

Learning Progress Review

Week 9

Kelompok 7 - Citizen Data Scientist

Diaz Jubiary - Hermulia Hadie - Desi
Sulistyowati - Farahul Jannah



Key Presentation Point

INTRODUCTION TO DATA VISUALIZATION

1. Karakteristik data
2. Data visualization
3. Importance of data visualization
4. Goals of data visualization
5. Visualize proportion
6. Visualize comparison
7. Visualize over time data

Key Presentation Point

INTERMEDIATE VISUALIZATION

- 1. Matplotlib
- 2. Pyplot
- 3. Axes
- 4. Subplot
- 5. Legend
- 6. Annotation
- 7. Pie chart
- 8. Bar plot
- 9. Histogram
- 10. Scatter Plot
- 11. Boxplots
- 12. Line chart/line plot
- 13. Violin plot



Introduction to Data Visualization

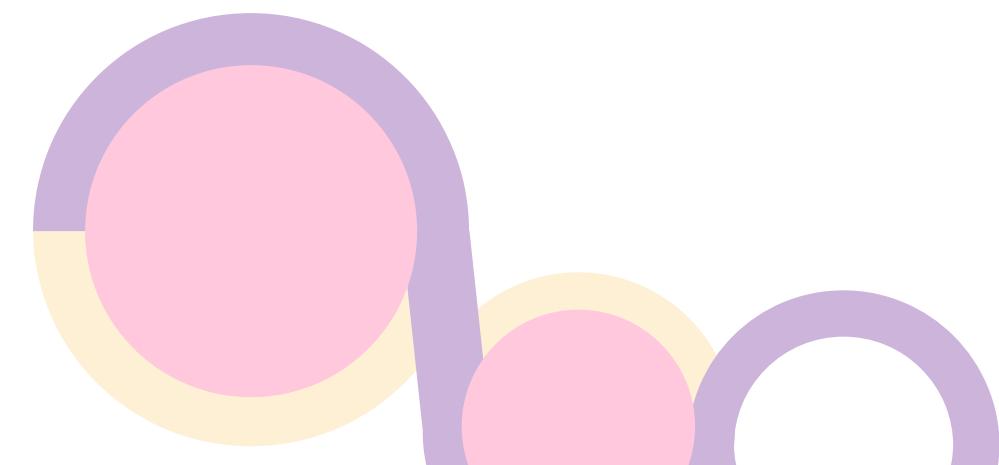
Karakteristik Data

- Source of truth
- Exist everywhere
- Usually abstract (unstructured)
- Need a medium to reveal the “mystery in their mind”

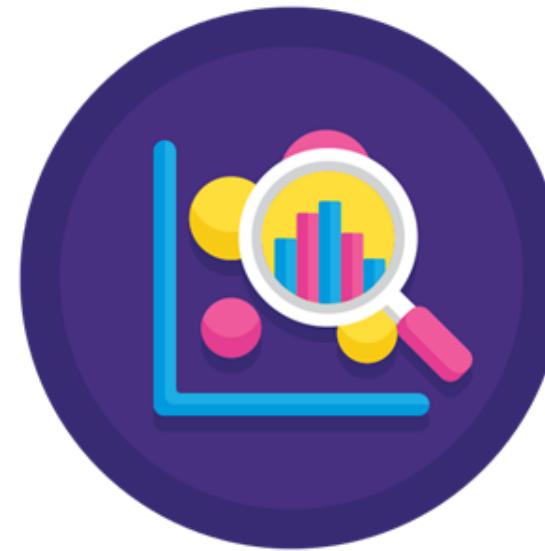
Data Visualization

- Tampilan berupa grafis atau visual dari informasi dan data
- Menyampaikan pesan kepada pembaca dengan sebaik mungkin
- Kombinasi science dan art.

Importance of Data Visualization



Goals of Data Visualization



EXPLORATORY

Melihat hubungan-hubungan
(analisis) dalam data



EXPLANATORY

Mempresentasikan data

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

DV shows Different Perspective

Mean & Variance

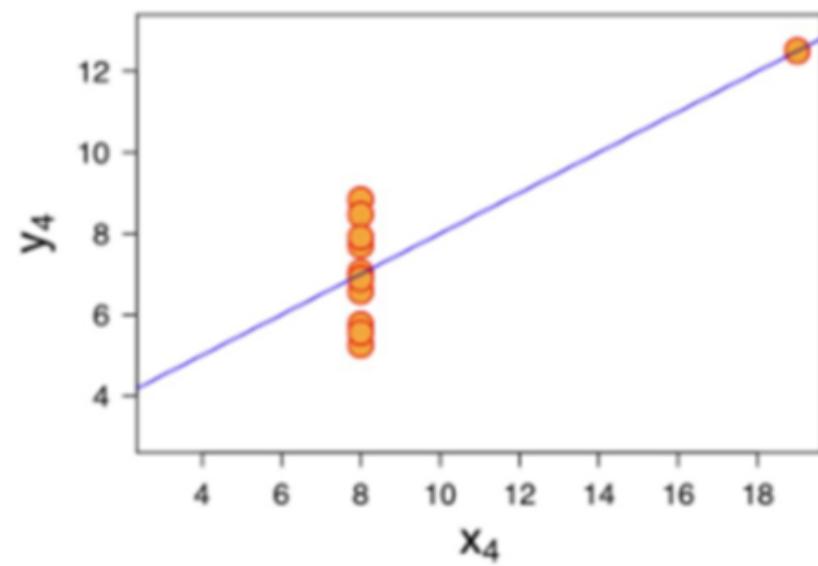
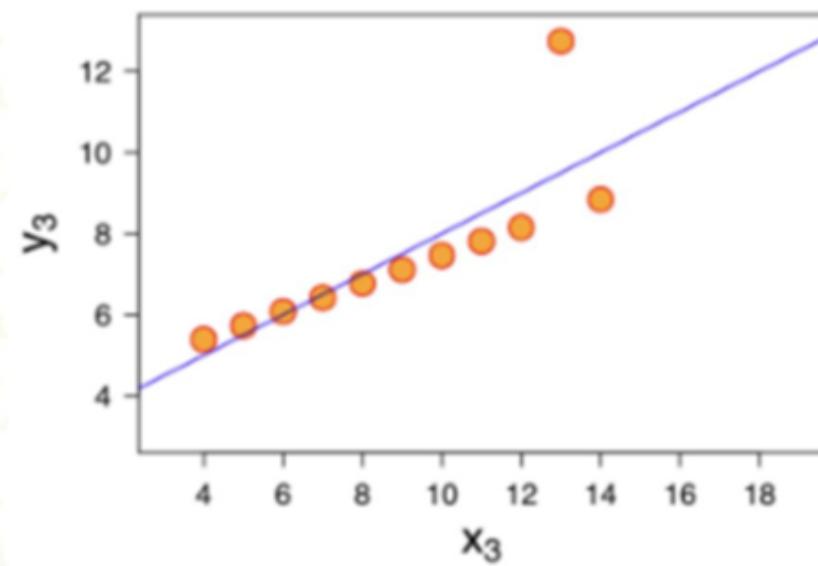
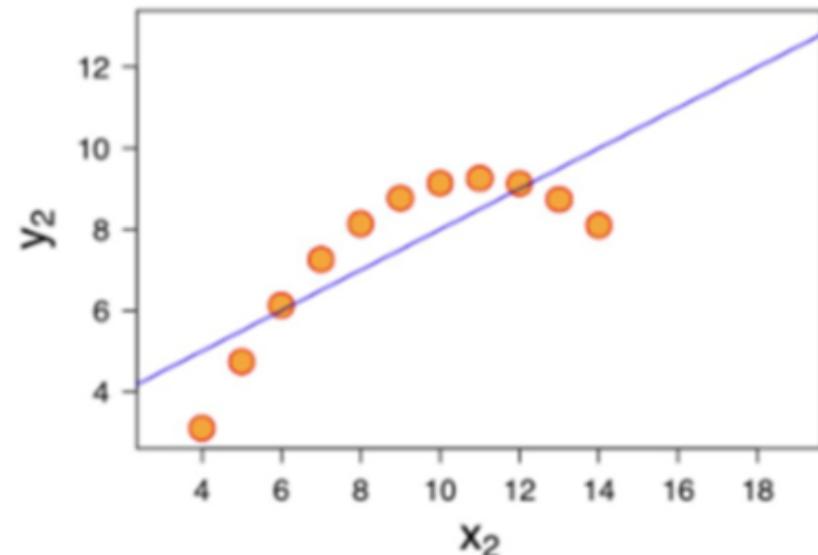
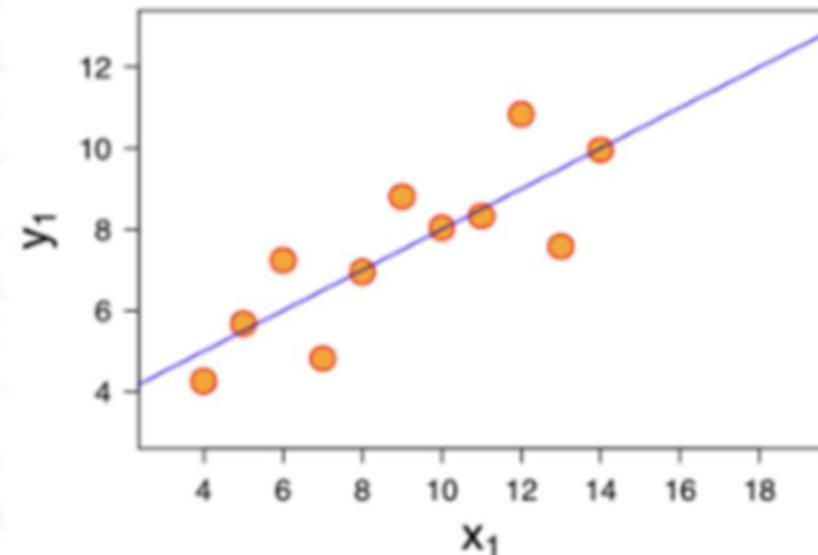
$$\mu_X = 9.0, \sigma_X = 11$$

$$\mu_Y = 7.5, \sigma_Y = 4.125$$

Linear Regression

$$Y = 0.5X + 3$$

$$R^2 = 0.67$$



Mean & Variance
 $\mu_X = 9.0$, $\sigma_X = 11$
 $\mu_Y = 7.5$, $\sigma_Y = 4.125$

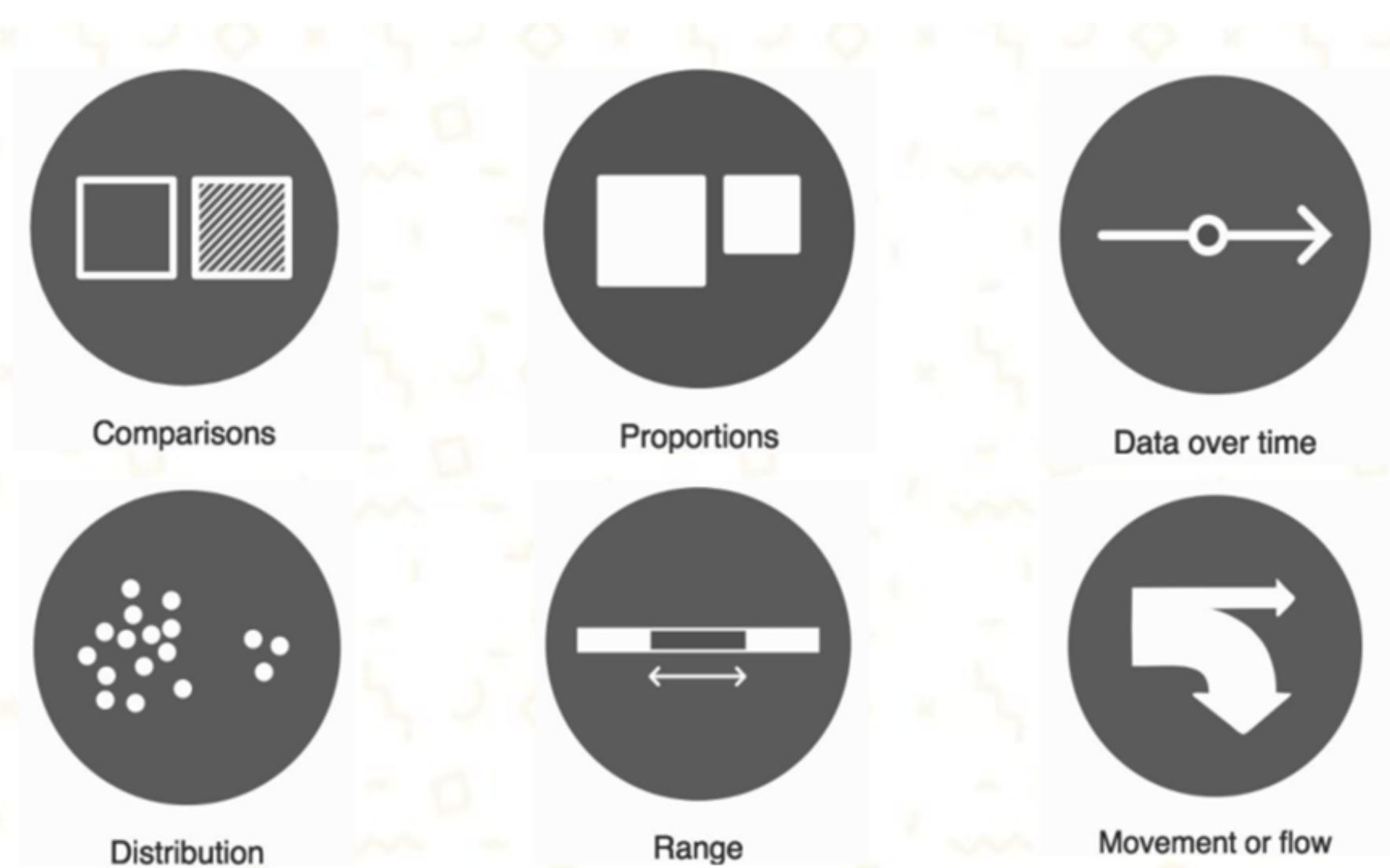
Linear Regression
 $Y = 0.5X + 3$
 $R^2 = 0.67$

DV shows Different Perspective

...datasets from the same statistics can be vastly different and prevent wrong actionables

Effective Visualization

1. Tentukan tujuan
2. Tentukan metrics yang ingin diperlihatkan
3. Pilih jenis chart yang paling representative



Visualize Proportion

Melihat proporsi per bagian dari keseluruhan



Stacked Bar Graph

E.g. "Proportion of transactions per store"



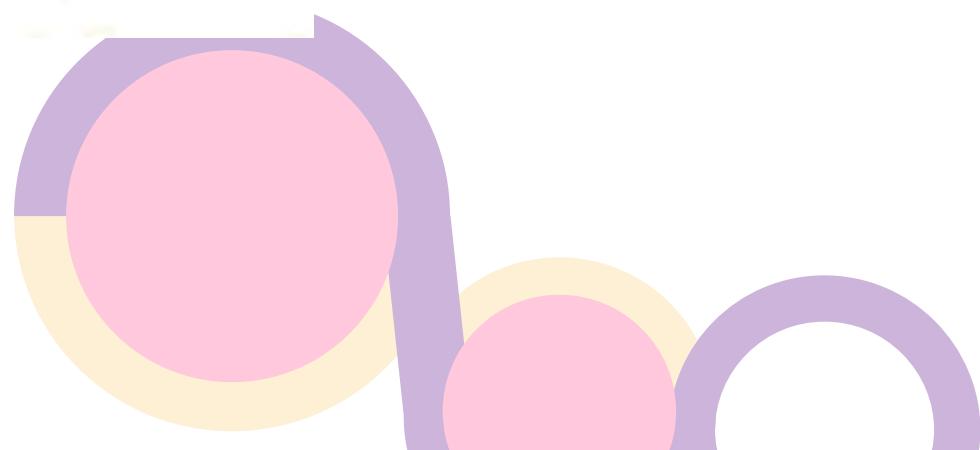
Treemap

E.g. "Proportion of transactions per area"



Pie Chart

E.g. "Proportion of transactions per gender customer"



Visualize Comparison

Melihat kesamaan/perbedaan antar kategori



Bar Chart



Multi-set Bar Chart



Stacked Bar Graph



Heatmap

E.g. "Number of user per segment"

E.g. "Number of purchase happened per product"



Visualize Over Time Data

Melihat perubahan/tren dalam suatu periode waktu



Line Graph

E.g. "# of transacting users per month"



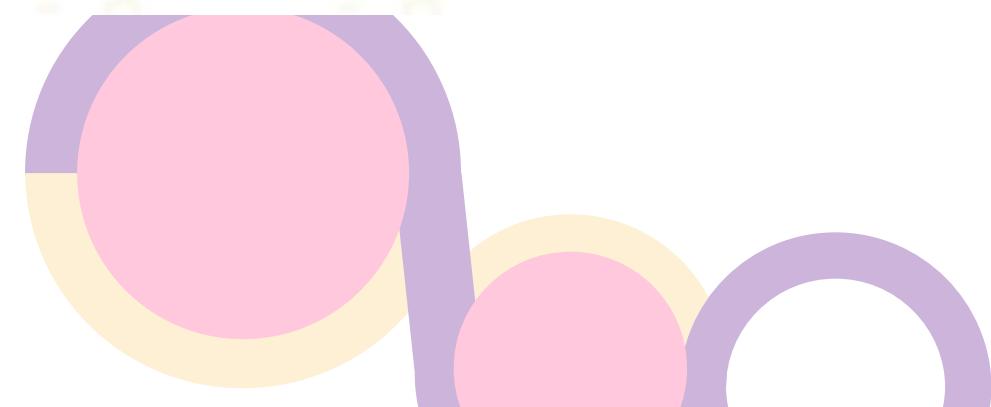
Stacked Area Graph

E.g. "# of transacting users in each store per month"



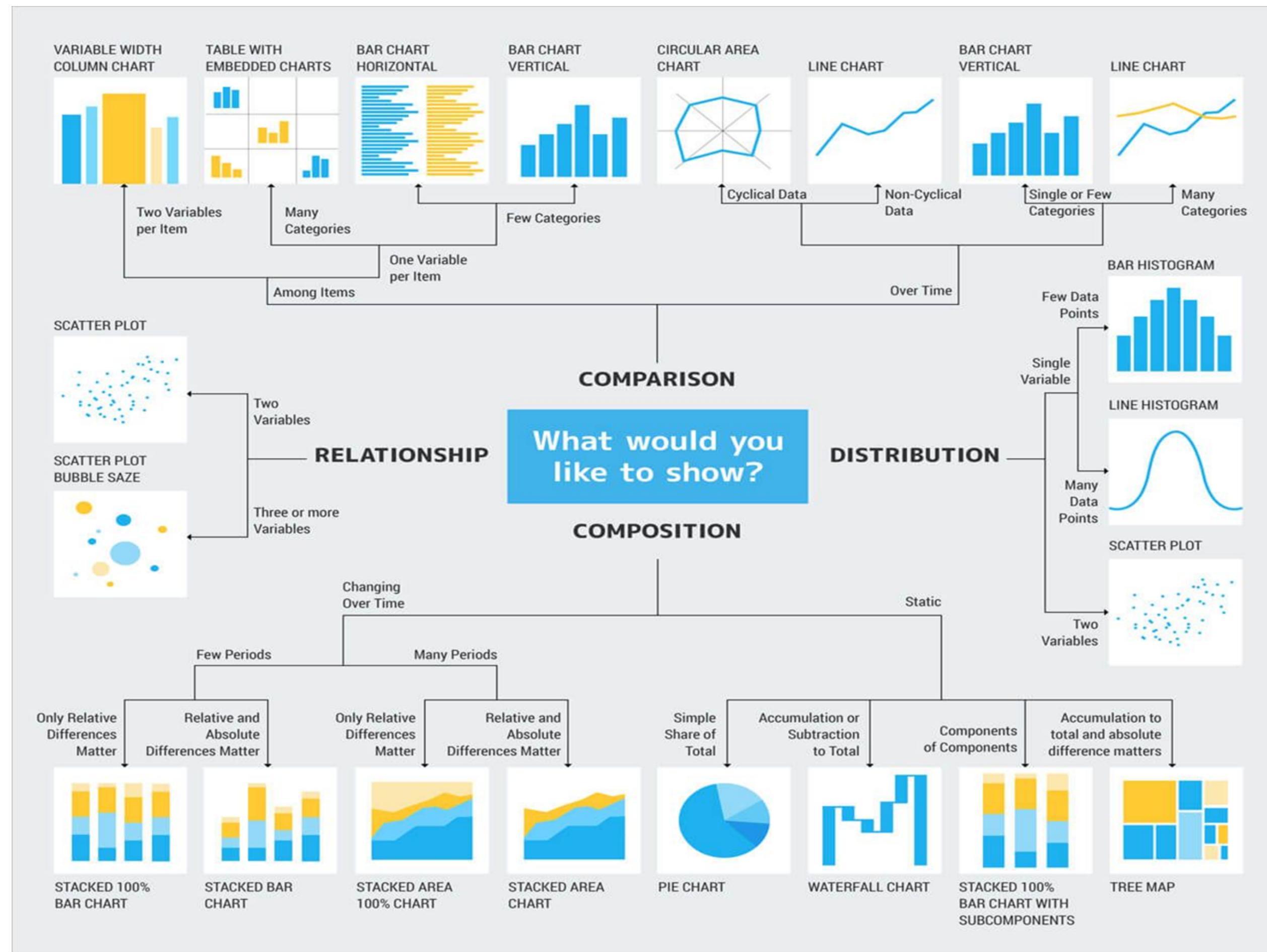
Bubble Chart

E.g. "# of transacting users in each products per month"





Intermediate Visualization



Matplotlib

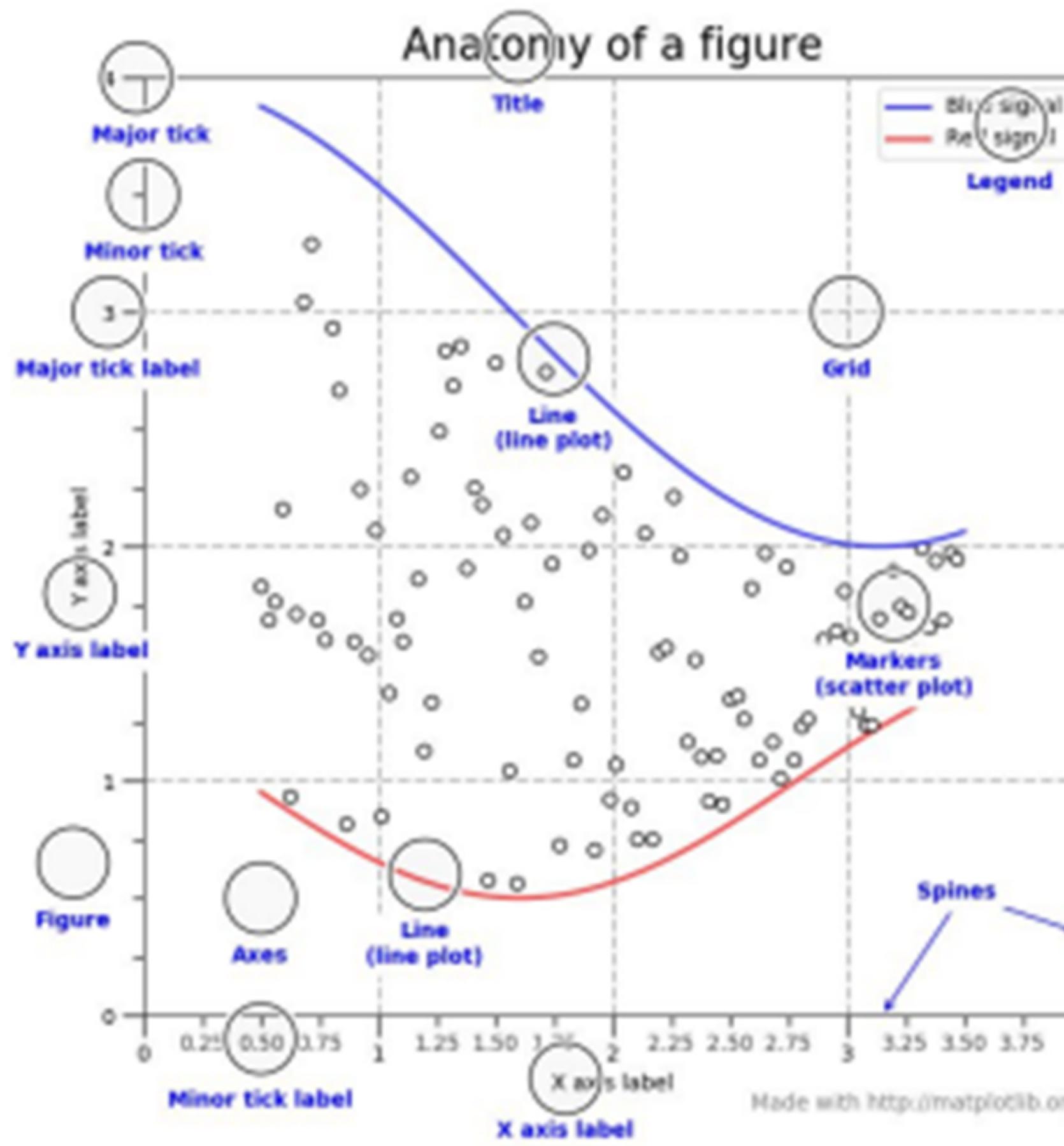
Matplotlib adalah Python 2D plotting library yang menghasilkan gambar berkualitas publikasi dalam berbagai format hardcopy dan environment interaktif di seluruh platform

```
import matplotlib
```

Pyplot

Pyplot adalah modul dari package matplotlib yang memungkinkan kita untuk membuat figures dan axes secara otomatis dan implisit

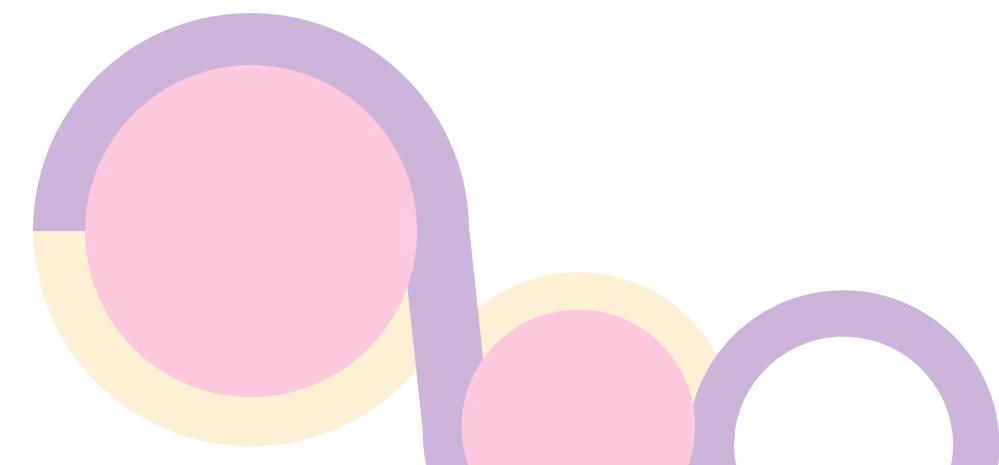
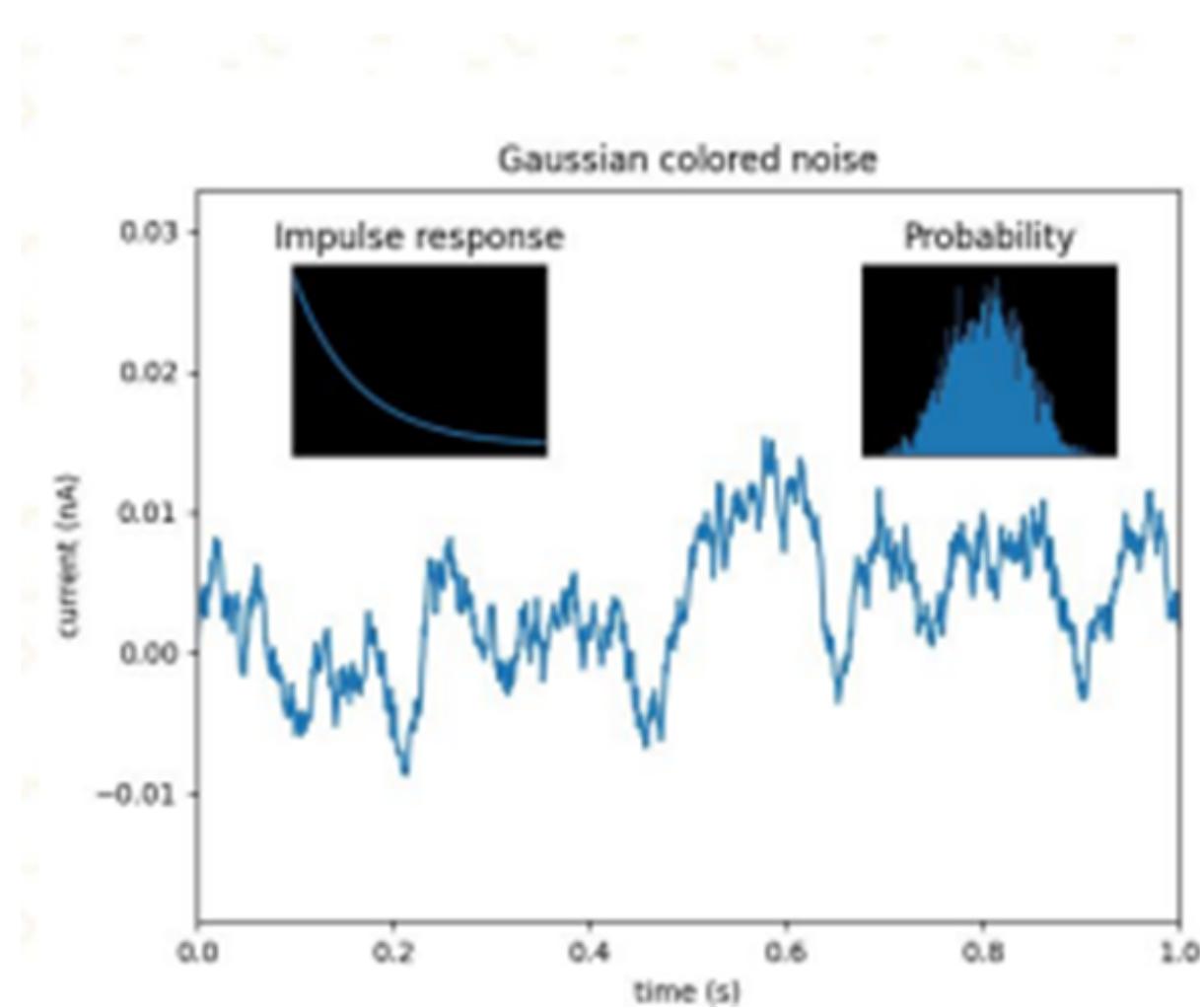
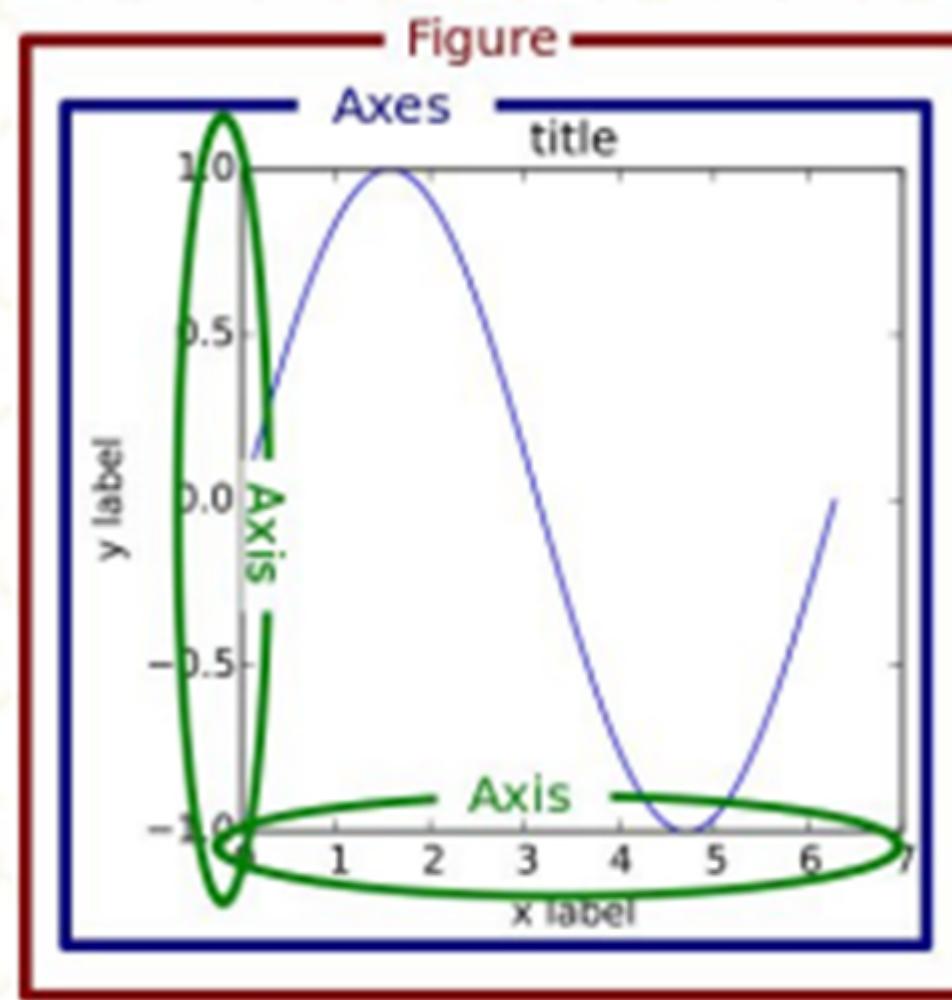
```
Import matplotlib.pyplot as plt
```



Anatomi Figur Matplotlib

Axes

Axes adalah element dari Figure yang memuat: Axis, Tick, Line2D, Text, Polygon, etc., dan rangkaian koordinat sistem. Satu figure pada matplotlib dapat memiliki lebih dari 1 axes



Subplot

Subplot adalah axes yang dituangkan ke dalam grid system

```
plt.subplot(nrows, ncols, axes)
```

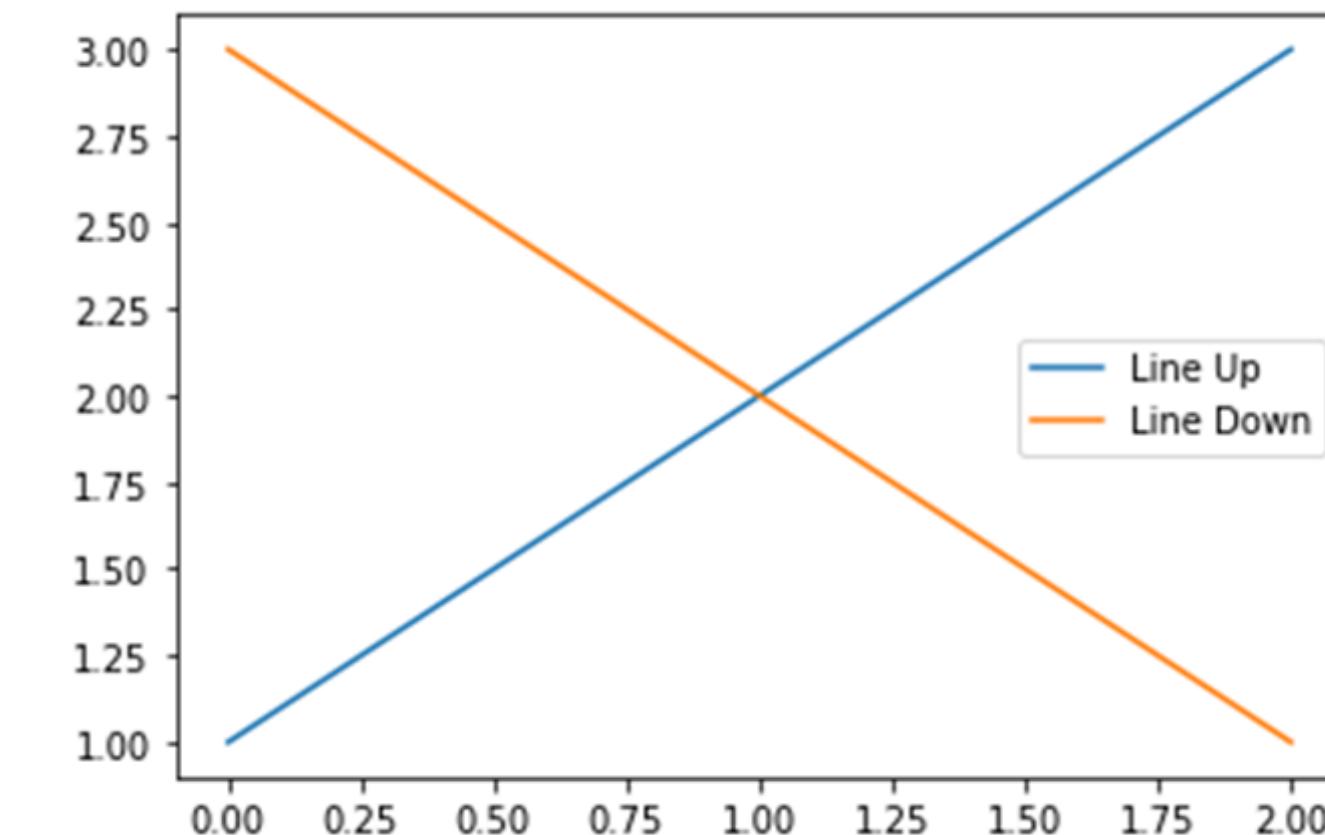
```
fig = plt.figure(figsize=(20,10))
ax1 = fig.add_subplot(121)
ax2 = fig.add_subplot(122)
```

Legend

Fungsi legend untuk meletakkan legend pada axes



```
fig, ax = plt.subplots()
line_up, = ax.plot([1, 2, 3], label='Line 2')
line_down, = ax.plot([3, 2, 1], label='Line 1')
ax.legend([line_up, line_down], ['Line Up', 'Line Down'])
```



Annotation

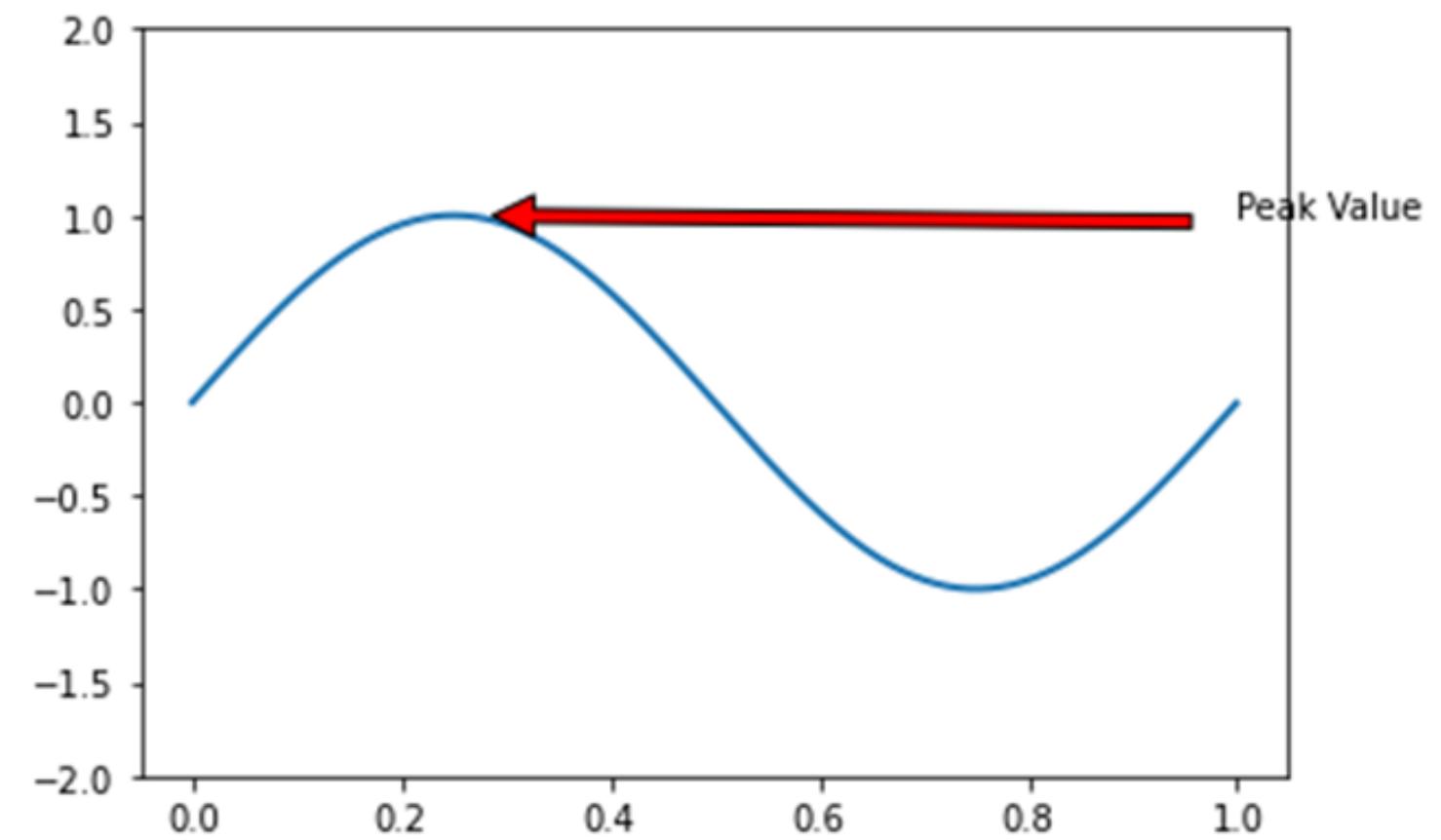
Fungsi annotation adalah untuk memberikan text, panah, atau object lainnya pada axes

```
fig, ax = plt.subplots()

t = np.arange(0.0, 1.0, 0.001)
s = np.sin(2 * np.pi * t)
line = ax.plot(t, s, lw=2)

ax.annotate('Peak Value', xy=(.25, 1),
            xytext=(1, 1), arrowprops=dict(facecolor='red',
                                             shrink=0.05), xycoords="data",)

ax.set_xlim(-2, 2)
plt.show()
```



SHOW PLOT

```
plt.show()
```

CLEAR AN AXIS

```
plt.cla()
```

SAVE FIGURES

```
plt.savefig('foo.png', transparent=True)
```

CLEAR THE ENTIRE FIGURE

```
plt.clf()
```

CLOSE A WINDOW

```
plt.close()
```

Template visualisasi

Matplotlib

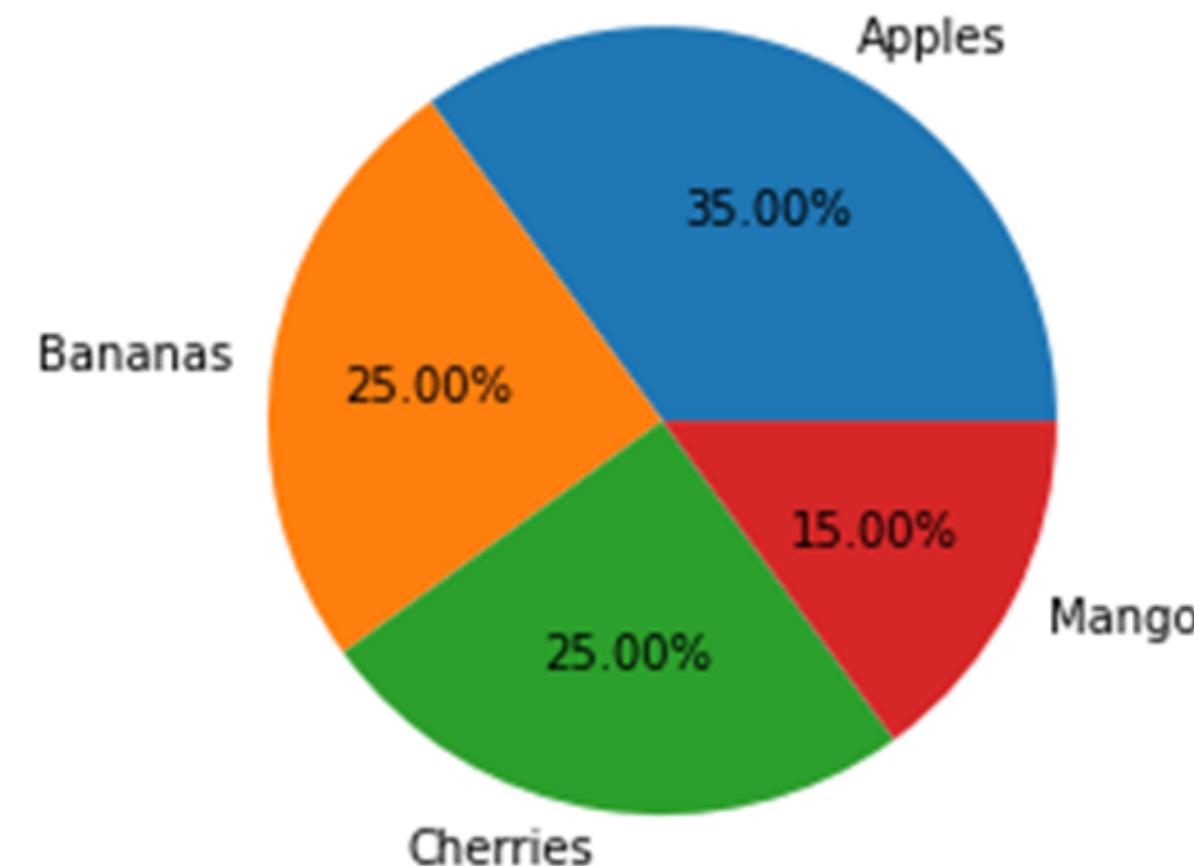
```
x = [1,2,3,4]
y = [10,20,25,30]
fig = plt.figure()
ax = fig.add_subplot(111)
ax.plot(x, y, color='lightblue', linewidth=3)
ax.scatter([2,4,6], [5,15,25], color='darkgreen', marker='^')
ax.set_xlim(1, 6.5)
plt.savefig()
plt.show()
```

Plotting in Matplotlib

Pie chart digunakan untuk menunjukkan komposisi komponen atau menunjukkan bagian-bagian dari sebuah kesatuan kategoris

```
y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Mango"]

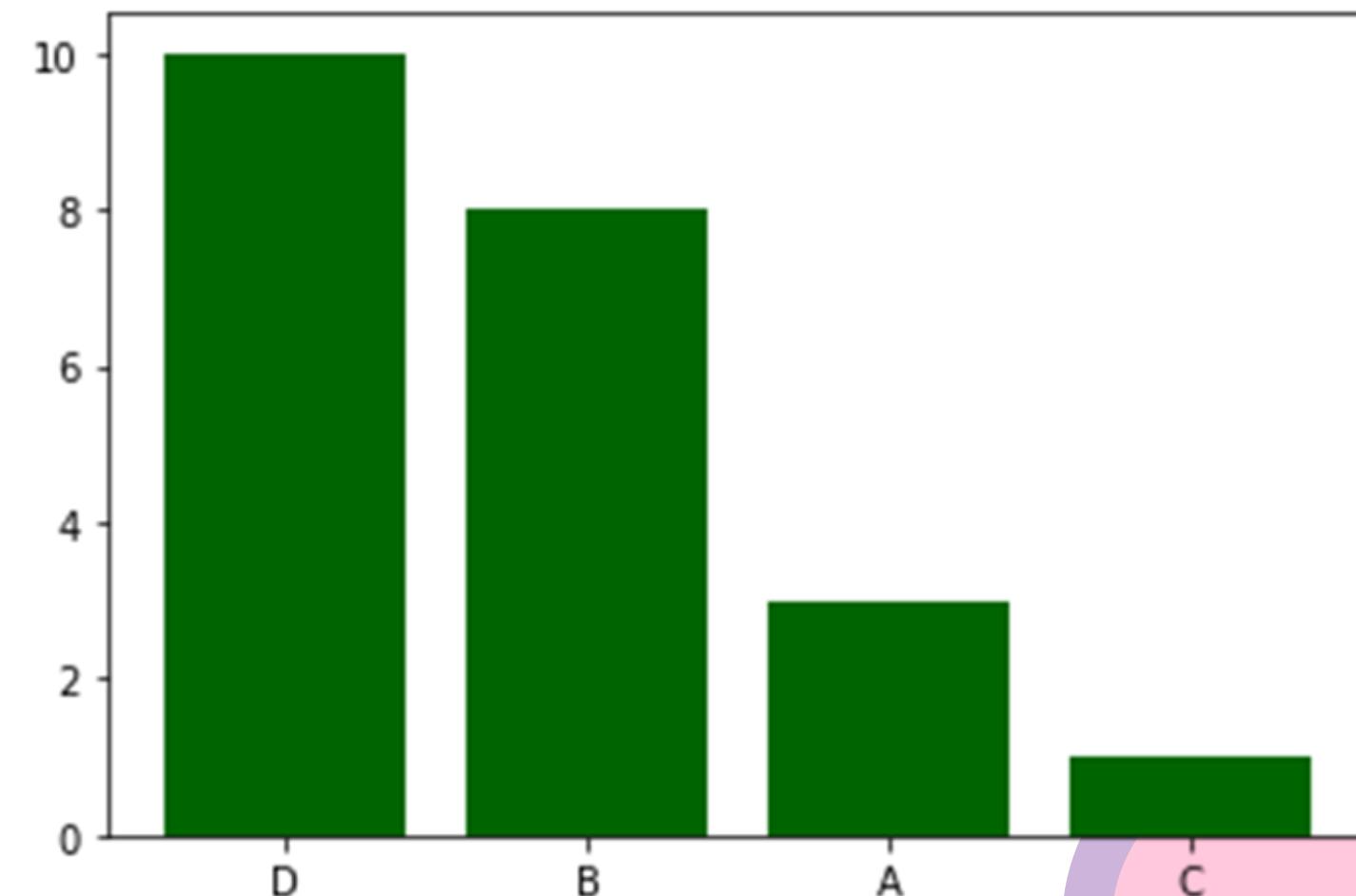
plt.pie(y, labels = mylabels, autopct = '%.2f%%')
plt.show()
```



Plotting in Matplotlib

Barplot atau diagram batang adalah diagram yang berguna untuk menyajikan perbandingan data pada satu atau beberapa variabel data

```
# Colored Barchart  
  
x = np.array(["D", "B", "A", "C"])  
y = np.array([10, 8, 3, 1])  
  
plt.bar(x, y, color = "darkgreen")  
plt.show()
```

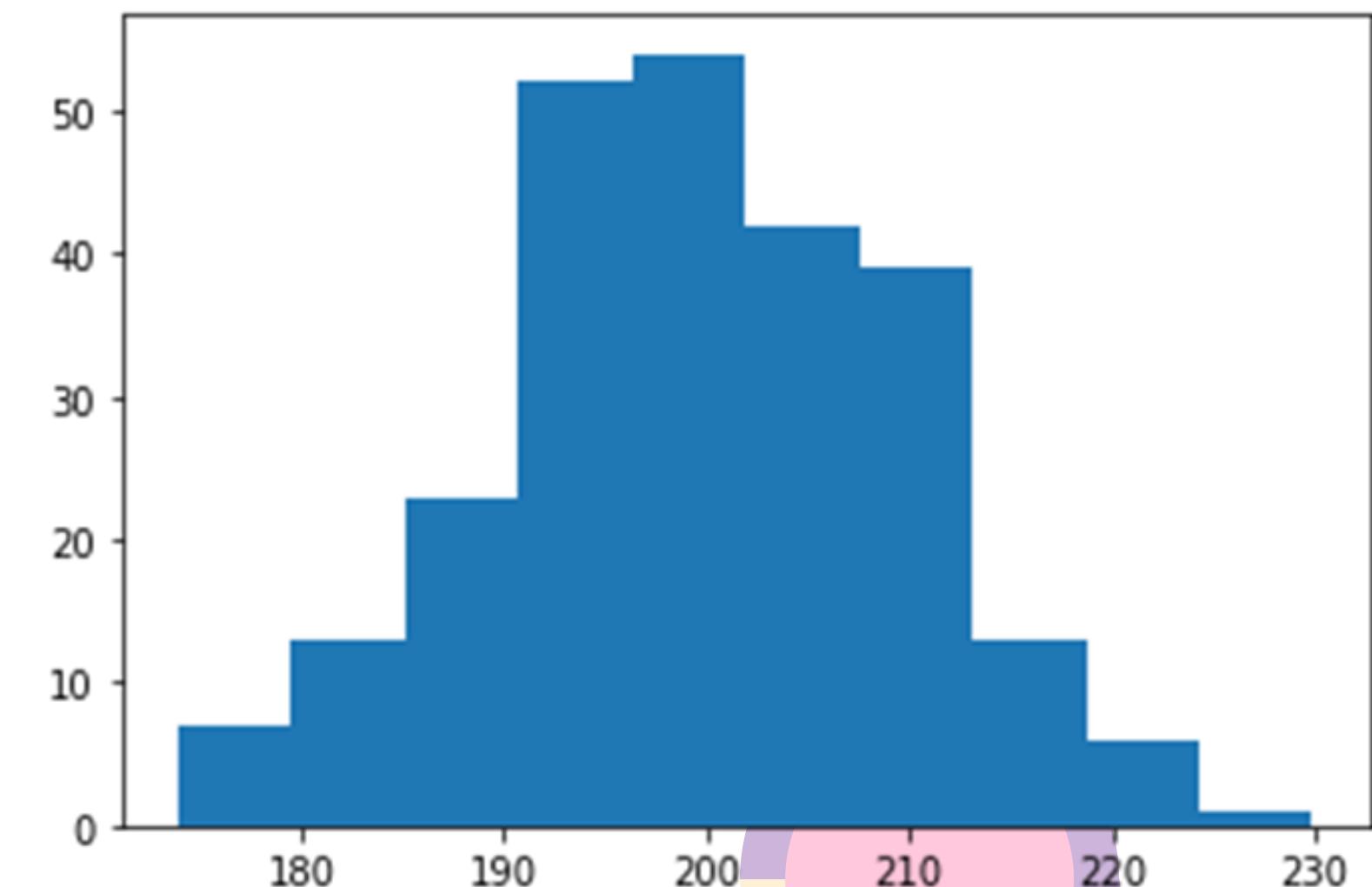


Plotting in Matplotlib

Histogram adalah grafik yang berisi ringkasan dari sebaran (dispersi atau variasi) suatu data



```
np.random.seed(192)  
x = np.random.normal(200, 10, 250)  
  
plt.hist(x)  
plt.show()
```



Plotting in Matplotlib

Scatter Plot digunakan untuk melihat relasi antara 2 variabel.

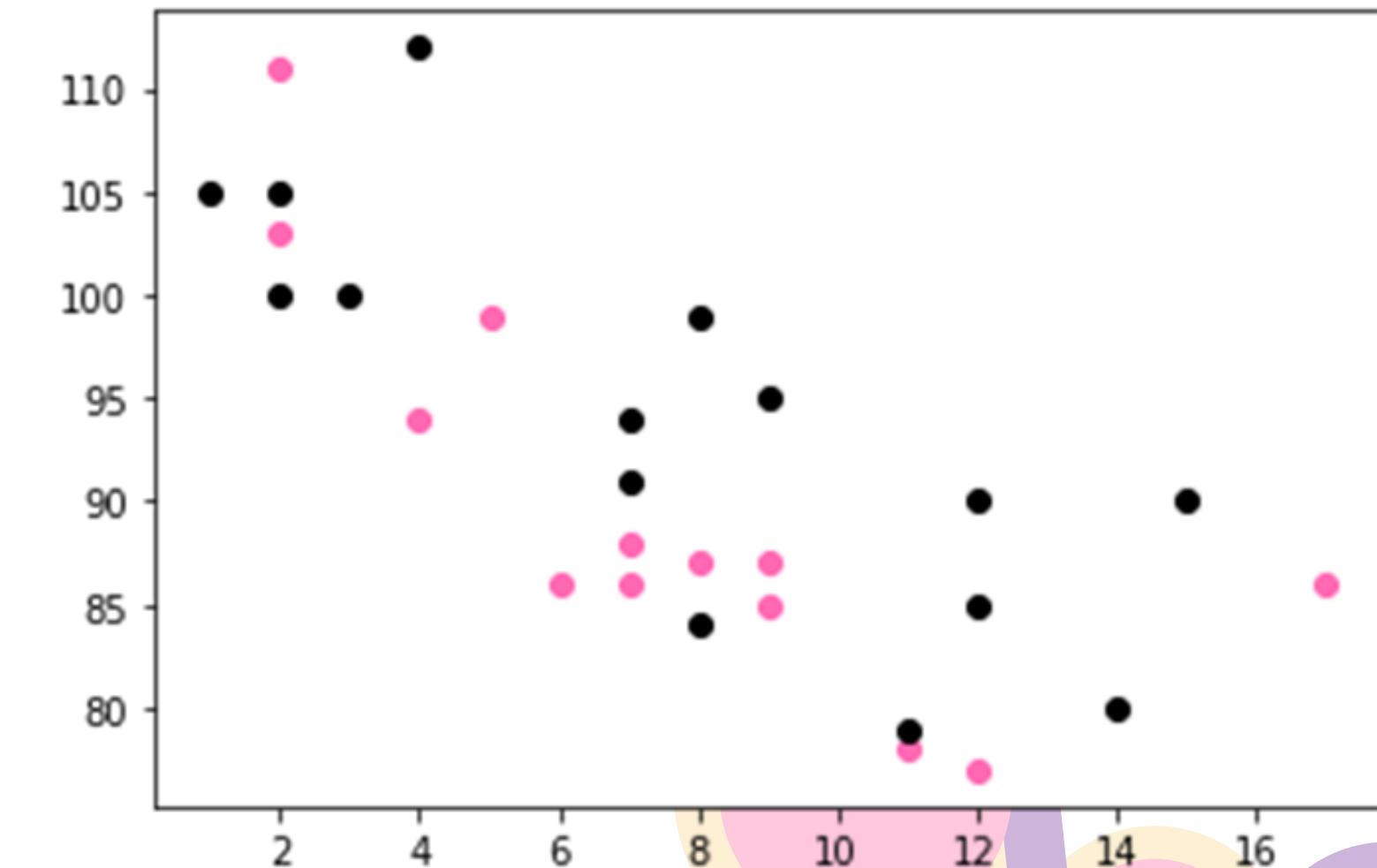
Scatter Plot juga dapat digunakan untuk melihat trend jumlah suatu data lebih besar di variabel mana.

```
# Colored Scatterplot

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
plt.scatter(x, y, color = 'hotpink')

x = np.array([2,2,8,1,15,8,12,9,7,3,11,4,7,14,12])
y = np.array([100,105,84,105,90,99,90,95,94,100,79,112,91,80,85])
plt.scatter(x, y, color = '#000000')

plt.show()
```

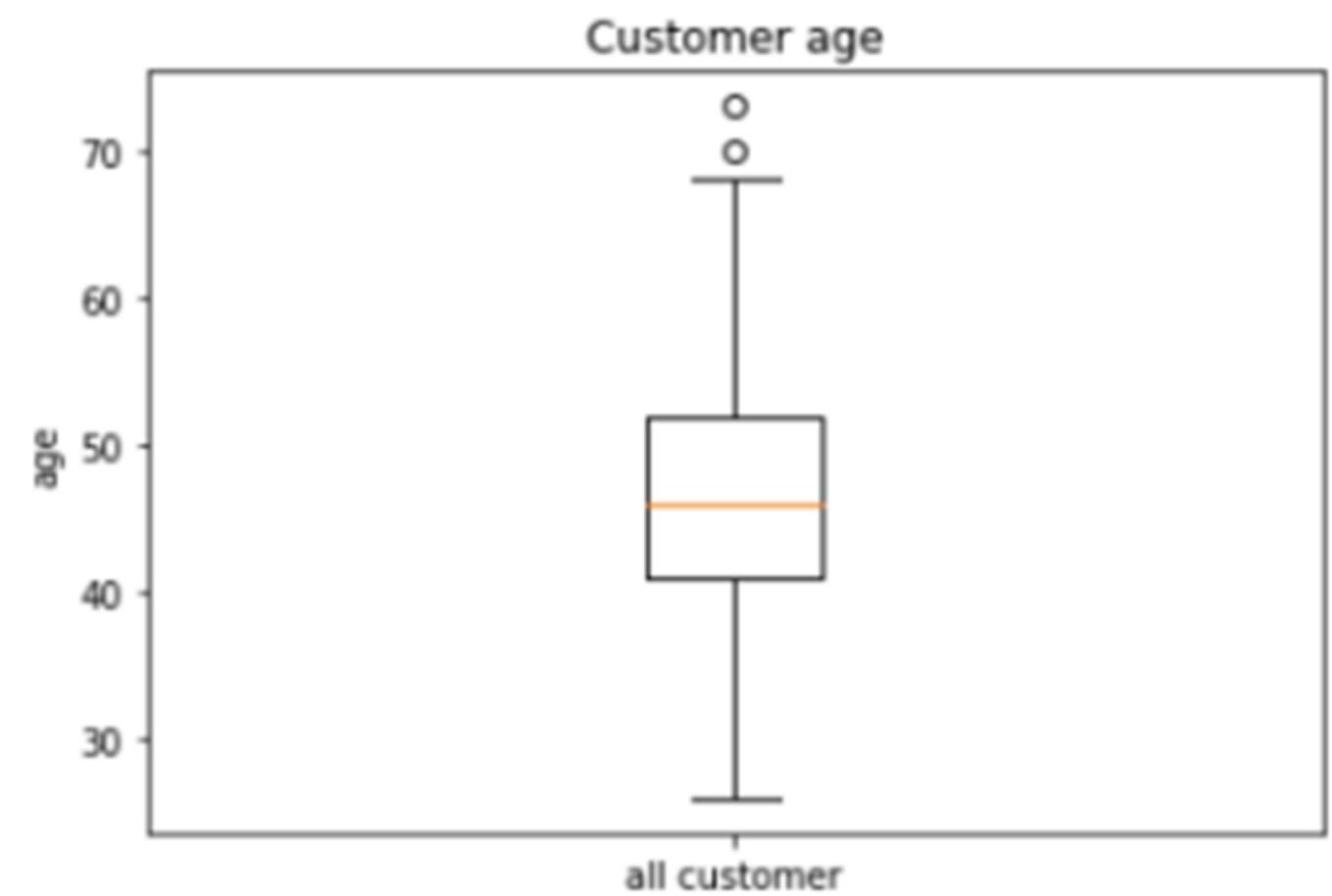


Plotting in Matplotlib

Boxplots dapat membantu kita dalam memahami karakteristik dari distribusi data. Selain untuk melihat derajat penyebaran data (yang dapat dilihat dari tinggi/panjang boxplot) juga dapat digunakan untuk menilai kesimetrisan sebaran data



```
# versi 1. contoh menggunakan fungsi boxplot dari plt. single group.  
plt.boxplot(df_churning['Customer_Age'])  
  
plt.title('Customer age')  
plt.ylabel('age')  
plt.xticks([1],labels=['all customer'])  
plt.show()
```

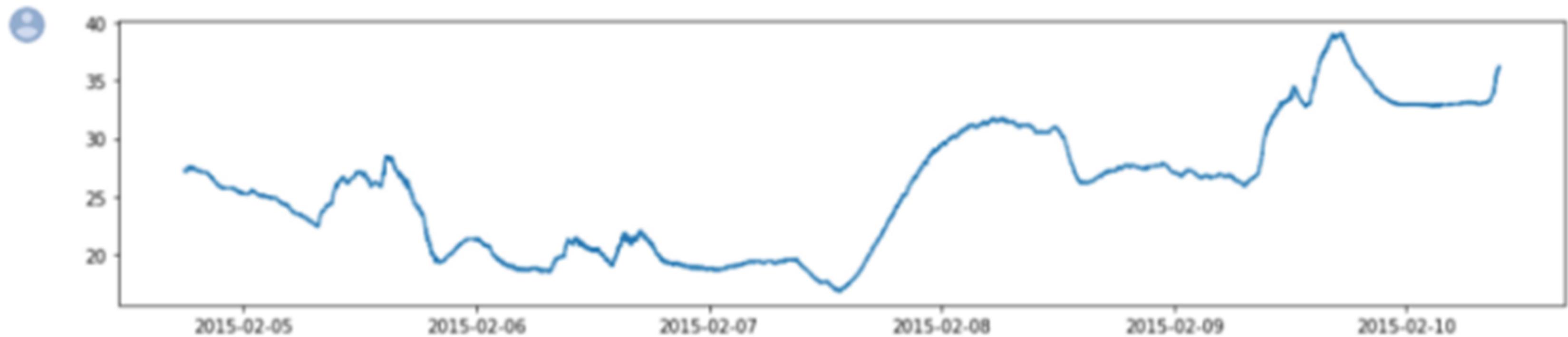


Plotting in Matplotlib

Line chart / Line Plot digunakan untuk menunjukkan tren dari waktu ke waktu.



```
plt.figure(figsize=(15, 3))
plt.plot(df_iot['date'], df_iot['Humidity'])
plt.show()
```

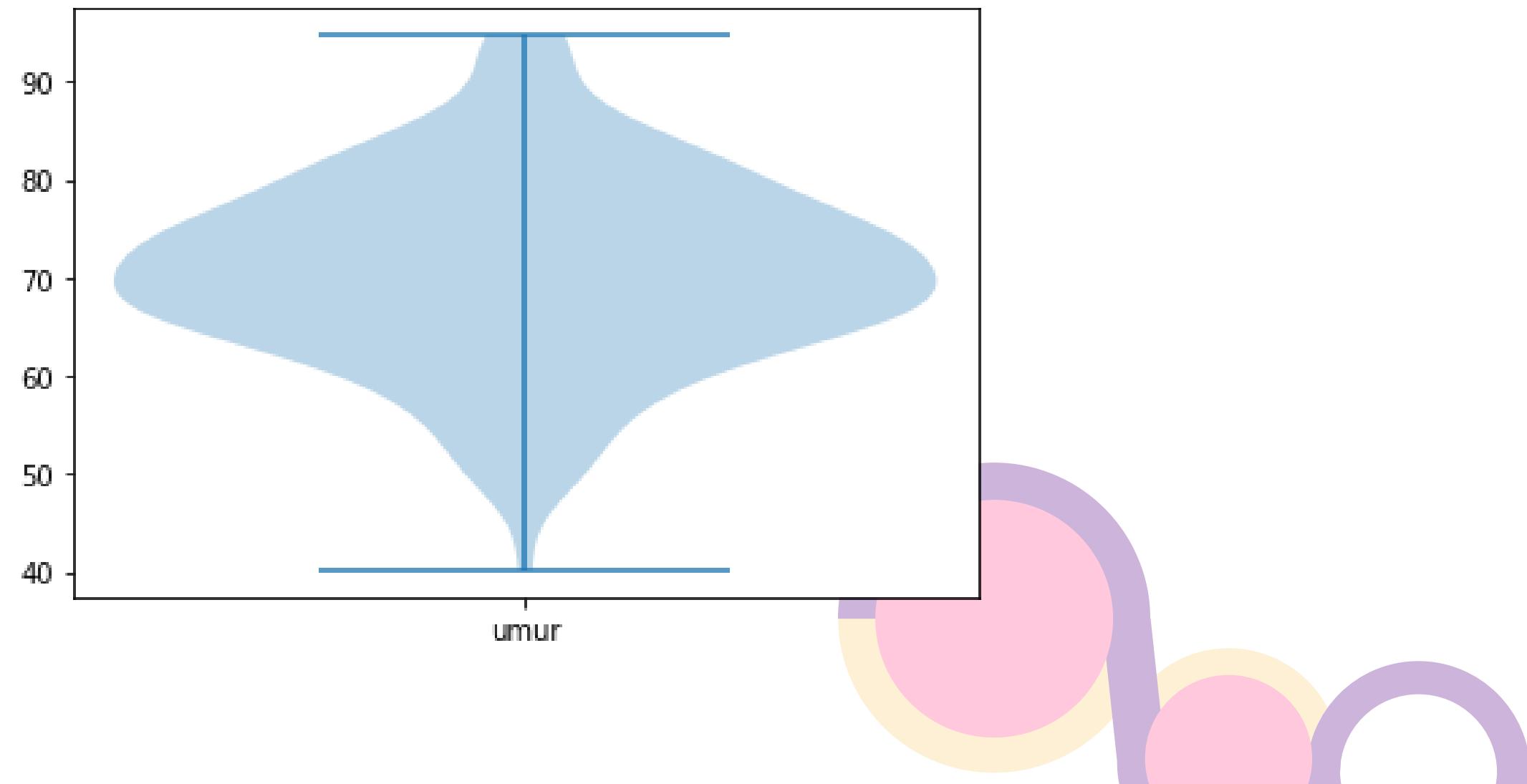


Plotting in Matplotlib

Violin plot merupakan penggabungan antara dua metode yaitu boxplot dan Estimasi Kepadatan Kernel (KDE). Tujuan dari violin plot yaitu untuk memudahkan pengguna menganalisis distribusi data yang kontinyu untuk setiap kategori.

```
np.random.seed(10)
data = np.random.normal(70,10,200)
plt.violinplot(data)

plt.xticks([1], labels=['umur'])
plt.show
```





Thank You
