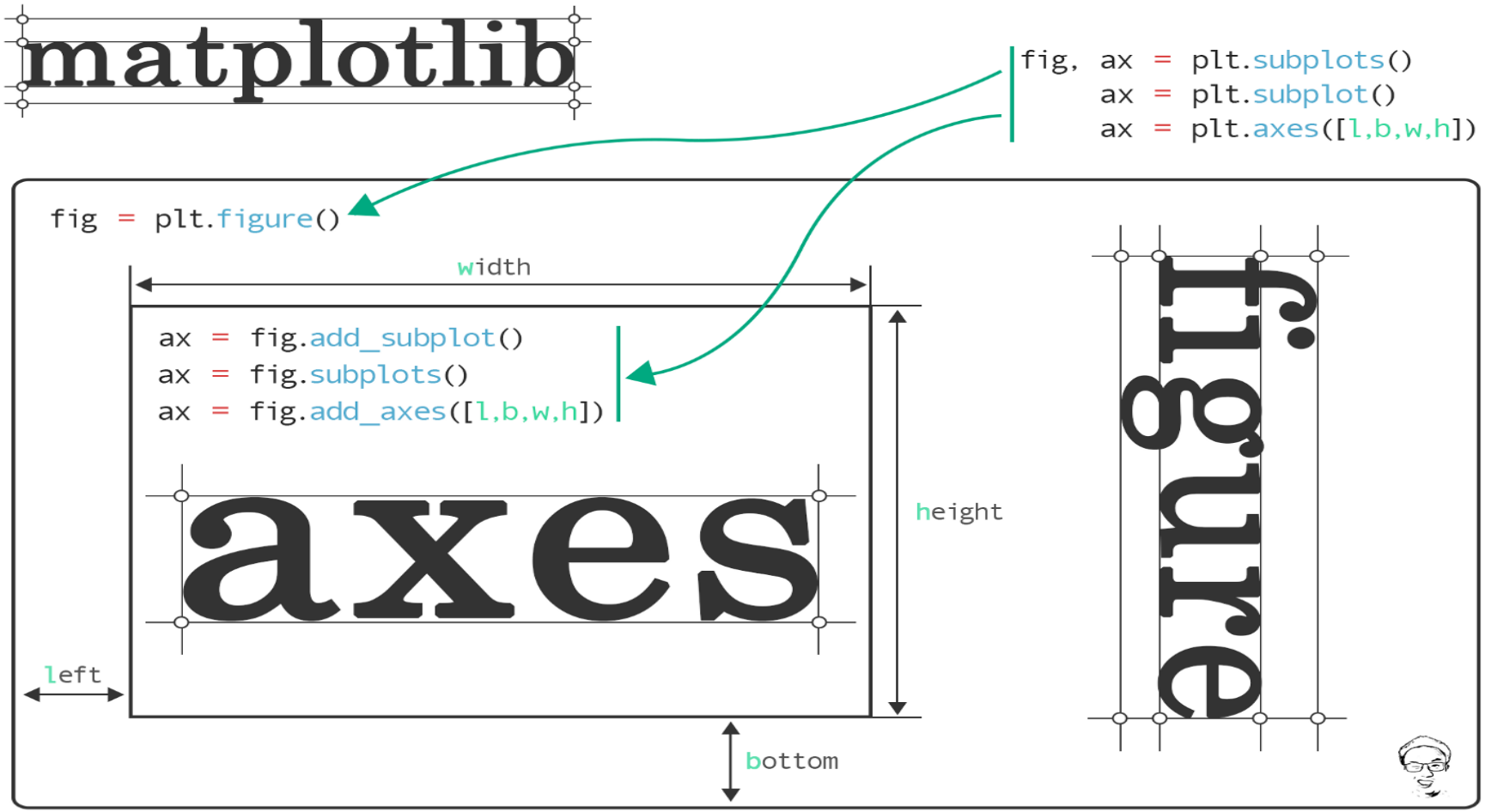
Graph plot using Matplotlib

```
Import matplotlib.pyplot as plt
// Import matplotlib
X = [1,2,3]
 //x-axis coordinates
Y= [3,4,5]
 //y-axis coordinates

plt.plot(x,y)
 // call plot function to draw plot
plt.show()
 //call show function to  show plot on output
```

Output



Figures and subplot in Matplotlib

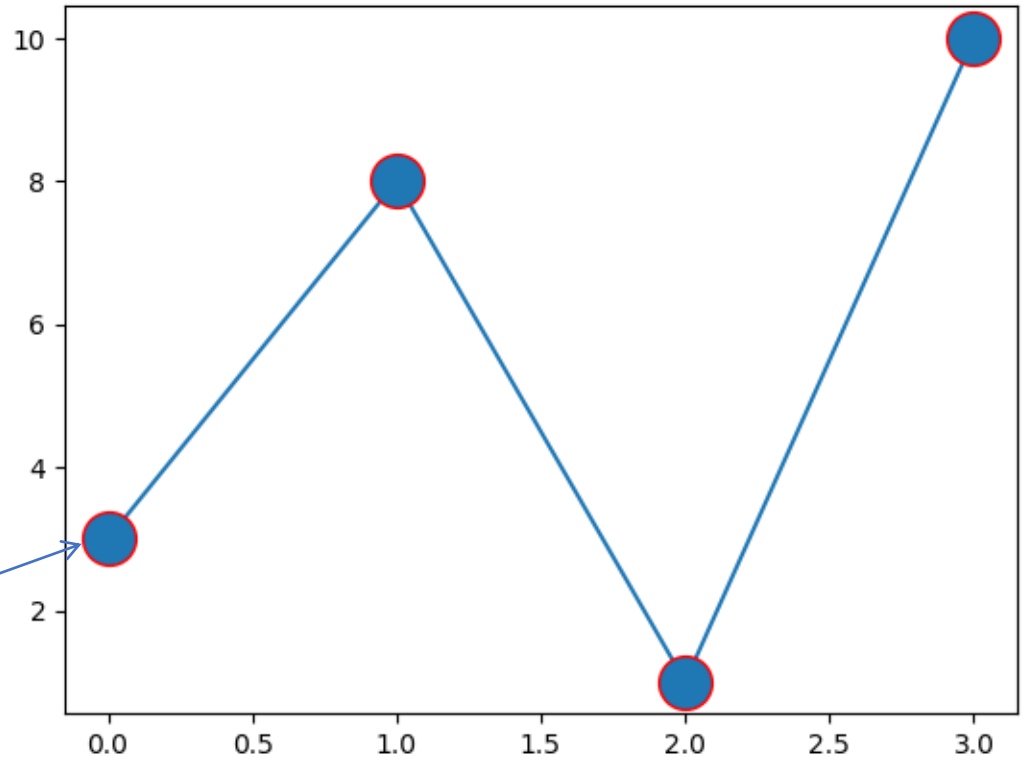


https://towardsdatascience.com/the-many-ways-to-call-axes-in-matplotlib-2667a7b06e06

Colors, Markers and line styles

Marker keyword used to emphasize each point with a specifies marker

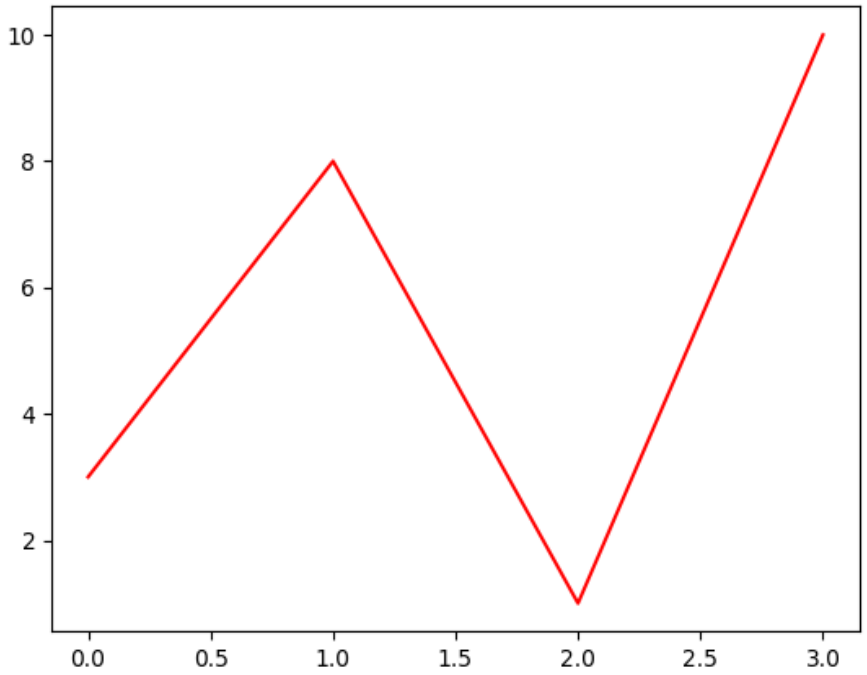
```
import matplotlib.pyplot as plt
import numpy as np

ypoints = np.array([3, 8, 1, 10])

plt.plot(ypoints, marker = 'o')
plt.show()
```



Colors, Markers and line styles

- You can use the keyword argument **color** or the shorter **c** to set the color of the line.

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

ypoints = np.array([3, 8, 1, 10])

plt.plot(ypoints, color = 'r')
plt.show()
```

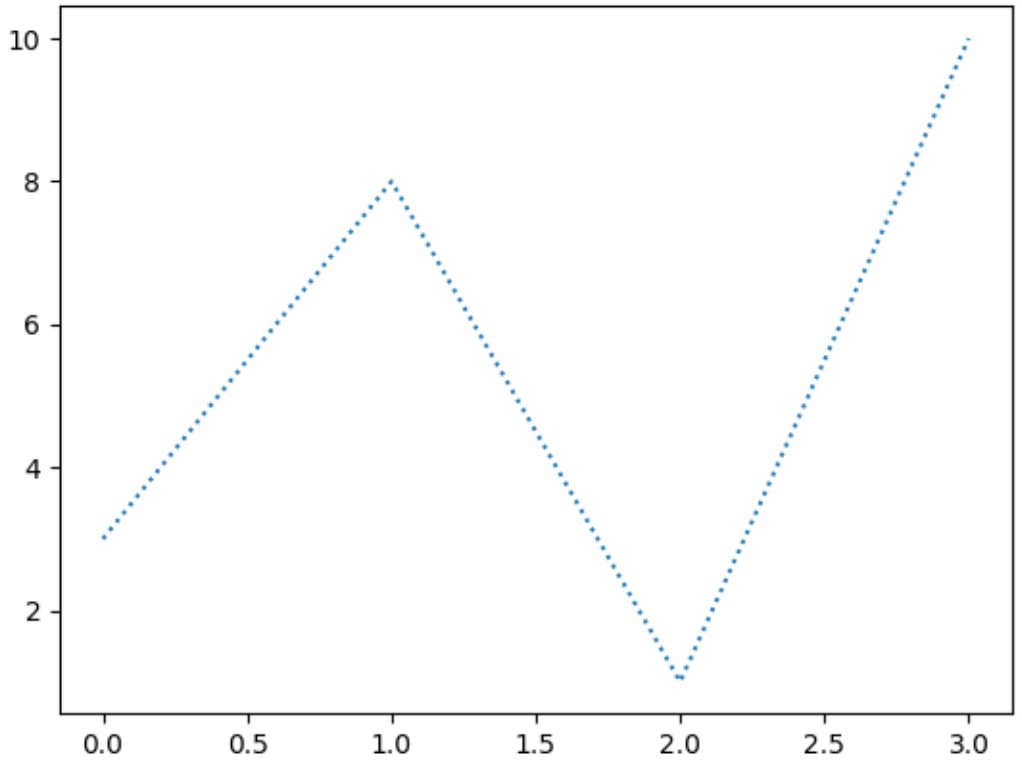


Colors, Markers and line styles

- **Linestyle** is keyword used to change the style of the plotted line:
- import matplotlib.pyplot as plt
  import numpy as np

  ypoints = np.array([3, 8, 1, 10])

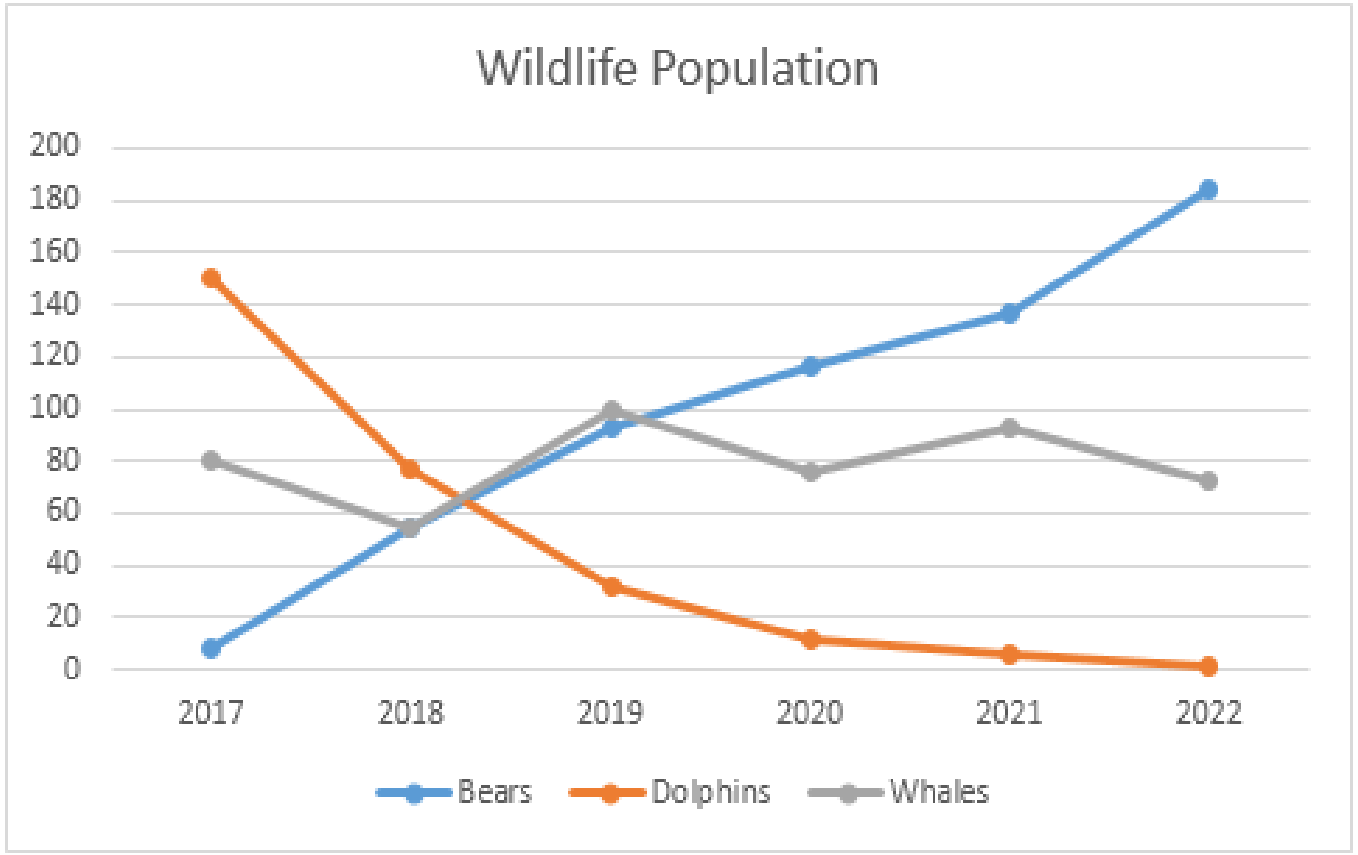
  plt.plot(ypoints, linestyle = 'dotted')
  plt.show()



Line Plot

According to the wiki,

“A line chart or line plot or line graph is a type of chart which displays information as a series of data points called ‘markers’ connected by straight line segments. It is a basic type of chart common in many fields”



https://medium.com/@patrickbfuller/line-plot-7b4068a3a9fc

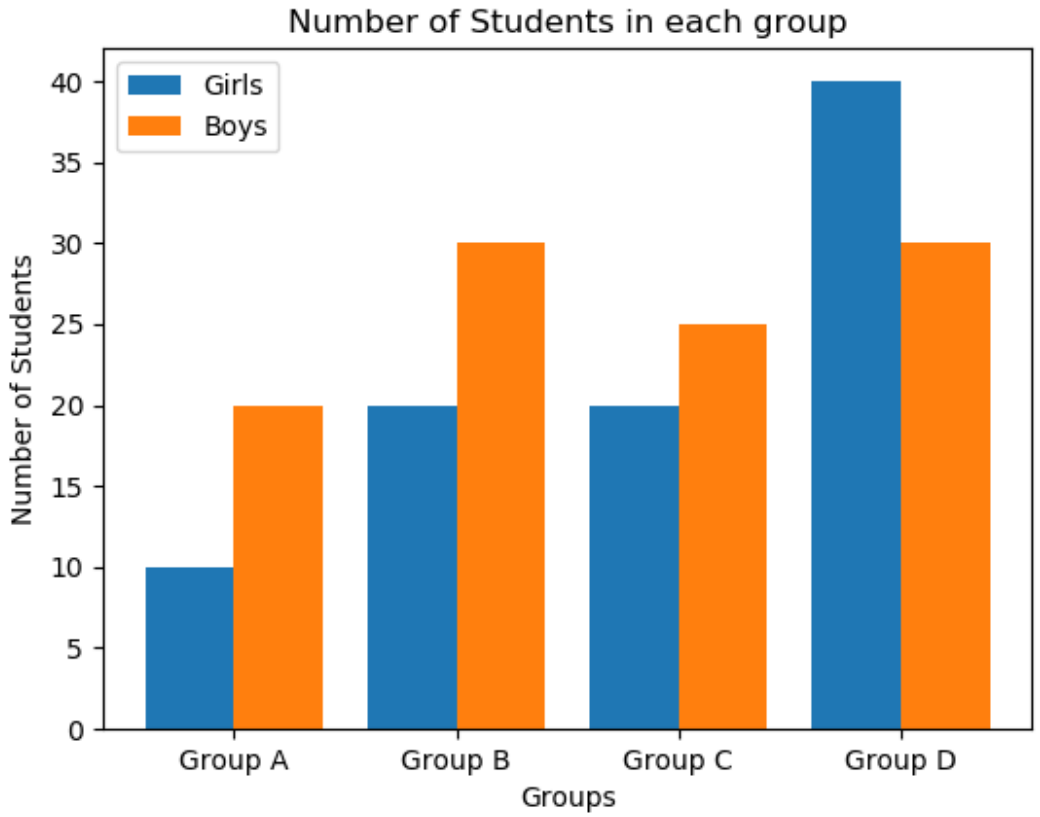
Bar Plot

A bar chart or bar graph is a chart or graph that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally.

Syntax:

ax.bar(x, height, width, bottom, align)

plt.bar(x,y)



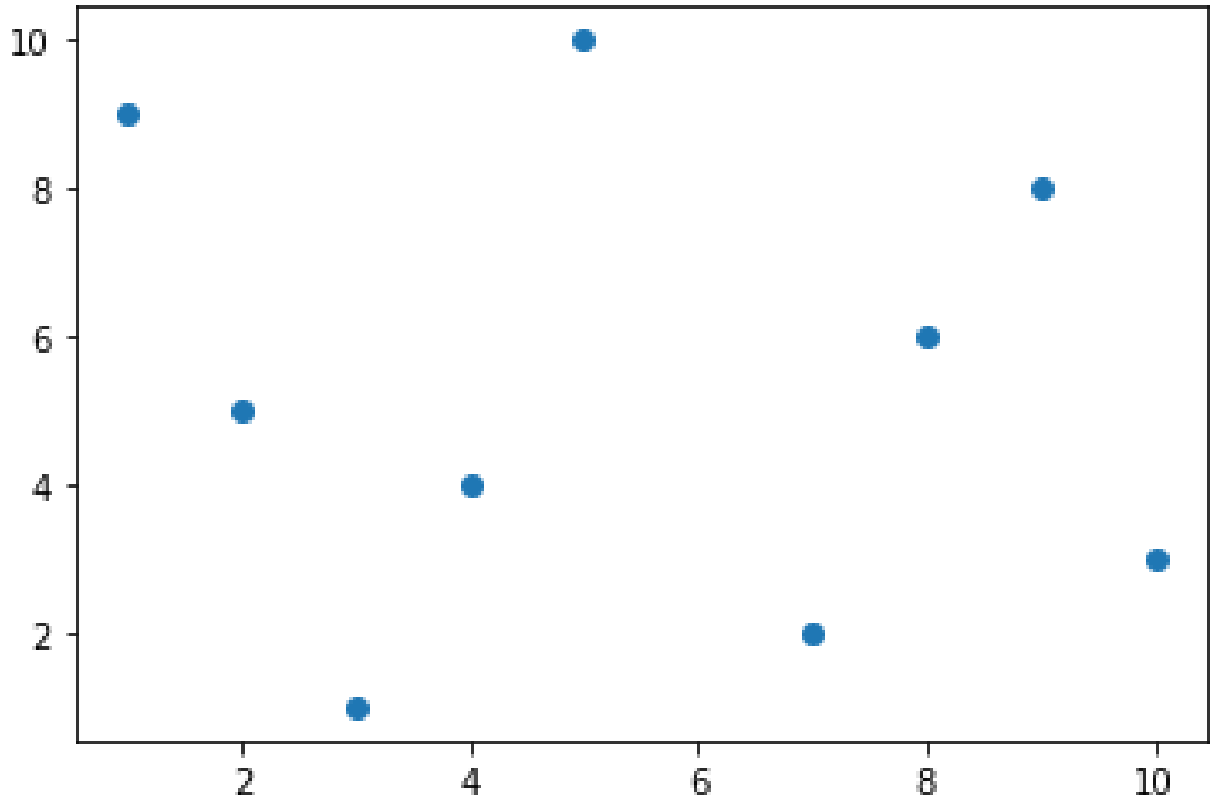
Scatter Plot

A scatter plot is a diagram where each value in the data set is represented by a dot.

Use the **scatter()** method to draw a scatter plot diagram:

Syntax:

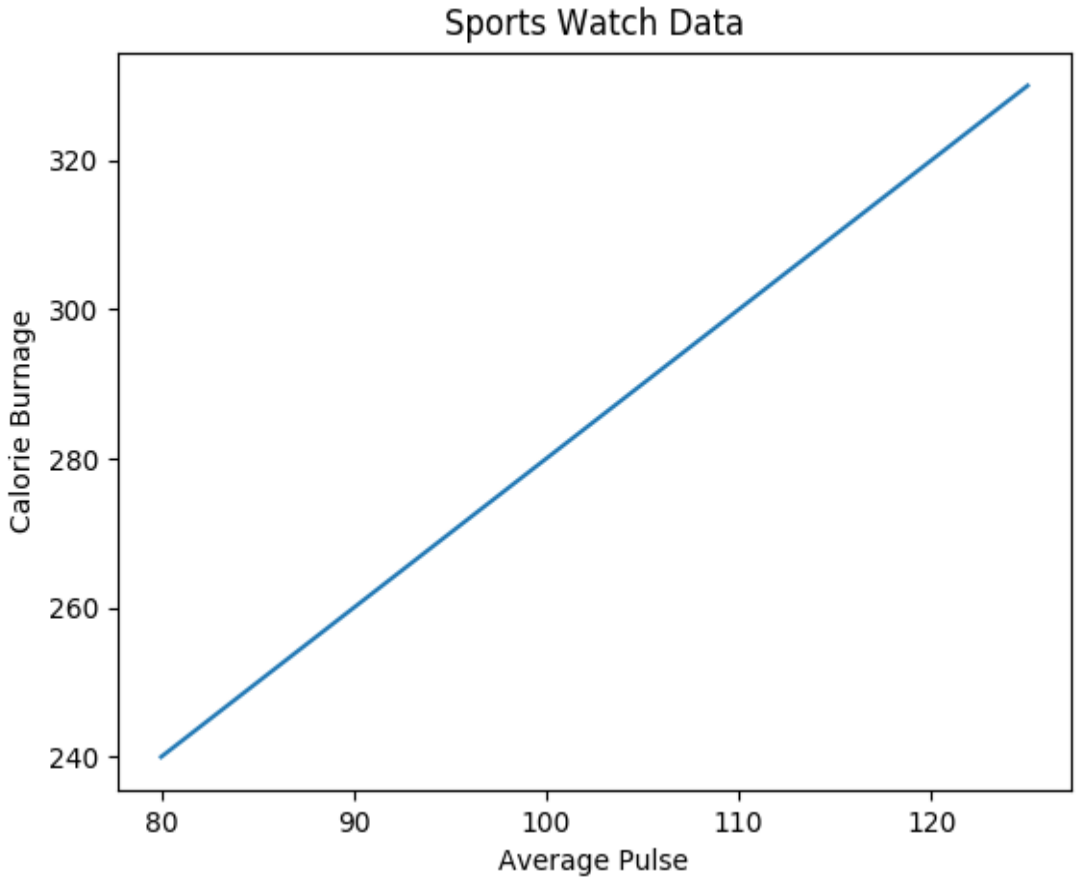
plt.scatter(x, y)



Matplotlib Labels and Title

- With Pyplot, you can use the xlabel() and ylabel() functions to set a label for the x- and y-axis.
- With Pyplot, you can use the title() function to set a title for the plot.
- Syntax:

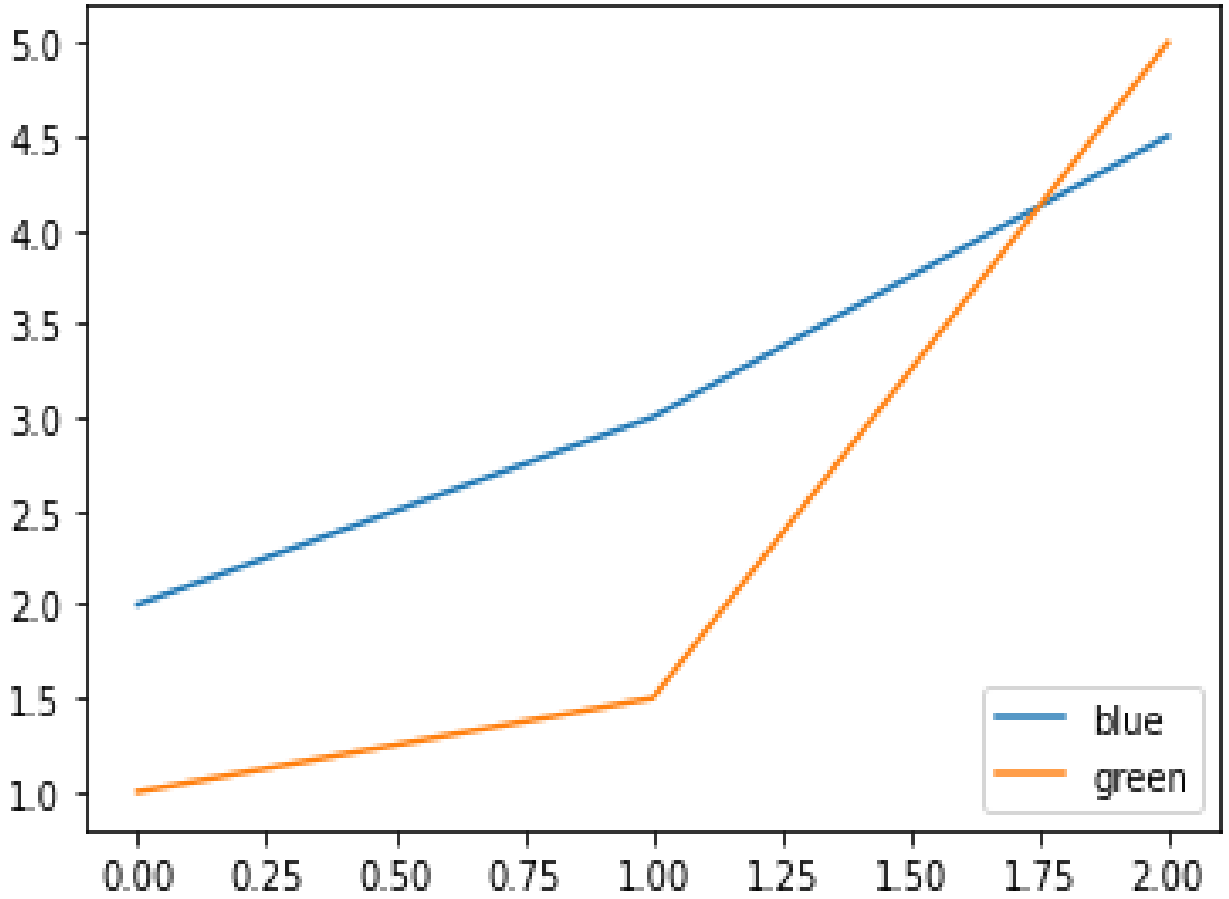
plt.title("Sports Watch Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")



Adding Legend in the graph

- A legend is an area describing the elements of the graph. In the matplotlib library, there's a function called legend() which is used to place a legend on the axes.

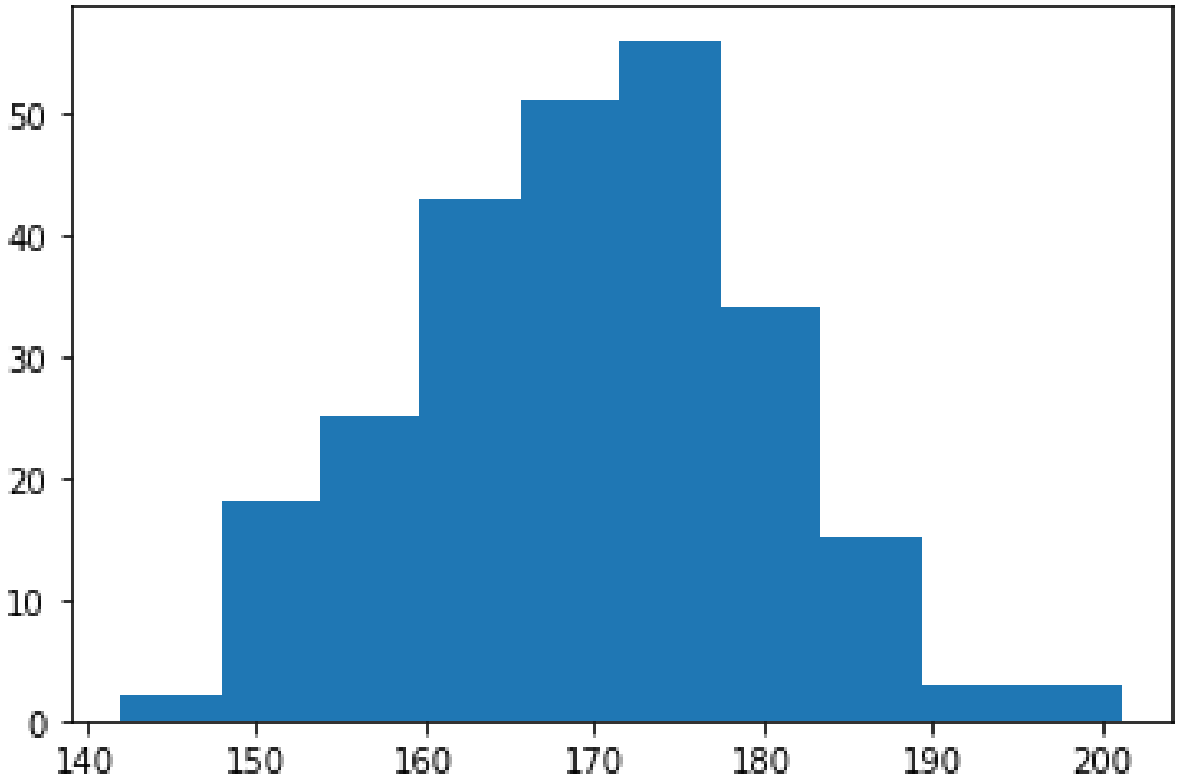
- Syntax:
  Matplotlib.pyplot.legend()



Histograms and Binning

- A histogram is an accurate graphical representation of the distribution of numerical
- In Matplotlib, we use the **hist()** function to create histograms.

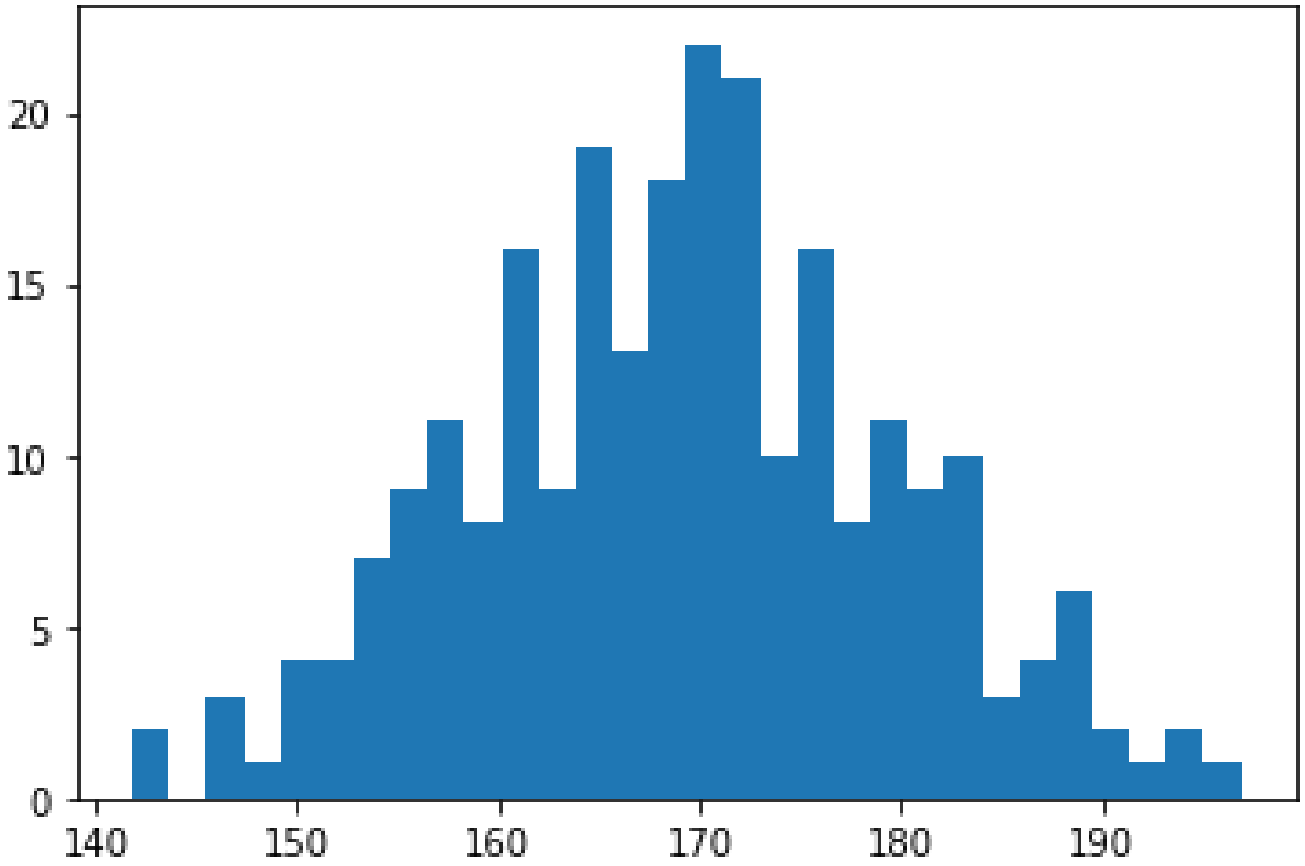
- Syntax:
  Matplotlib.pyplot.hist()



Histograms and Binning

- Set the bins value to the histogram

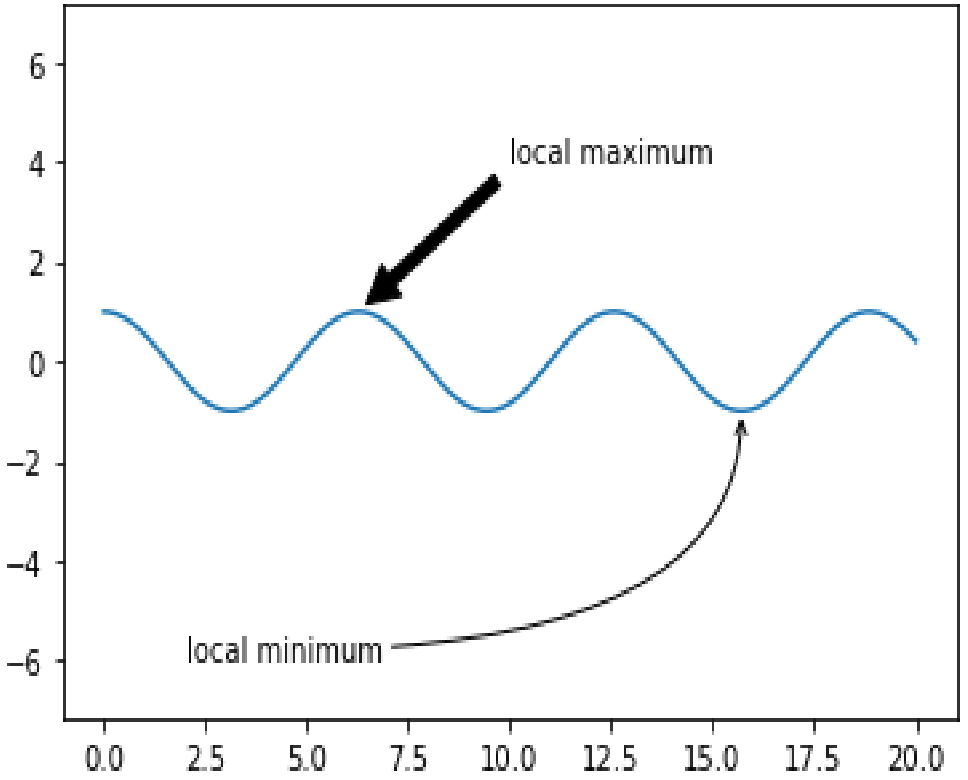
- Syntax:
  Matplotlib.pyplot.hist(data, bins = 30);



Text and Annotations

- Create an annotation: a piece of text referring to a data point.
- The **annotate()** function in **pyplot** module of matplotlib library is used to annotate the point xy with text

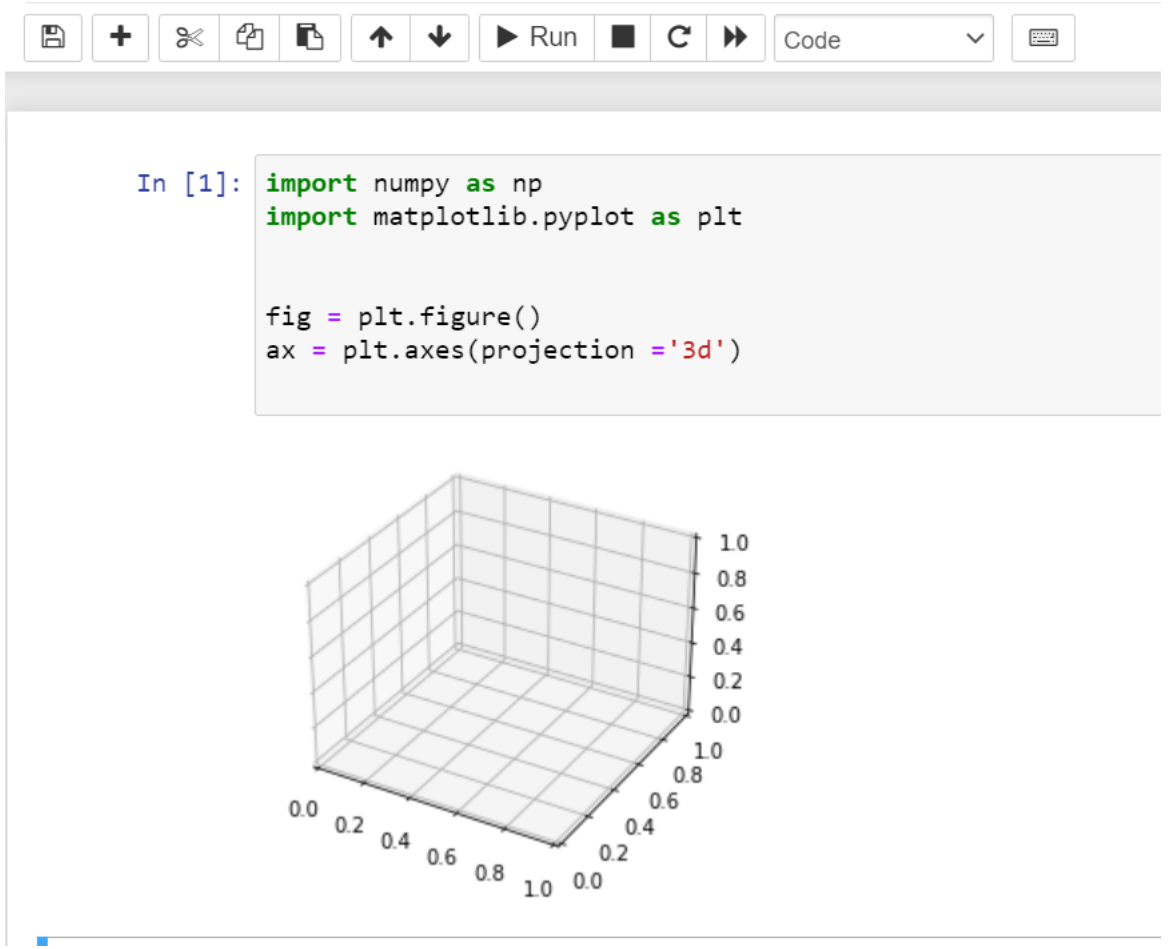
- Syntax:
  Matplotlib.pyplot.annotate("text");



3-D Plotting in Matplotlib

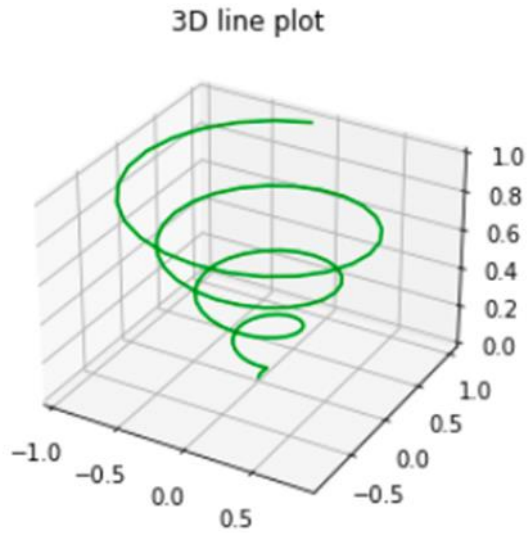
- Three-dimensional plots are enabled by importing the **mplot3d** toolkit, included with the main Matplotlib installation:
- **from mpl_toolkits import** mplot3d

- Syntax:
  plt.plot3D()



Plotting 3-D Lines and Points

Graph with lines and point are the simplest 3 dimensional graph. ax.plot3d and ax.scatter are the function to plot line and point graph respectively.



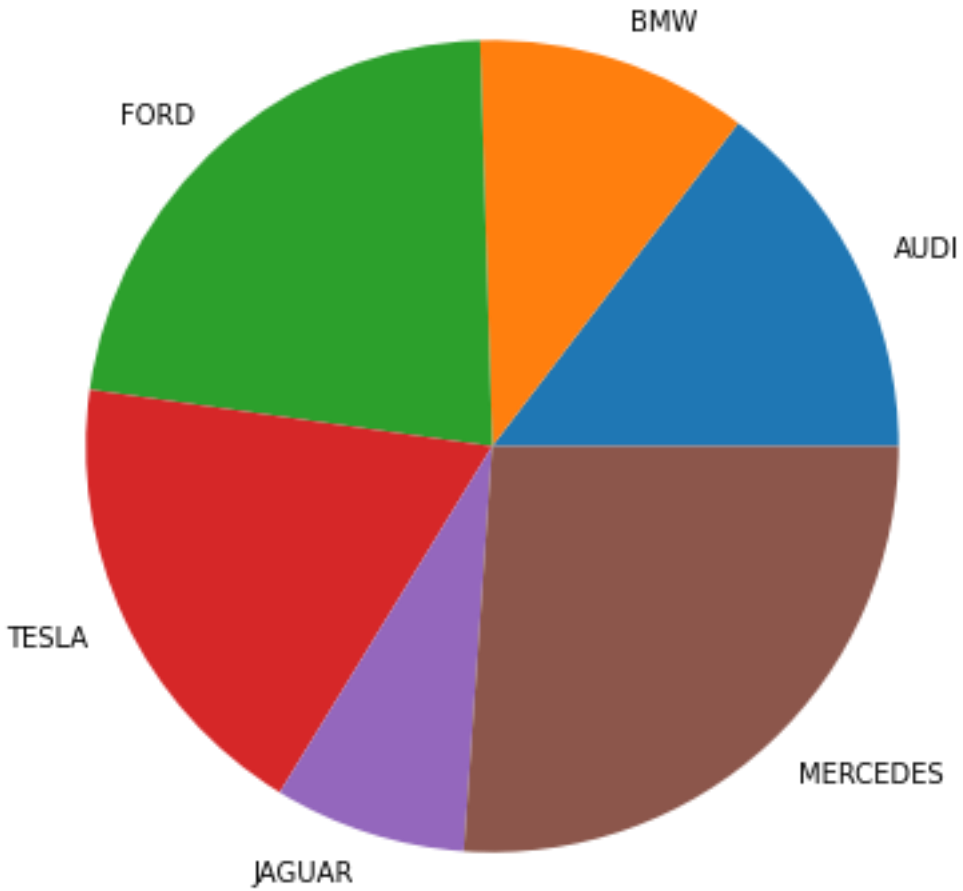
Run Code

```
In [1]: # importing mplot3d toolkits, numpy and matplotlib
from mpl_toolkits import mplot3d
import numpy as np
import matplotlib.pyplot as plt

fig = plt.figure()

# syntax for 3-D projection
ax = plt.axes(projection ='3d')

# defining all 3 axes
z = np.linspace(0, 1, 100)
x = z * np.sin(25 * z)
y = z * np.cos(25 * z)

# plotting
ax.plot3D(x, y, z, 'green')
ax.set_title('3D line plot')
plt.show()
```

Pie Chart

- Matplotlib API has **pie()** function in its pyplot module which create a pie chart representing the data in an array.

- Syntax:
  plt.pie(data)



https://www.geeksforgeeks.org/plot-a-pie-chart-in-python-using-matplotlib/

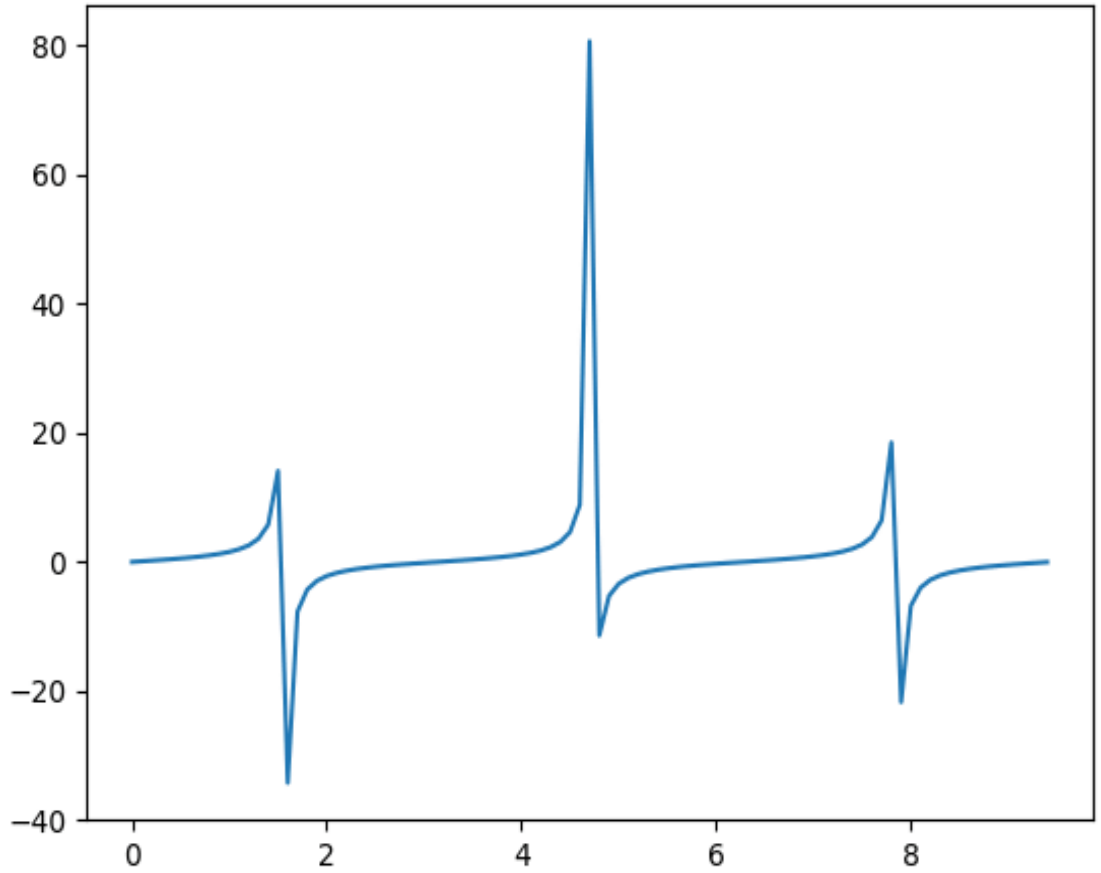
Visualizing with NumPy

- **NumPy** is a Python library used for working with arrays.

- Syntax:
  import numpy as np

Code :

```
import numpy as np
import matplotlib.pyplot as plt
x= np.arange(0,3*np.pi,0.1)
y=np.tan(x)
plt.plot(x,y)
plt.show()
```



References

1. https://www.tableau.com/learn/articles/data-visualization/glossary
2. https://www.w3schools.com/python/matplotlib_markers.asp
3. https://www.geeksforgeeks.org/matplotlib-pyplot-legend-in-python/
4. https://jakevdp.github.io/PythonDataScienceHandbook/04.09-text-and-annotation.html
5. https://www.geeksforgeeks.org/three-dimensional-plotting-in-python-using-matplotlib/
6. https://www.tutorialspoint.com/numpy/numpy_matplotlib.htm:

Microsoft

SAP

edunet foundation

THANK YOU