Data Visualization and MatPlotLib

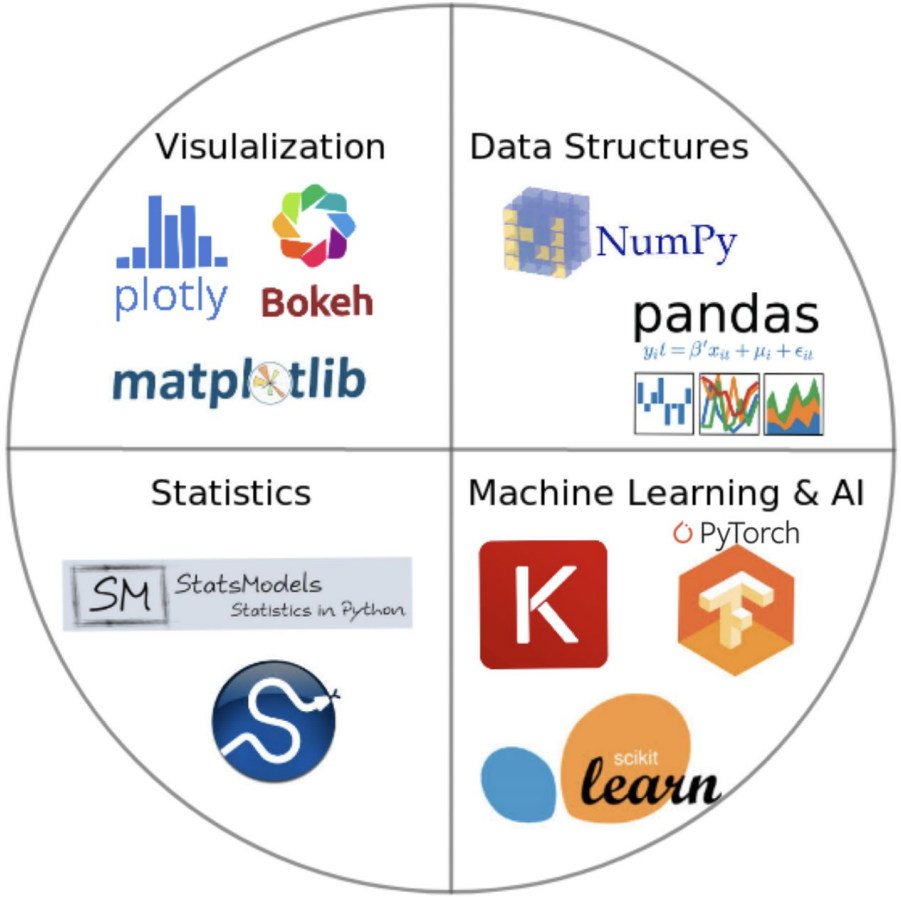
Introduction to Python & Data Science

Instructor: Dr. Sneha Singh, SCEE

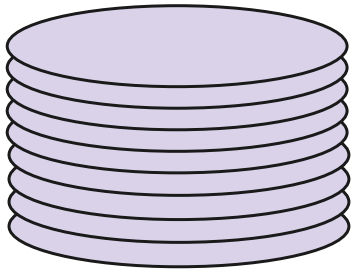
Term : August 2025

Python: Ecosystem

Image by duchesnay.github.io



Data: Understanding



Data

Ultimate information of interest.

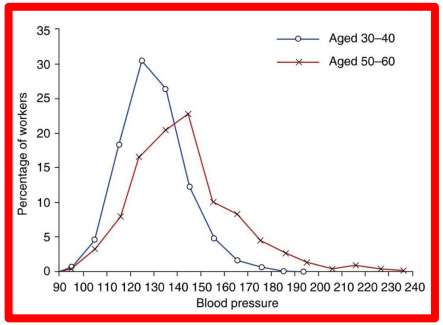
Metadata

Data about the data

Metadata is characterized using the different aspects:

- Format and acquisition process: *Structure*
- Entities the data record contains: *Granularity*
- *Quality* of data
- *Temporality*
- No. of features and population they cover: *Scope*

Data: Description



- Numerical findings: need to be presented concisely.
- Especially needed for large datasets.
- Features of the data include:
  - Range
  - Degree of symmetry
  - Concentrated or spread out
  - Where are they concentrated? Etc.
- Univariate or multivariate
  - One variable or multiple!

Data: Description

- Frequency Table for small datasets
- *No. of leaves taken by 15 employees*, over six weeks:

  0, 2, 1, 2, 0, 5, 5, 0, 1, 3, 2, 0, 1, 2, 0

Value	Frequency
0	5
1	3

Data: Description

- Frequency Table for small datasets
- *No. of leaves taken by 15 employees*, over six weeks:

  0, 2, 1, 2, 0, 5, 5, 0, 1, 3, 2, 0, 1, 2, 0

Value	Frequency
0	5
1	3
2	4
3	1
4	0
5	2

Data: Description

- Frequency Table for small datasets
- *No. of leaves taken by 15 employees*, over six weeks:

  0, 2, 1, 2, 0, 5, 5, 0, 1, 3, 2, 0, 1, 2, 0

Value	Frequency
0	5
1	3
2	4
3	1
4	0
5	2
Total	**15**

Data: Description

- Frequency Table for small datasets
- *No. of leaves taken by 15 employees*, over six weeks:

  0, 2, 1, 2, 0, 5, 5, 0, 1, 3, 2, 0, 1, 2, 0

Value	Frequency
0	5
1	3
2	4
3	1
4	0
5	2
Total	**15**

➔ How many workers had at least one day of leave?

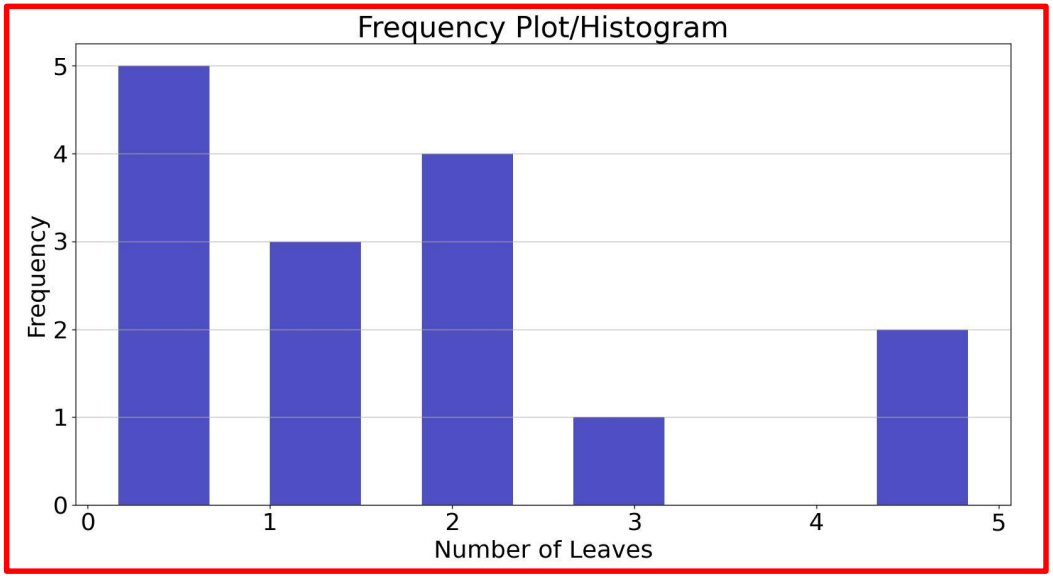
➔ How many workers had leaves between 3 and 5 days?

➔ How many had more than 5 days of leave?

Data: Description

- How many workers had at least one day of leave?
- How many workers had leaves between 3 and 5 days?
- How many had more than 5 days of leave?

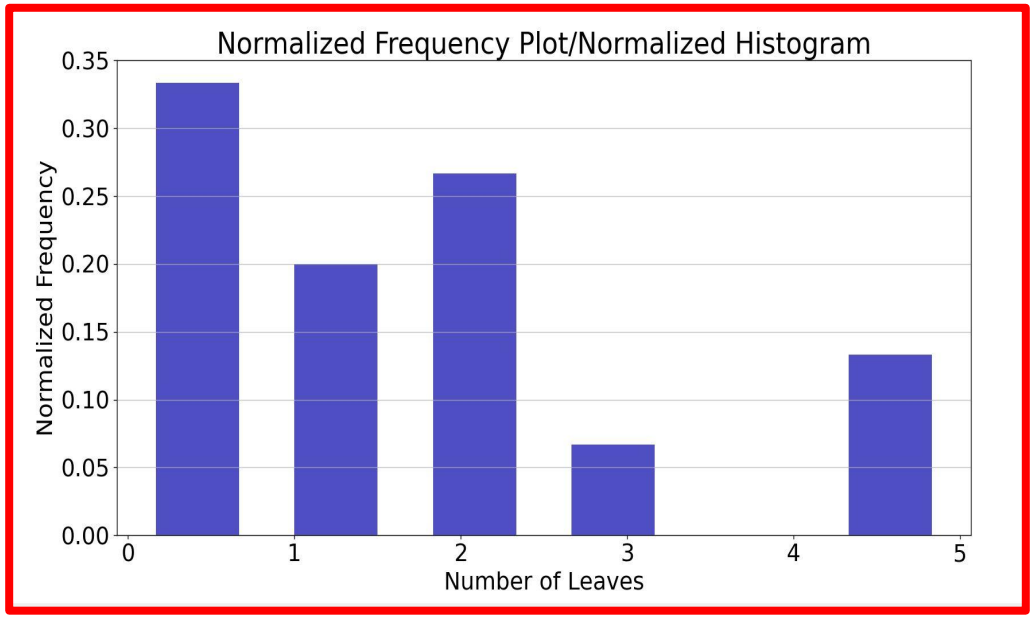
Value	Frequency
0	5
1	3
2	4
3	1
4	0
5	2
Total	**15**



0 =5 1=3 2=4 3=1 4=0 5=2

Data: Description

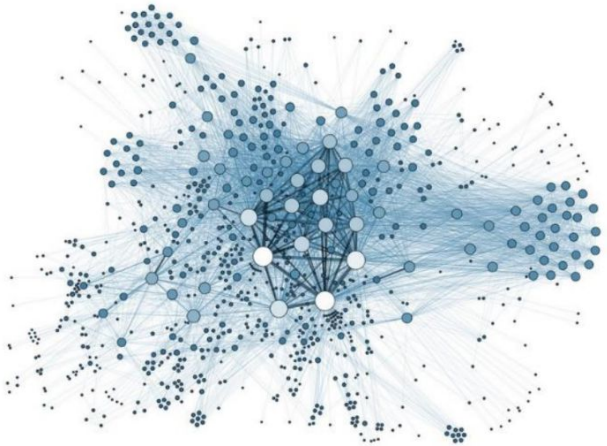
Value	Normalized Frequency
0	5/15
1	3/15
2	4/15
3	1/15
4	0/15
5	2/15
Total	**1**

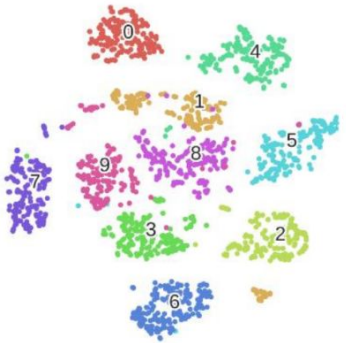


Sometimes normalized frequency is more convenient.
Sum of frequency values = 1.

Data: Statistics

Statistics are functions of a sample of random variables and so are random variables themselves, with distribution functions.





statistic (estimator, estimate)	parameter
mean $\bar{X}$, $\bar{x}$	$\mu = E[X]$
variance S^2, s^2	$\sigma^2 = Var[X] = E[X^2] - \mu^2$
standard deviation S, s	$\sigma = SD[X]$
proportion $\hat{P}$, $\hat{p}$	p
histogram/barplot	pdf/pmf

Statistics: Random Space & Variables

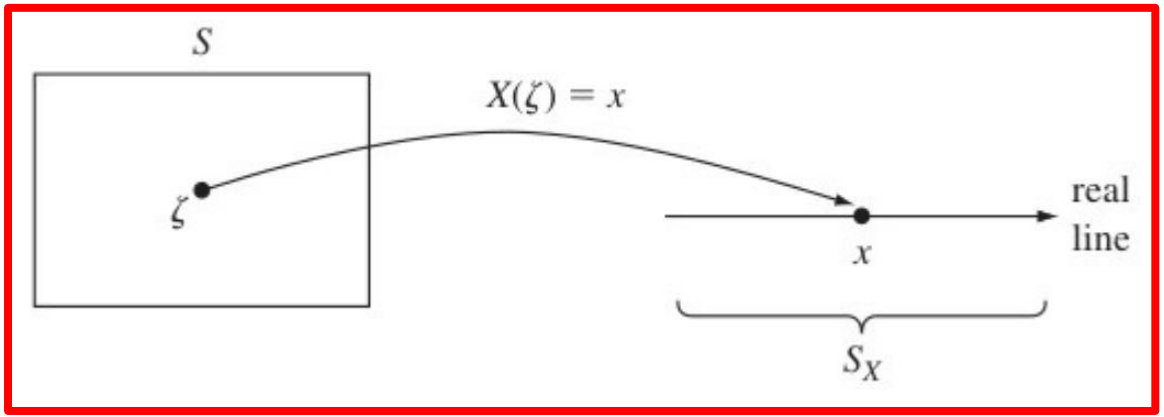
Statistics are functions of a sample of random variables and so are random variables themselves, with distribution functions.

- Sample Space ***S***: All possible outcomes of a random experiment.
- Define the event class $\mathcal{F}$ of all events of interest.
  Note: Event is set of outcomes.
- Events in $\mathcal{F}$ are assigned probabilities.

In many random experiments, we may not be interested in the observed values, but in some *numerical quantity determined by the observed values*.

Statistics: Random Space & Variables

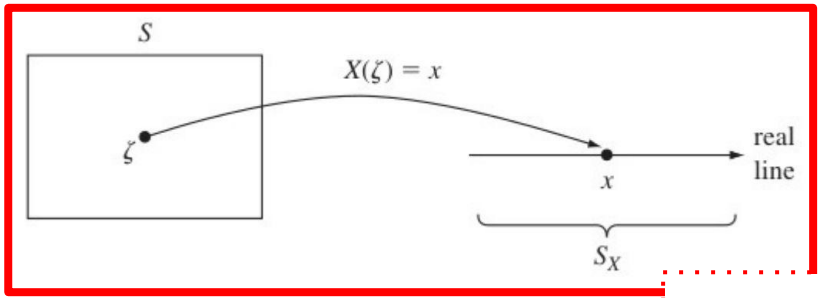
Statistics are functions of a sample of random variables and so are random variables themselves, with distribution functions.



- Random variables are any such quantities determined by the results of random experiments.
- A random variable is a ***function*** that assigns a real number to each outcome in the sample space.

Statistics: Expected Values

Statistics are functions of a sample of random variables and so are random variables themselves, with distribution functions.



- The expected value of X is

$$E[X] = \mu = \sum_{x \in S_X} x P(X=x)$$

- The variance is

$$\sigma^2 = E[(x-\mu)^2] = \sum_{x \in S_X} (x-\mu)^2 P(X=x)$$

Contrast this with the sample mean and sample variance

Statistics: Expected Values

Statistics are functions of a sample of random variables and so are random variables themselves, with distribution functions.

Value of X (denoted as x) X = Sum of 2 dice throws.	**Probability Mass Function (PMF)** of X **P(X = x)**
2	1/36
3	2/36
4	3/36
5	
. . .	
12	1/36

- The expected value of X is

$$E[X] = \mu = \sum_{x \in S_X} x P(X=x)$$

- The variance is

$$\sigma^2 = E[(x-\mu)^2] = \sum_{x \in S_X} (x-\mu)^2 P(X=x)$$

Contrast this with the sample mean and sample variance

Statistics: Expected Values

Question?

Find Expected value of following PMF:

$P(X=x) = k / x^2$ for $x \geqslant 1$, $x \in Z^+$

What will be value of constant k?

- The expected value of X is

$$E[X] = \mu = \sum_{x \in S_X} x P(X=x)$$

- The variance is

$$\sigma^2 = E[(x-\mu)^2] = \sum_{x \in S_X} (x-\mu)^2 P(X=x)$$

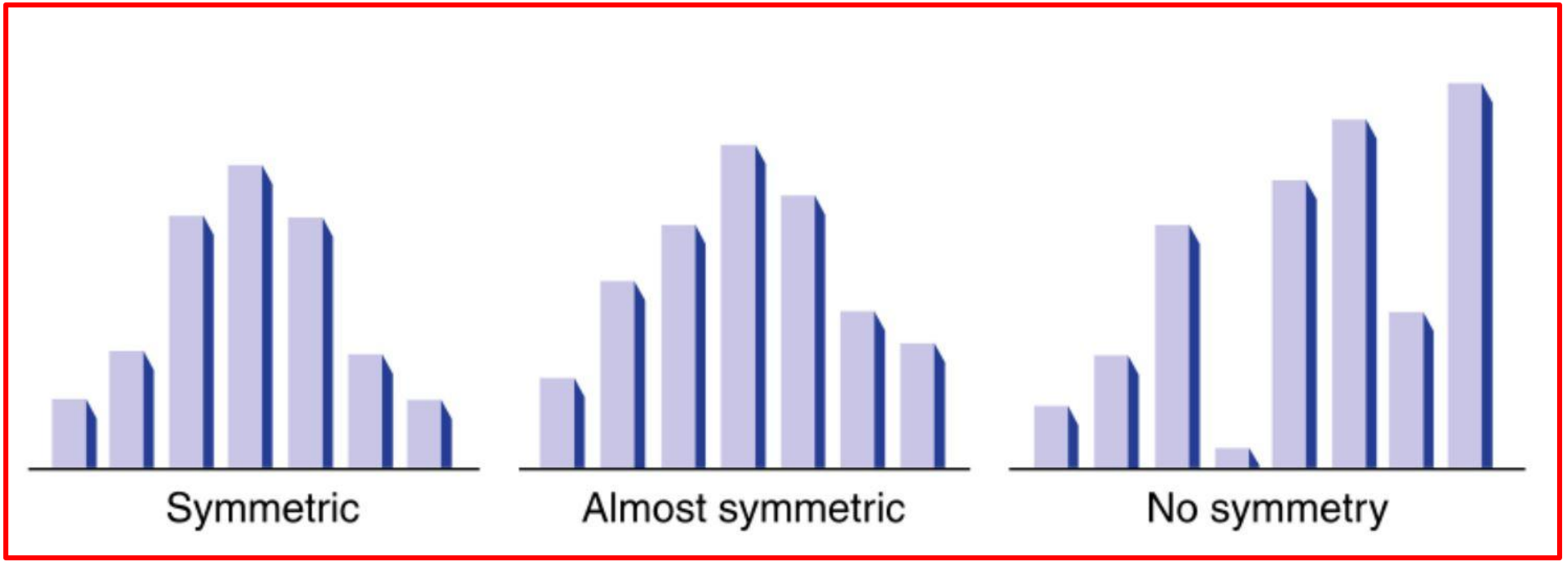
Contrast this with the sample mean and sample variance

Data Visualization: Pie Chart

- Non-numerical data Visualization of relative frequency plot
- A whole split into parts
- Emphasizes that all parts sum to a constant
- Single dimension with discrete categories

Weapons used in crimes.

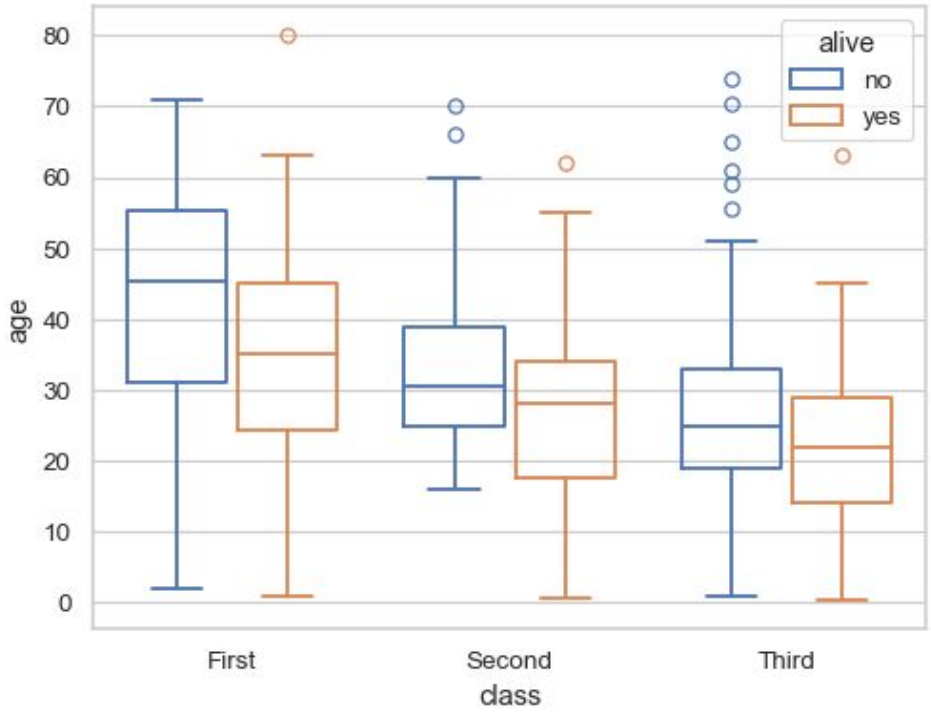
Data Visualization: Bar Graph



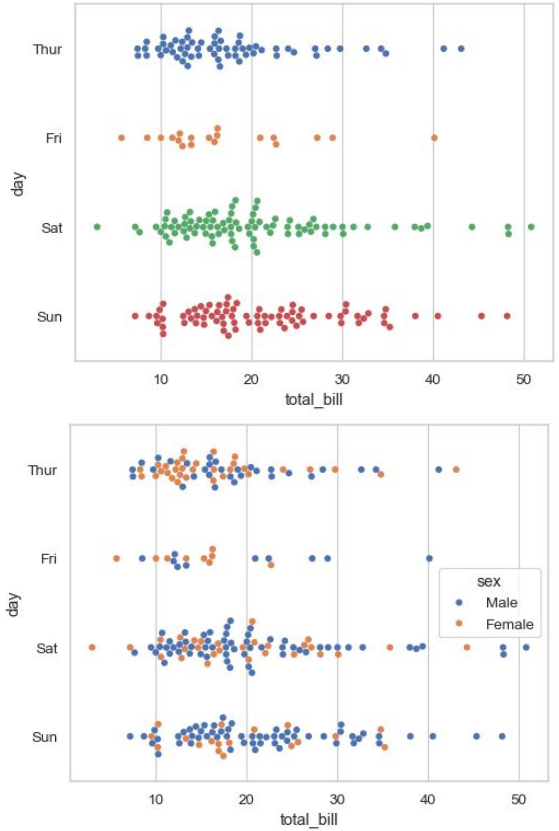
Can be used for lots of discrete grouping factors

Data Visualization: Bar Graph

Can be used for lots of discrete/Categorical grouping factors



Box Plot:Titanic dataset

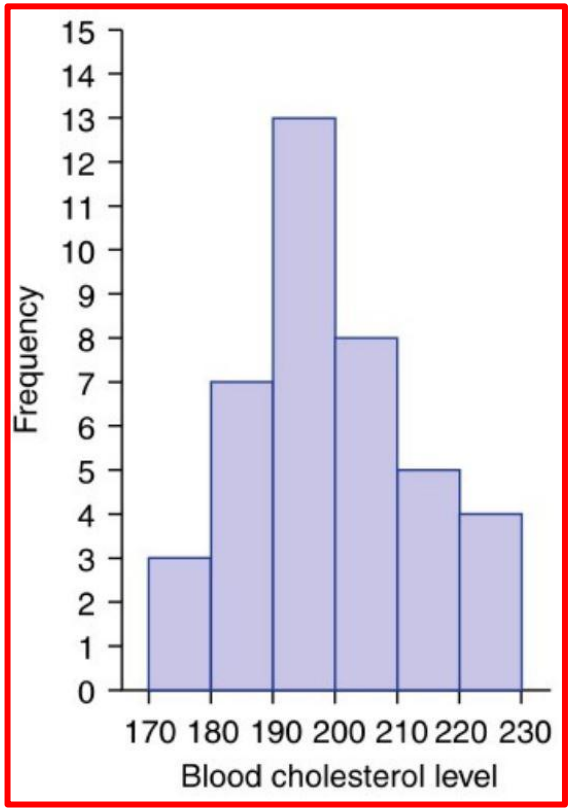


Swarm Plot: Tip as the function of day

Data Visualization: Histogram

213	174	193	196	220	183	194	200
192	200	200	199	178	183	188	193
187	181	193	205	196	211	202	213
216	206	195	191	171	194	184	191
221	212	221	204	204	191	183	227

Data: Blood cholesterol levels



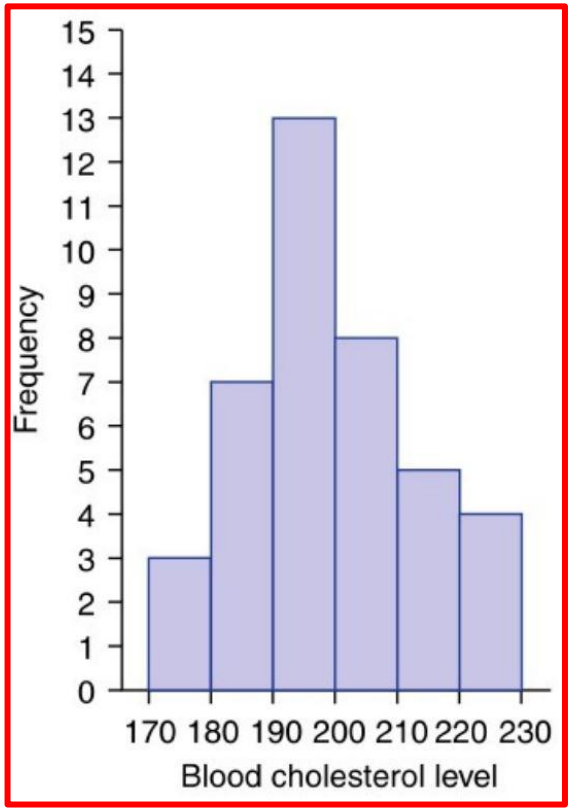
Data Visualization: Histogram

Data: Sorted Blood cholesterol levels

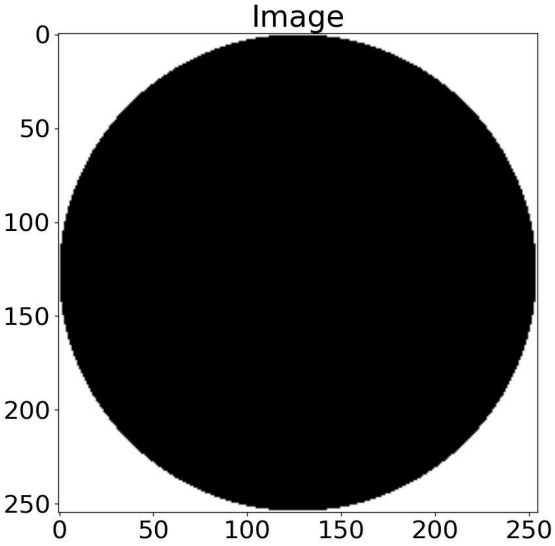
171, 174, 178, 181, 183, 183, 183, 184, 187, 188, 191,
191, 191, 191, 192, 193, 193, 193, 194, 194, 195
196, 196, 199, 200, 200, 200, 202, 204, 204, 205, 206,
206, 211, 212, 213, 213, 216, 220, 221, 221, 227

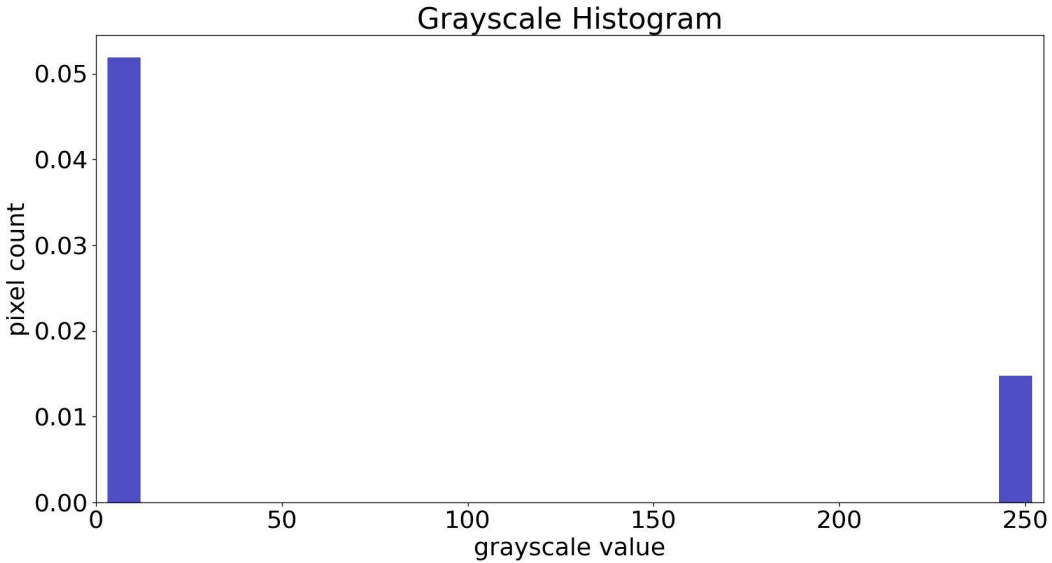
Class interval contains left-end, but not right-end

Class intervals	Frequency	Relative frequency
170–180	3	$\frac{3}{40} = 0.075$
180–190	7	$\frac{7}{40} = 0.175$
190–200	13	$\frac{13}{40} = 0.325$
200–210	8	$\frac{8}{40} = 0.20$
210–220	5	$\frac{5}{40} = 0.125$
220–230	4	$\frac{4}{40} = 0.10$

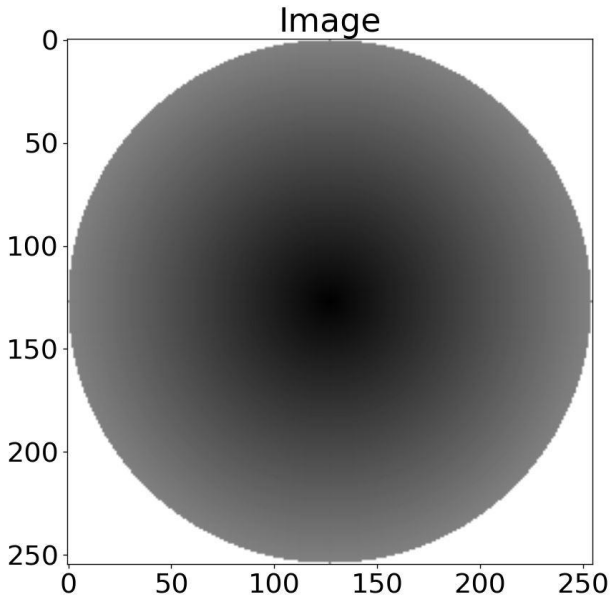


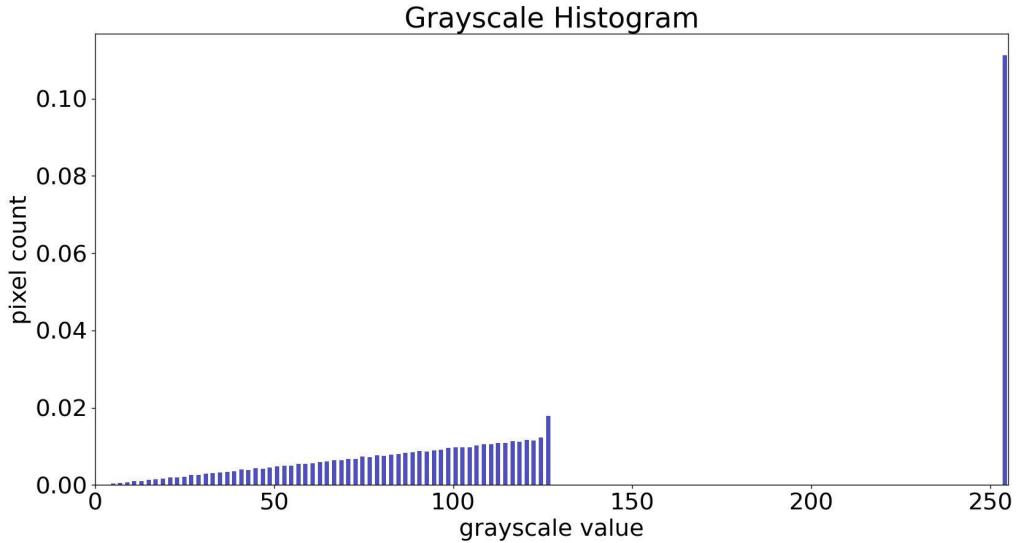
Data Visualization: Image Histogram



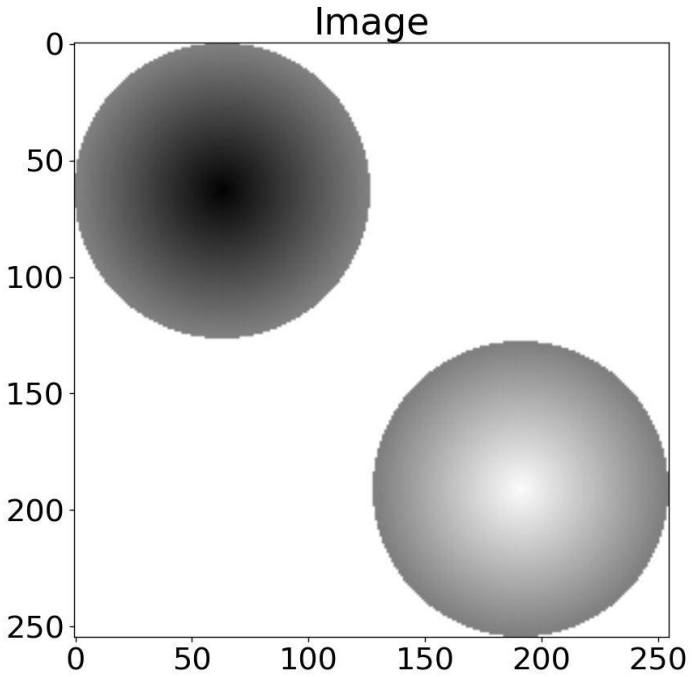


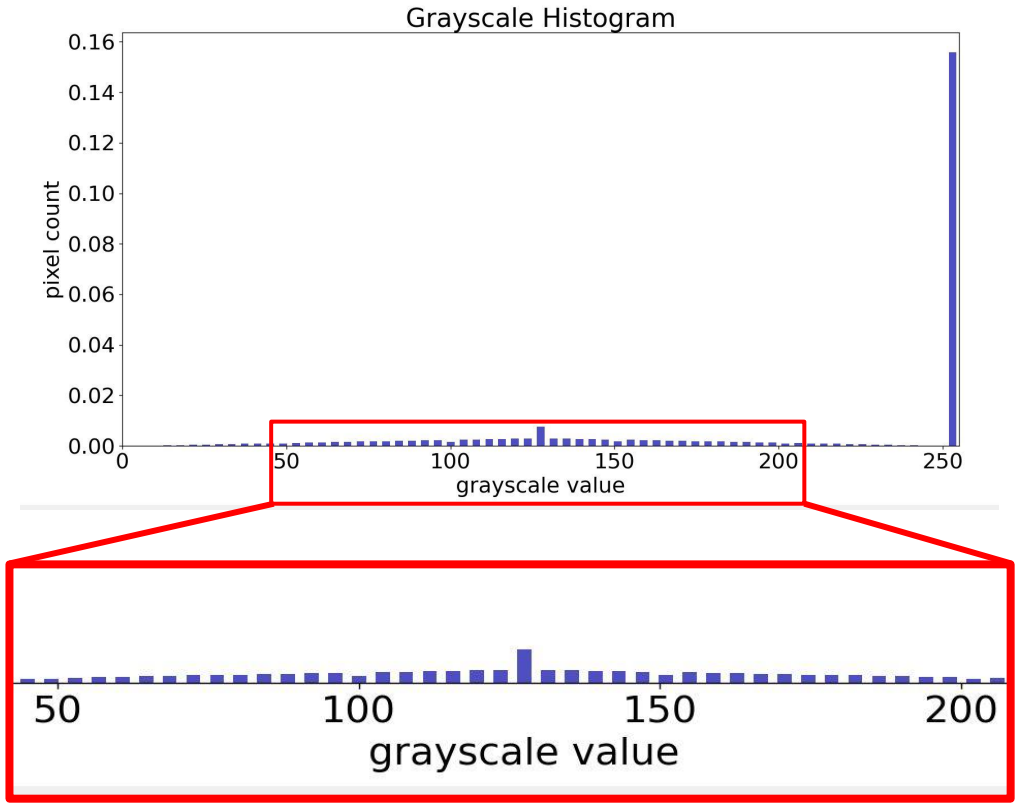
Data Visualization: Image Histogram





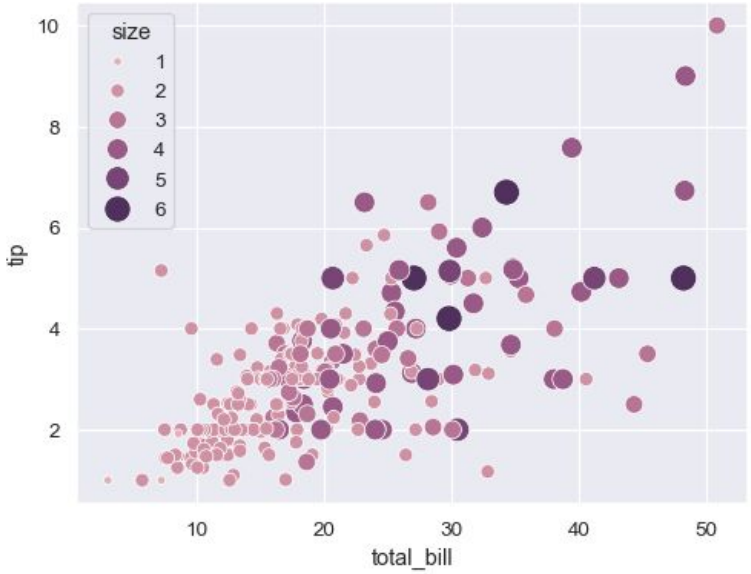
Data Visualization: Image Histogram

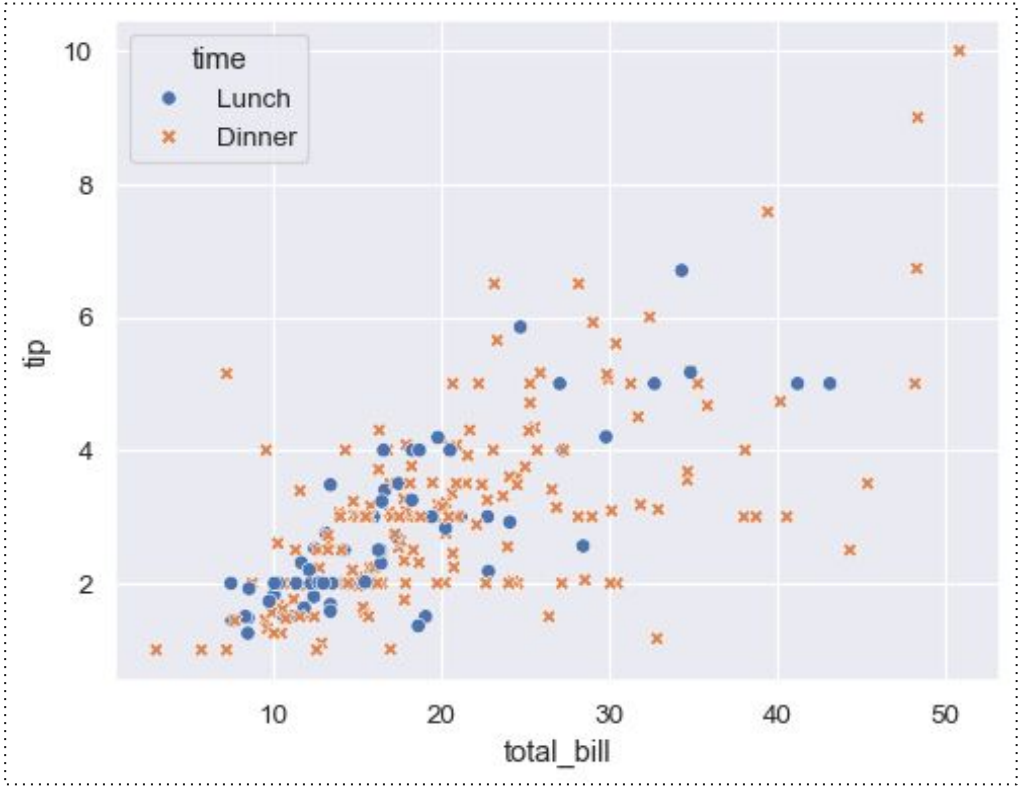




Data Visualization: Scatter and Bubble Plot

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4





Data Visualization: Exercise

A student recorded their **average daily screen time (in hours)** for five apps:

App	Male Users	Female Users	No. of Hours (Avg.)
WhatsApp	40	45	3
Instagram	30	50	2.5
YouTube	35	30	4
Telegram	20	25	1.5
Chrome	25	20	2

Create **two pie charts side by side** — one for *male* users and one for *female* users.

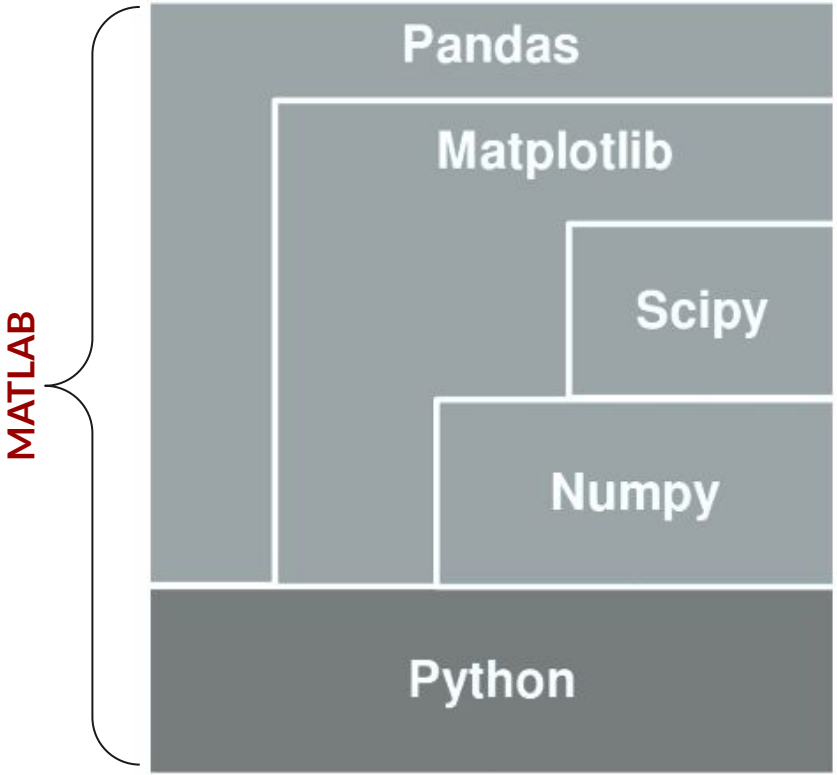
- Each chart should show the proportion of app usage within that gender group.
- Use the same colors for corresponding apps across both charts.
- Add titles: *"App Usage by Males"* and *"App Usage by Females"*.

Create a **grouped bar plot** comparing the number of male and female users for each app above.

Create a **multi-line plot** showing how the screen time for each app changes over 7 days (Day 1–Day 7).

- Use a separate line for each app with different colors and markers.
- Label the x-axis as *"Day"* and y-axis as *"Screen Time (hours)"*.
- Add a legend to identify apps.
  Identify which app shows the **highest variation** in usage over the week.

Python Libraries: Data



Capable of Data Manipulation and Analysis. Offers operations for manipulating numerical tables/tabular data and time series.

Plotting Library for Python worked under mathematical modules; NumPy and SciPy

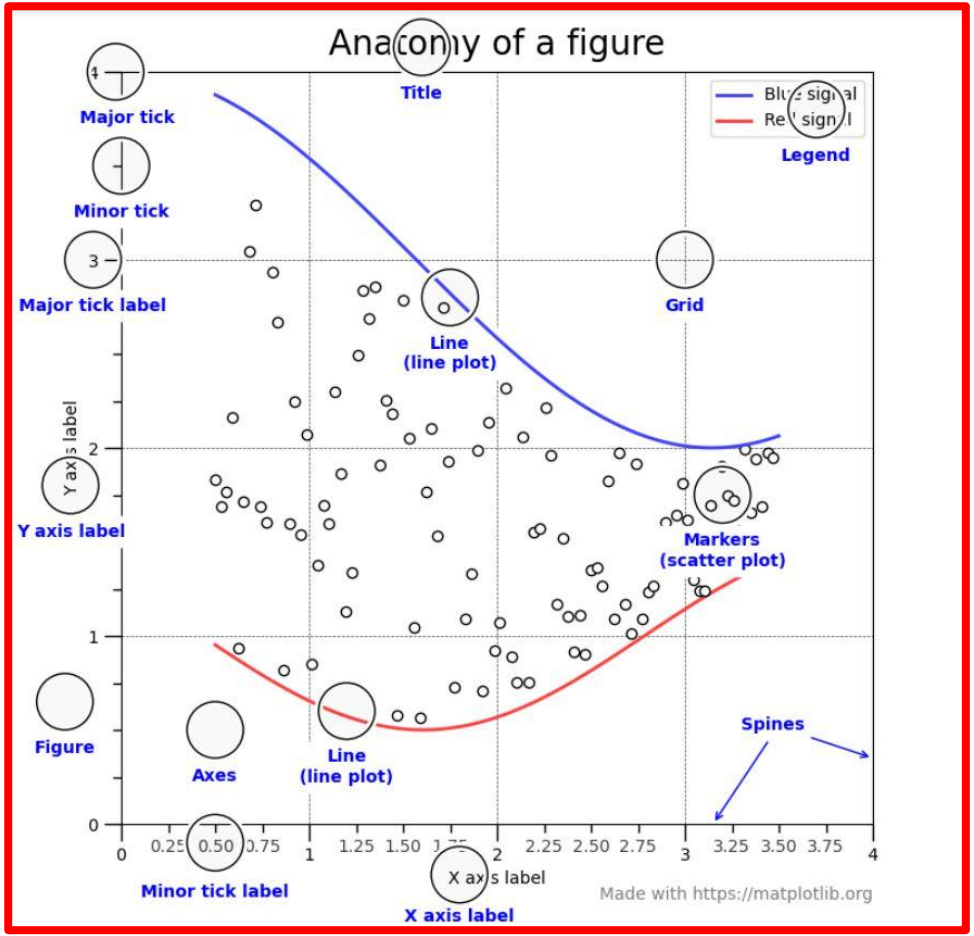
Based on NumPy, Extends the functional capabilities of NumPy

Provides basic Data Structure using n-dimensional arrays and matrices. Create and manipulate these data structures

MATLAB IS NOT FREE

Introducing MatPlotLib/Seaborn

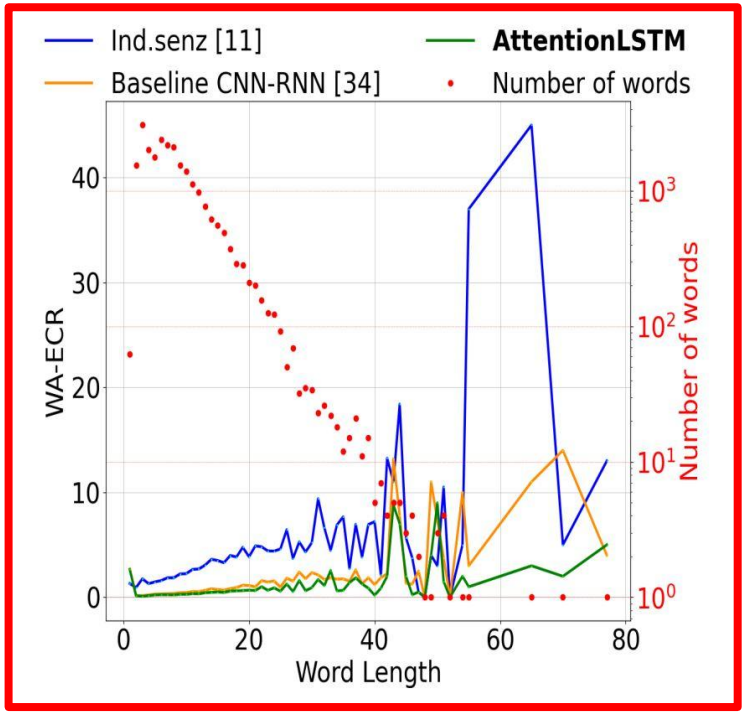
Data Visualization: Figure Anatomy



Data Visualization: Plotting

- Axis ranges: must be common if you want to compare figures
- Axis scale: linear or log, based on range of values
- Axis labels: always label the axis.
- Font: Use font size which is clearly visible.
- Give name and units. e.g. number, length, response time (sec)
- Axis tick labels: format it right

  eg. 0.000000034m vs 34nm, 100000 vs 10^5



Visualization Library: MatPlotLib

```
from matplotlib import pyplot as plt

import matplotlib.pyplot as plt

from matplotlib import pyplot
```

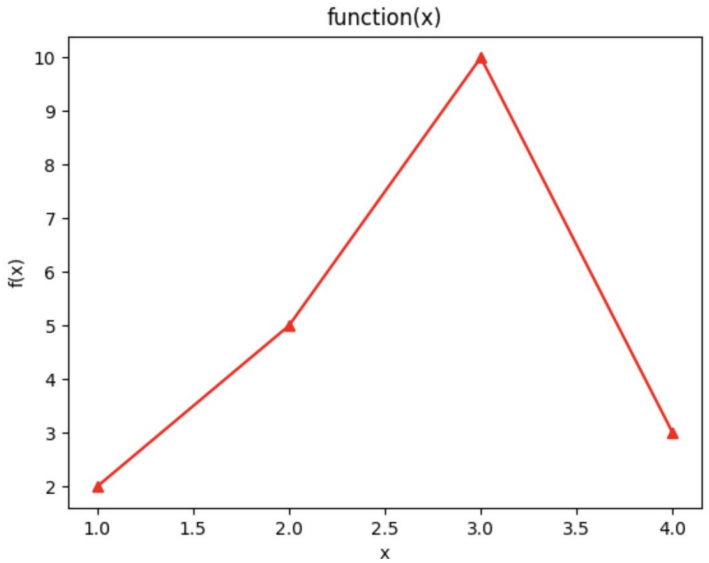
```
import math
from matplotlib import pyplot
```

Data Example

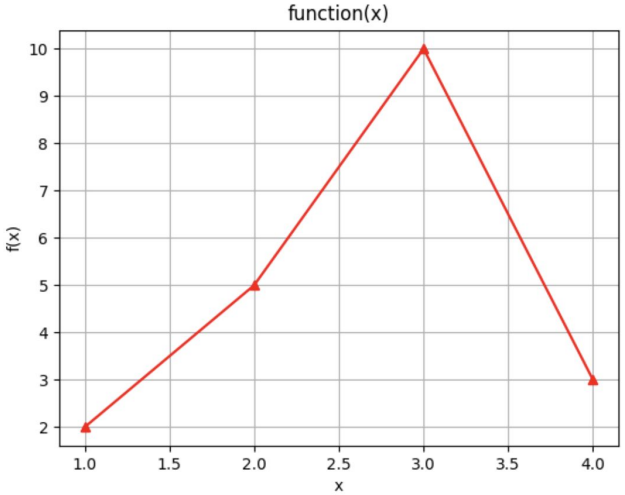
Student	Age	Marks	Height (cm)	Gender	Favorite Subject	Grade
A	18	85	165	Male	Math	A
B	19	90	170	Female	Science	A
C	18	76	160	Male	English	B
D	20	88	172	Female	Math	A
E	19	82	168	Male	Science	B

Visualization Library: MatPlotLib

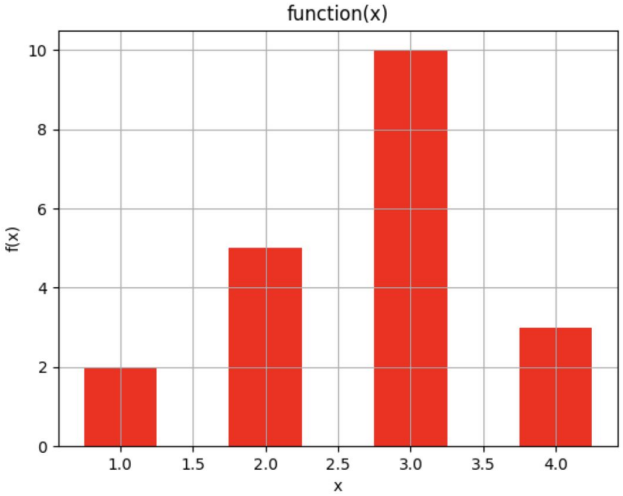
```
1   import matplotlib.pyplot as plt
2
3   x = [1, 2, 3, 4]
4   y = [2, 5, 10, 3]
5
6   plt.title('function(x)')
7   plt.xlabel('x')
8   plt.ylabel('f(x)')
9   #plt.grid(True)
10  plt.plot( *args: x,y, color='r',marker='^')
11  plt.show()
12
```



Visualization Library: MatPlotLib

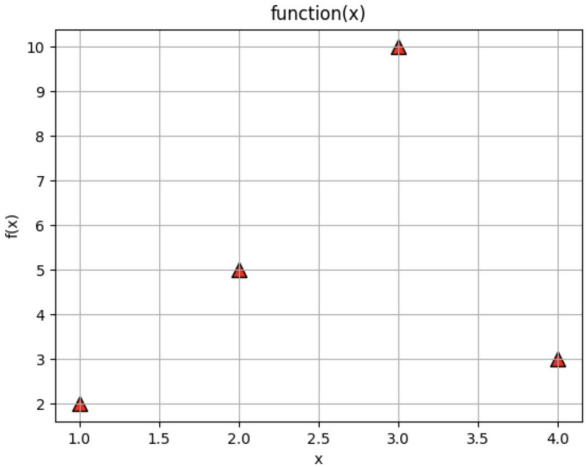


Line



Bar

```
plt.bar(x,y, color='r',width=0.5)
```

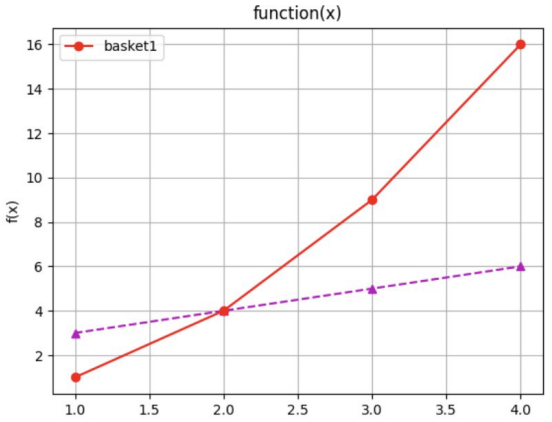


Scatter

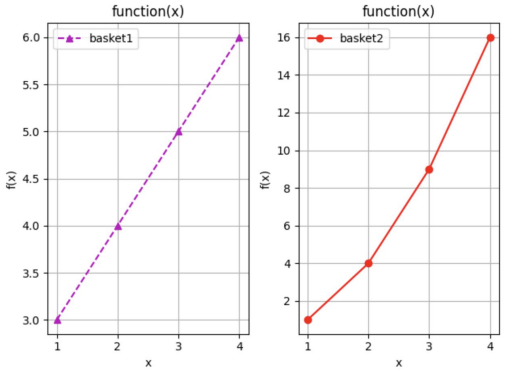
```
plt.scatter(x, y, color='r', marker='^', edgecolor='k', s=100, label='basket1')
```

Visualization Library: MatPlotLib

```python
import matplotlib.pyplot as plt

x = [1, 2, 3, 4]
y = [2, 5, 10, 3]



# bar plot
# plt.bar(x,y, color='r',width=0.5)

fig = plt.figure()
# Plot1
plt.plot( *args: x,[i+2 for i in x],color='m',marker='^', linestyle='--', linewidth=1.5)
# Plot2
plt.plot( *args: x, [i**2 for i in x], color='r', marker='o',  label='basket1')
plt.legend(loc='best')
plt.title('function(x)')
plt.xlabel('x')
plt.ylabel('f(x)')
plt.grid(True)
plt.show()
plt.close(fig)
```

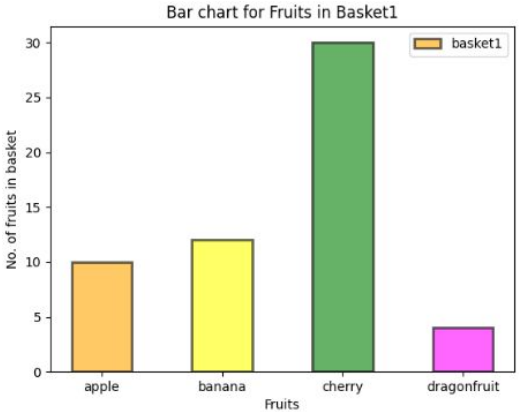



```python
fig = plt.figure()
plt.subplot( *args: 1, 2, 1)
# Plot1
plt.plot( *args: x,[i+2 for i in x],color='m',marker='^', linestyle='--', linewidth=1.5, label='basket1')
plt.legend(loc='best')
plt.title('function(x)')
plt.xlabel('x')
plt.ylabel('f(x)')
plt.grid(True)
# Plot2
plt.subplot( *args: 1, 2, 2)
plt.plot( *args: x, [i**2 for i in x], color='r', marker='o',  label='basket2')
plt.legend(loc='best')
plt.title('function(x)')
plt.xlabel('x')
plt.ylabel('f(x)')
plt.grid(True)
plt.show()
plt.close(fig)
```

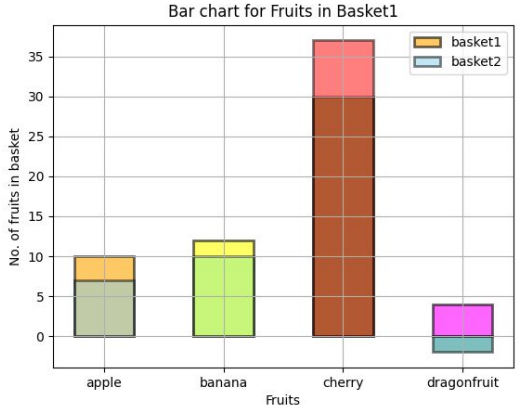


Visualization Library: MatPlotLib

```python
import matplotlib.pyplot as plt

x = ['apple', 'banana', 'cherry', 'dragonfruit']
y = [10, 12, 30, 4]

c = ['orange', 'yellow', 'green', 'magenta']
plt.bar(x, y, color=c, align='center', edgecolor='black', linewidth=2, alpha=0.6, label = 'basket1')
plt.legend()
plt.title("Bar chart for Fruits in Basket1")
plt.xlabel("Fruits")
plt.ylabel("No. of fruits in basket")
plt.show()
```

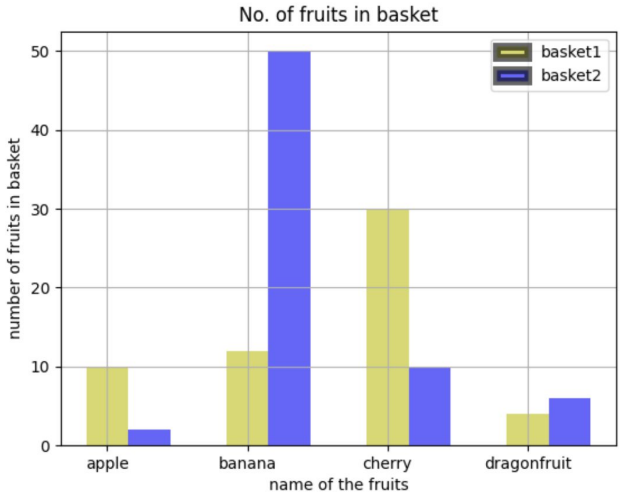


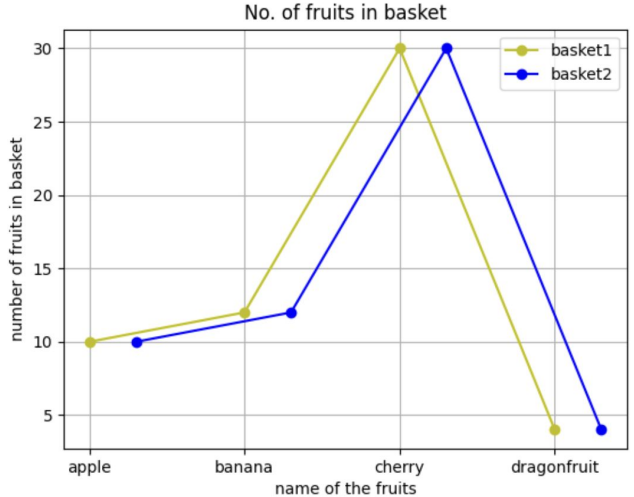

```python
import matplotlib.pyplot as plt

x = ['apple', 'banana', 'cherry', 'dragonfruit']
y = [10, 12, 30, 4]
z = [1.5*i-8 for i in y]
print(z)
c1 = ['orange', 'yellow', 'green', 'magenta']
c2 = ['skyblue', 'lightgreen', 'red', 'teal']
plt.bar(x, y, color=c1, width= 0.5, align='center', edgecolor='black', linewidth=2, alpha=0.6, label = 'basket1')
plt.bar(x, z, color=c2, width= 0.5, align='center', edgecolor='black', linewidth=2, alpha=0.5, label = 'basket2')
plt.legend()
plt.title("Bar chart for Fruits in Basket1")
plt.grid(True)
plt.xlabel("Fruits")
plt.ylabel("No. of fruits in basket")
plt.show()
```



Visualization Library: MatPlotLib

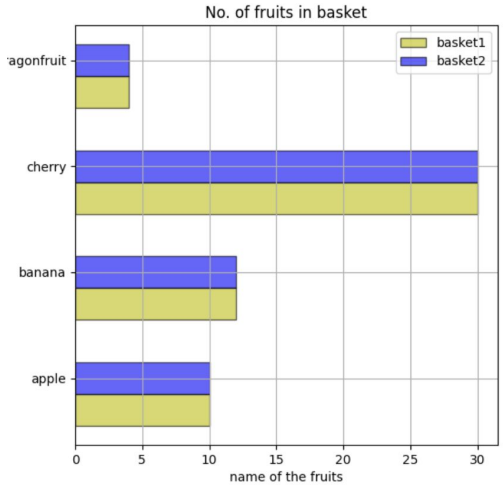
```python
import matplotlib.pyplot as plt

x = ['apple', 'banana', 'cherry', 'dragonfruit']
y = [10, 12, 30, 4]
z = [2, 50, 10, 6]
#c = ['red','yellow','magenta','green']
width = 0.3
x1 = [i+width for i in range(len(x))]

plt.title('No. of fruits in basket')
plt.xlabel('name of the fruits')
plt.ylabel('number of fruits in basket')
plt.grid(True)
plt.bar(x, y, width=0.3,color='y',align='center',edgecolor= 'k',linewidth = 5,linestyle = '',alpha = 0.6,label='basket1')
plt.bar(x1, z, width=0.3,color='b',align='center',edgecolor= 'k',linewidth = 5,linestyle = '',alpha = 0.6,label='basket2')
plt.legend()
plt.show()

```





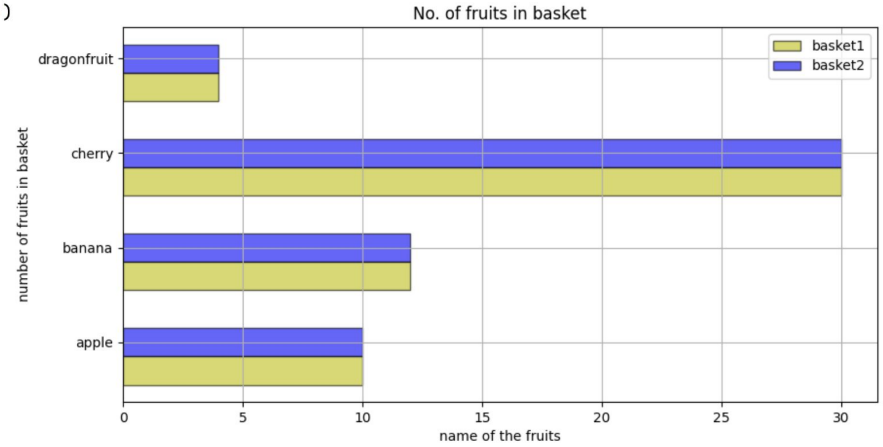
Visualization Library: MatPlotLib

```python
import matplotlib.pyplot as plt

x = ['apple', 'banana', 'cherry', 'dragonfruit']
y = [10, 12, 30, 4]
z = [2, 50, 10, 6]
#c = ['red','yellow','magenta','green']
width = 0.3
x1 = [i+width for i in range(len(x))]

plt.title('No. of fruits in basket')
plt.xlabel('name of the fruits')
plt.ylabel('number of fruits in basket')
plt.grid(True)
#plt.bar(x, y, width=0.3,color='y',align='center',edgecolor= 'k',linewidth = 5,l
#plt.bar(x1, z, width=0.3,color='b',align='center',edgecolor= 'k',linewidth = 5,
plt.barh(x, y,height=0.3, color='y',edgecolor= 'k',alpha = 0.6,label='basket1')
plt.barh(x1, y,height=0.3, color='b',edgecolor= 'k',alpha = 0.6,label='basket2')
plt.yticks(x1,x)
plt.legend()
plt.show()
```



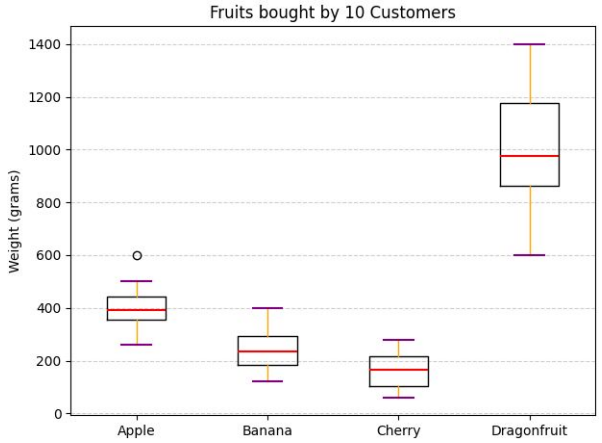
Visualization Library: MatPlotLib

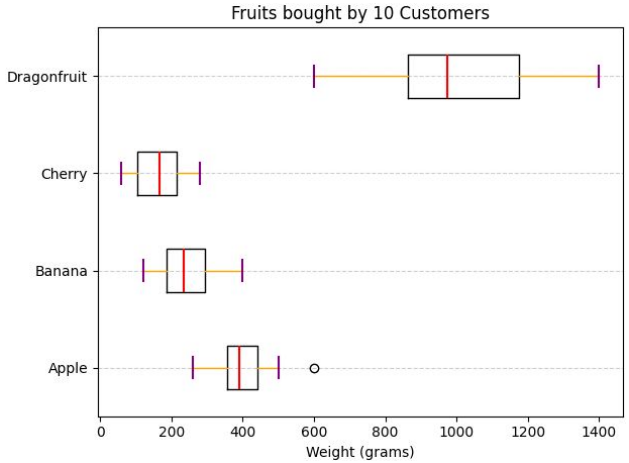
```python
import matplotlib.pyplot as plt

x = ['apple', 'banana', 'cherry', 'dragonfruit']
y = [10, 12, 30, 4]
z = [2, 50, 10, 6]
#c = ['red','yellow','magenta','green']
width = 0.3
x1 = [i+width for i in range(len(x))]

plt.figure(figsize=(10,5))
plt.title('No. of fruits in basket')
plt.xlabel('name of the fruits')
plt.ylabel('number of fruits in basket')
plt.grid(True)
#plt.bar(x, y, width=0.3,color='y',align='center',edgecolor= 'k',linewidth = 5,linestyle =
#plt.bar(x1, z, width=0.3,color='b',align='center',edgecolor= 'k',linewidth = 5,linestyle
plt.barh(x, y,height=0.3, color='y',edgecolor= 'k',alpha = 0.6,label='basket1')
plt.barh(x1, y,height=0.3, color='b',edgecolor= 'k',alpha = 0.6,label='basket2')
plt.yticks(x1,x)
plt.legend()
plt.show()
```



Visualization Library: MatPlotLib

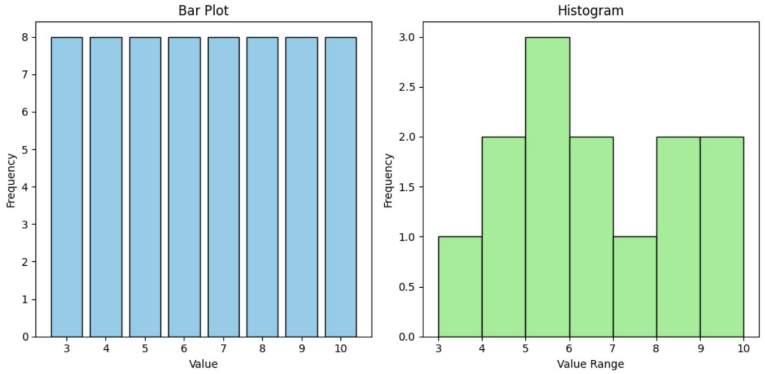
```python
import matplotlib.pyplot as plt

data = [
    [260, 320, 400, 350, 500, 420, 380, 600, 450, 370],
    [120, 150, 200, 180, 250, 300, 220, 270, 320, 400],
    [60, 80, 100, 120, 150, 180, 200, 220, 250, 280],
    [600, 700, 850, 900, 950, 1000, 1100, 1200, 1300, 1400]
]
labels = ['Apple', 'Banana', 'Cherry', 'Dragonfruit']

plt.boxplot(
    data,
    tick_labels=labels,
    capprops=dict(color='purple', linewidth=1.5),
    boxprops=dict(color='black'),
    whiskerprops=dict(color='orange', linewidth=1),
    medianprops=dict(color='red', linewidth=1.5)
)

plt.title("Fruits bought by 10 Customers")
plt.ylabel("Weight (grams)")
plt.grid(axis='y', linestyle='--', alpha=0.6)
plt.tight_layout()
plt.show()
```





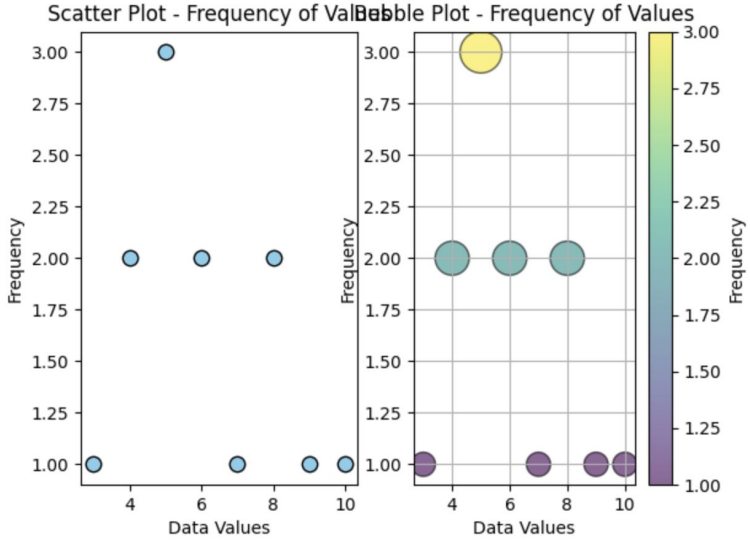
Visualization Library: MatPlotLib

```
24      data = [3, 4, 4, 5, 5, 5, 6, 6, 7, 8, 8, 9, 10]
25      unique = set(data)
26      # --- Bar Plot ---
27      plt.subplot( *args: 1, 2, 1)
28      plt.bar(list(unique), len(unique), color='skyblue', edgecolor='black')
29      plt.title("Bar Plot")
30      plt.xlabel("Value")
31      plt.ylabel("Frequency")
32
33      # --- Histogram ---
34      plt.subplot( *args: 1, 2, 2)
35      plt.hist(data, bins=7, color='lightgreen', edgecolor='black')
36      plt.title("Histogram")
37      plt.xlabel("Value Range")
38      plt.ylabel("Frequency")
39
40      plt.tight_layout()
41      plt.show()
```



Visualization Library: MatPlotLib

```
43    data = [3, 4, 4, 5, 5, 5, 6, 6, 7, 8, 8, 9, 10]
44    # Find unique values and their frequencies (without numpy)
45    unique = sorted(set(data))
46    counts = [data.count(i) for i in unique]
47    plt.subplot( *args: 1, 2, 1)
48    plt.scatter(unique, counts, color='skyblue', marker='o', edgecolor='black', s=80)
49    plt.title("Scatter Plot - Frequency of Values")
50    plt.xlabel("Data Values")
51    plt.ylabel("Frequency")
52
53    # Bubble sizes proportional to frequency
54    sizes = [count * 200 for count in counts]  # adjust scale as needed
55    colors = counts  # color based on frequency
56    plt.subplot( *args: 1, 2, 2)
57    plt.scatter(unique, counts, s=sizes, c=colors, cmap='viridis',alpha=0.6, edgecolor='black')
58
59    plt.title("Bubble Plot - Frequency of Values")
60    plt.xlabel("Data Values")
61    plt.ylabel("Frequency")
62    plt.colorbar(label="Frequency")
63    plt.grid(True)
64    plt.show()
```



Visualization Library: MatPlotLib

```
import math
from matplotlib import pyplot

#picking 1000 points between -10 and 10, and multiplying with pi
xList = list(range(-500,500))#still need to multiply with (1/50) and pi

yList = []
```

Visualization Library: MatPlotLib

```
import math
from matplotlib import pyplot

#picking 1000 points between -10 and 10, and multiplying with pi
xList = list(range(-500,500))#still need to multiply with (1/50) and pi

yList = []

#multiplying x with pi and 1/50, saving sine of the converted values
for i in range(len(xList)):
    xList[i] *= (math.pi * (1/50))
```

Visualization Library: MatPlotLib

```
import math
from matplotlib import pyplot

#picking 1000 points between -10 and 10, and multiplying with pi
xList = list(range(-500,500))#still need to multiply with (1/50) and pi

yList = []

#multiplying x with pi and 1/50, saving sine of the converted values
for i in range(len(xList)):
    xList[i] *= (math.pi * (1/50))
    yList.append(math.sin(xList[i]))
print(min(xList), max(xList))
```

Visualization Library: MatPlotLib

```
import math
from matplotlib import pyplot

#picking 1000 points between -10 and 10, and multiplying with pi
xList = list(range(-500,500))#still need to multiply with (1/50) and pi

yList = []

#multiplying x with pi and 1/50, saving sine of the converted values
for i in range(len(xList)):
    xList[i] *= (math.pi * (1/50))
    yList.append(math.sin(xList[i]))
print(min(xList), max(xList))
pyplot.plot(xList, yList) # makes the line plot
```

Visualization Library: MatPlotLib

```
import math
from matplotlib import pyplot

#picking 1000 points between -10 and 10, and multiplying with pi
xList = list(range(-500,500))#still need to multiply with (1/50) and pi

yList = []

#multiplying x with pi and 1/50, saving sine of the converted values
for i in range(len(xList)):
    xList[i] *= (math.pi * (1/50))
    yList.append(math.sin(xList[i]))
print(min(xList), max(xList))
pyplot.plot(xList, yList) # makes the line plot
pyplot.grid(True) #turning on grid
pyplot.xlabel('x (radians)', fontsize = 24) #adding label to x-axis
pyplot.ylabel('Sin(x)', fontsize = 24) #adding label to y-axis
```

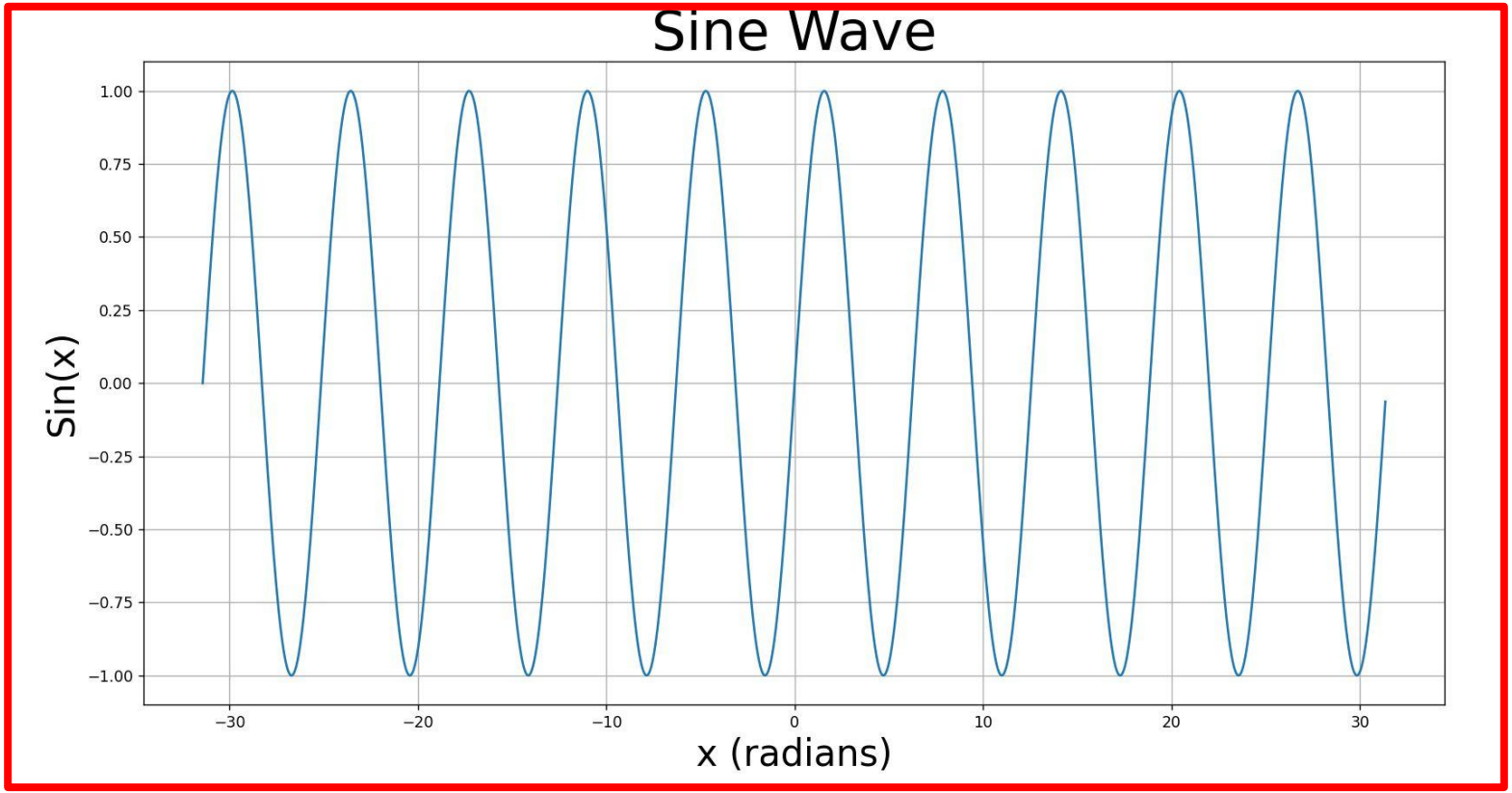
Visualization Library: MatPlotLib

```
import math
from matplotlib import pyplot

#picking 1000 points between -10 and 10, and multiplying with pi
xList = list(range(-500,500))#still need to multiply with (1/50) and pi

yList = []

#multiplying x with pi and 1/50, saving sine of the converted values
for i in range(len(xList)):
    xList[i] *= (math.pi * (1/50))
    yList.append(math.sin(xList[i]))
print(min(xList), max(xList))
pyplot.plot(xList, yList) # makes the line plot
pyplot.grid(True) #turning on grid
pyplot.xlabel('x (radians)', fontsize = 24) #adding label to x-axis
pyplot.ylabel('Sin(x)', fontsize = 24) #adding label to y-axis
pyplot.title('Sine Wave', fontsize = 36) #adding the title
pyplot.show() #you can only see plot if you use this statement
```

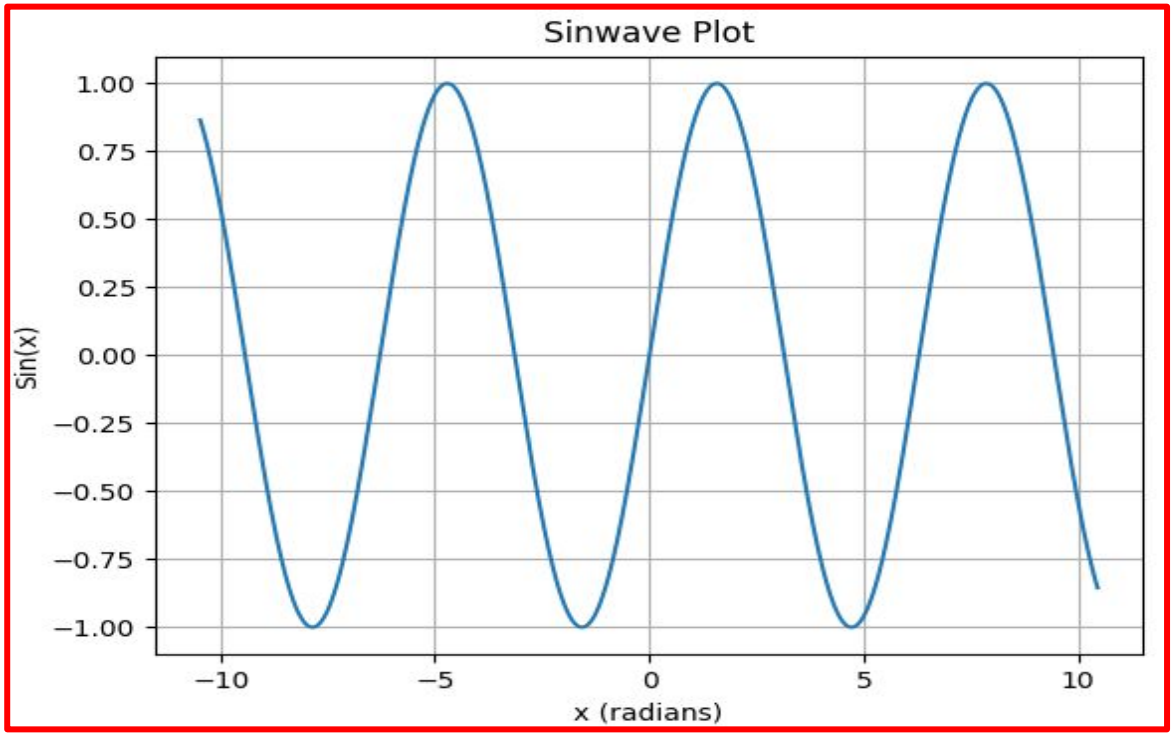
Visualization Library: MatPlotLib

```
import math
from matplotlib import pyplot

#picking 1000 points between -10 and 10, and multiplying with pi
xList = list(range(-500,500))#still need to multiply with (1/50) and pi

yList = []

#multiplying x with pi and 1/50, saving sine of the converted values
for i in range(len(xList)):
    xList[i] *= (math.pi * (1/50))
    yList.append(math.sin(xList[i]))
print(min(xList), max(xList))
pyplot.plot(xList, yList) # makes the line plot
pyplot.grid(True) #turning on grid
pyplot.xlabel('x (radians)', fontsize = 24) #adding label to x-axis
pyplot.ylabel('Sin(x)', fontsize = 24) #adding label to y-axis
pyplot.title('Sine Wave', fontsize = 36) #adding the title
pyplot.show() #you can only see plot if you use this statement
```

Visualization Library: MatPlotLib



Visualization Library: MatPlotLib



1. plt.figure
   Creates a new Plot
   Arguments: figsize, dpi, facecolor (background)
2. plt.plot
   Creates a plot
3. plt.grid
4. plt.xlabel/ylabel
5. plt.xticks/yticks
6. plt.legend
7. plt.title
8. plt.show

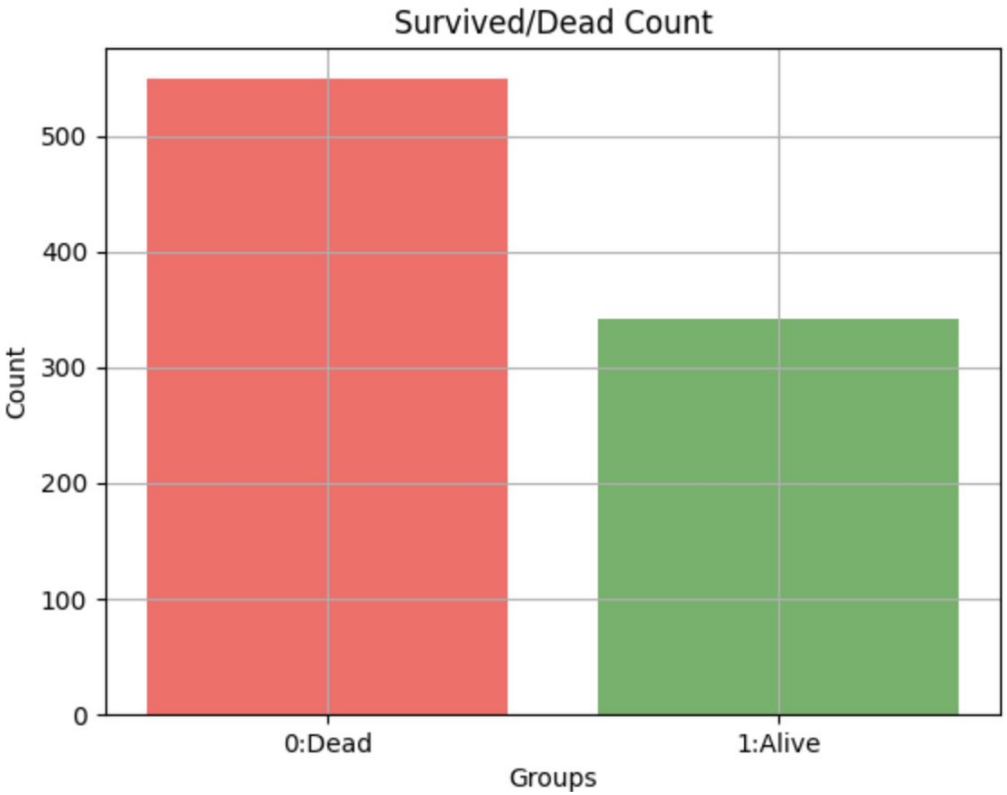
Visualization Library: MatPlotLib

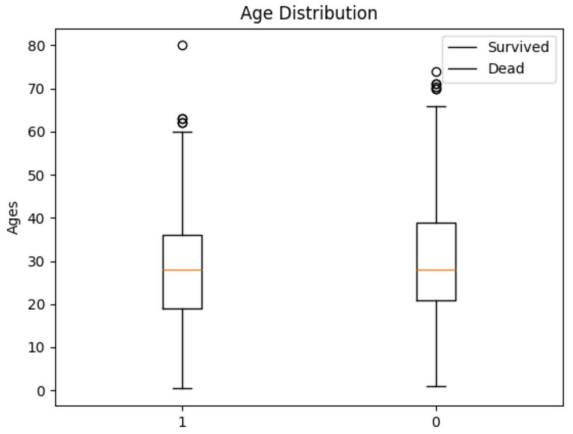
```python
import csv
import matplotlib.pyplot as plt

with open("titanic.csv") as data:
    reader = csv.DictReader(data)
    d1 = list(reader)
dead = 0
alive = 0
ages_survived = []
ages_dead = []
for row in d1:
    try:
        if row["Survived"]== '1':
            alive += 1
            ages_survived.append(float(row["Age"]))
        elif row["Survived"]== '0':
            dead += 1
            ages_dead.append(float(row["Age"]))
    except ValueError:
        pass

plt.figure()
plt.bar( x: ["0:Dead","1:Alive"], height: [dead,alive],color=["red","green"],alpha=0.6)
plt.title("Survived/Dead Count")
plt.xlabel("Groups")
plt.ylabel("Count")
plt.grid(True)
plt.show()
plt.figure()
plt.boxplot( x: [ages_survived, ages_dead],labels=["Survived","Dead"],patch_artist=True)
plt.title("Age Distribution")
plt.ylabel("Ages")
plt.grid(True)
plt.show()
```

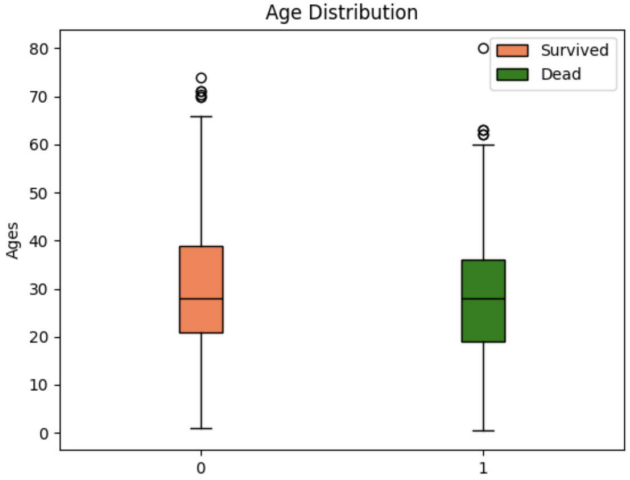
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
1	1	0	3	Braund, Mr. Owen …	male	22	1	0	A/5 21171	7.25	<null>
2	2	1	1	Cumings, Mrs. Joh…	female	38	1	0	PC 17599	71.2833	C85
3	3	1	3	Heikkinen, Miss. …	female	26	0	0	STON/O2. 3101282	7.925	<null>
4	4	1	1	Futrelle, Mrs. Ja…	female	35	1	0	113803	53.1	C123
5	5	0	3	Allen, Mr. Willia…	male	35	0	0	373450	8.05	<null>
6	6	0	3	Moran, Mr. James	male	<null>	0	0	330877	8.4583	<null>
7	7	0	1	McCarthy, Mr. Tim…	male	54	0	0	17463	51.8625	E46
8	8	0	3	Palsson, Master. …	male	2	3	1	349909	21.075	<null>
9	9	1	3	Johnson, Mrs. Osc…	female	27	0	2	347742	11.1333	<null>
10	10	1	2	Nasser, Mrs. Nich…	female	14	1	0	237736	30.0708	<null>
11	11	1	3	Sandstrom, Miss. …	female	4	1	1	PP 9549	16.7	G6
12	12	1	1	Bonnell, Miss. El…	female	58	0	0	113783	26.55	C103
13	13	0	3	Saundercock, Mr. …	male	20	0	0	A/5. 2151	8.05	<null>
14	14	0	3	Andersson, Mr. An…	male	39	1	5	347082	31.275	<null>
15	15	0	3	Vestrom, Miss. Hu…	female	14	0	0	350406	7.8542	<null>
16	16	1	2	Hewlett, Mrs. (Ma…	female	55	0	0	248706	16	<null>
17	17	0	3	Rice, Master. Eug…	male	2	4	1	382652	29.125	<null>
18	18	1	2	Williams, Mr. Cha…	male	<null>	0	0	244373	13	<null>
19	19	0	3	Vander Planke, Mr…	female	31	1	0	345763	18	<null>
20	20	1	3	Masselmani, Mrs. …	female	<null>	0	0	2649	7.225	<null>
21	21	0	2	Fynney, Mr. Josep…	male	35	0	0	239865	26	<null>
22	22	1	2	Beesley, Mr. Lawr…	male	34	0	0	248698	13	D56

Titanic.csv

Visualization Library: MatPlotLib







?