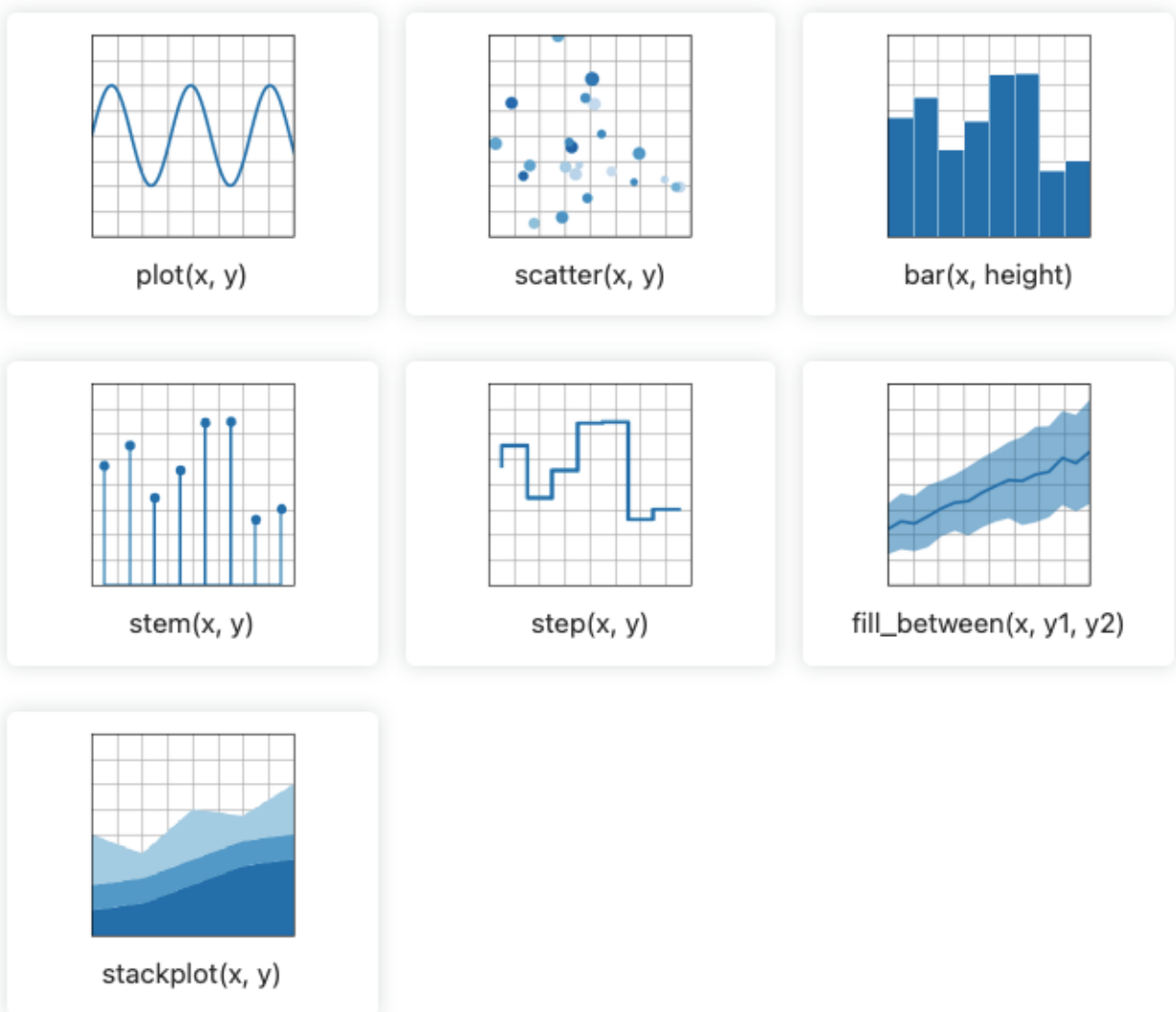


원본 링크 : <https://velog.io/@suminwooo/파이썬-Matplotlib-활용>

Matplotlib 활용

Matplotlib 란?

- 파이썬에서 데이터를 시각화해주는 라이브러리이다.
- 아래와 이미지와 같이 간단한 plot(바 차트, 파이차트, 히스토그램 등)을 그릴 수 있다.
- matplotlib의 경우 pandas나 numpy와 같이 처음부터 기초를 공부하기 보다는 필요할때마다 구글이나 [matplotlib 문서 참고 링크](#)를 찾아보면서 실습해보는 것이 좋다.

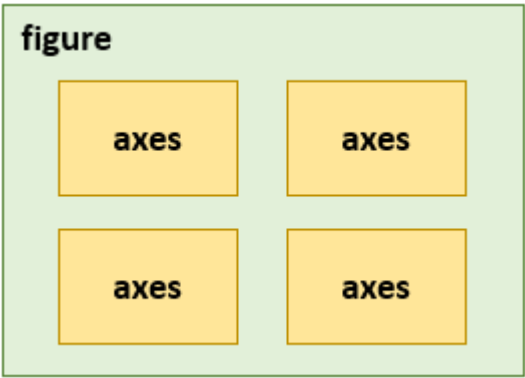


Matplotlib 사용 방법

- matplotlib 를 사용하기 위해서 다음과 같이 모듈을 import 합니다.
- 임포트를 할 때에는 plt 라는 축약된 이름을 관례적으로 많이 사용합니다.

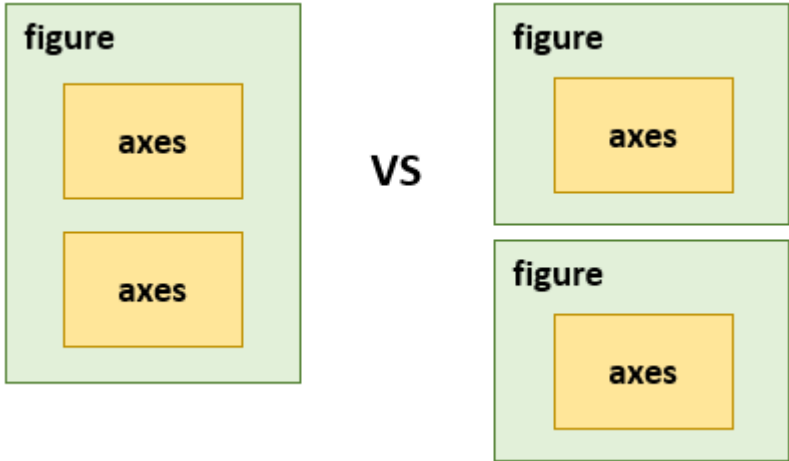
```
import matplotlib.pyplot as plt # 설치시 pip install matplotlib
```

여러개의 그래프를 표현할 경우



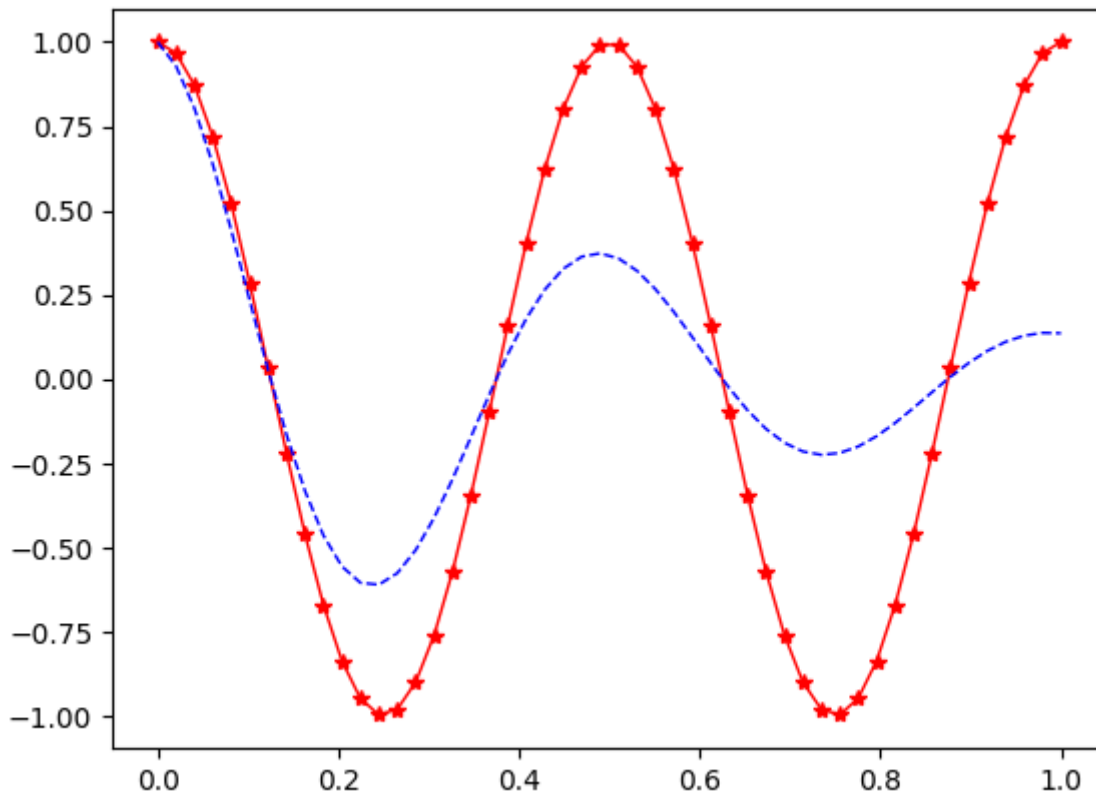
- figure(그림), axes(좌표축)

[경우1] figure을 1개 + 내부에 axes를 2개 만드는 경우 [경우2] figure을 2개 + 내부에 axes를 1개씩 만드는 경우



사용 방법

- Matplotlib을 활용해서 그림을 그리는 방법은 여러가지가 있습니다.
- 그 중에서 간단하게 쓸 수 있는 pyplot API와 객체지향 API가 있습니다.
- 아래의 이미지는 2가지 방식으로 표현할 수 있습니다.



1. Pyplot API를 이용

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

x = np.linspace(0,1,50)
y1 = np.cos(4*np.pi*x)
y2 = np.cos(4*np.pi*x)*np.exp(-2*x)

plt.plot(x,y1,'r-*',lw=1)
plt.plot(x,y2,'b--',lw=1)
plt.show()
```

2. 객체지향 API를 이용

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

x = np.linspace(0,1,50)
y1 = np.cos(4*np.pi*x)
y2 = np.cos(4*np.pi*x)*np.exp(-2*x)

# Figure 객체를 생성 후 직접 axes를 생성
fig = plt.figure()
ax = fig.subplots()
ax.plot(x,y1,'r-*',lw=1)
ax.plot(x,y2,'b--',lw=1)
plt.show()
```

기본 차트

- 아래에는 가장 많이 사용되는 내용을 정리해두었습니다.
- Matplotlib 공식 문서에 나온 예시입니다.
- 앞서나온 객체지향 API를 활용하였으나, Pyplot API를 활용해도 무관합니다.

plot(x,y)

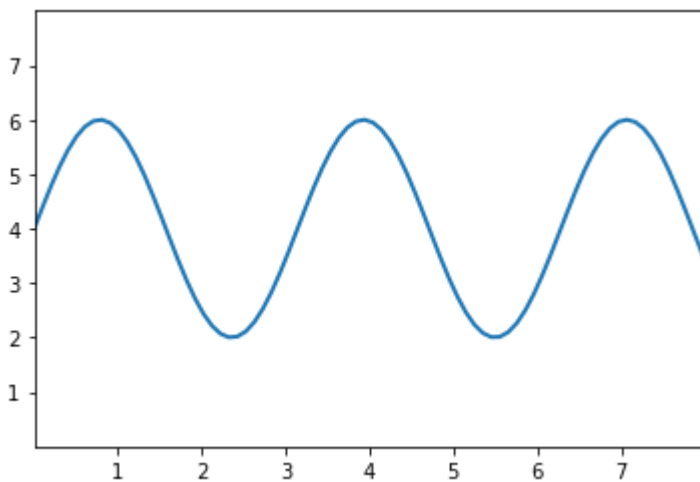
```
import matplotlib.pyplot as plt
import numpy as np

# 사용 데이터 설정
x = np.linspace(0, 10, 100)
y = 4 + 2 * np.sin(2 * x)

# 방법1
plt.plot(x, y)

# 방법2
fig, ax = plt.subplots()
ax.plot(x, y, linewidth=2.0)

plt.show()
```



scatter(x, y)

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

# 사용 데이터 설정
x = 4 + np.random.normal(0, 2, 24)
y = 4 + np.random.normal(0, 2, len(x))
# size and color:
```

```

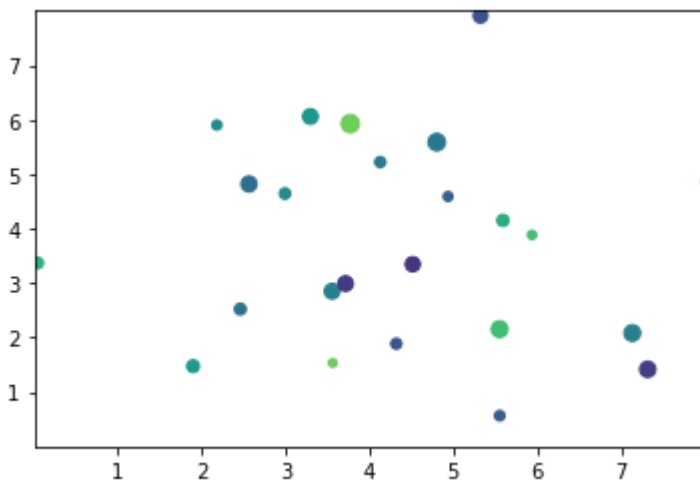
sizes = np.random.uniform(15, 80, len(x))
colors = np.random.uniform(15, 80, len(x))

# 방법1
plt.scatter(x, y, s=sizes, c=colors, vmin=0, vmax=100)

# 방법2
fig, ax = plt.subplots()
ax.scatter(x, y, s=sizes, c=colors, vmin=0, vmax=100)

plt.show()

```



bar(x, height)

```

import matplotlib.pyplot as plt
import numpy as np

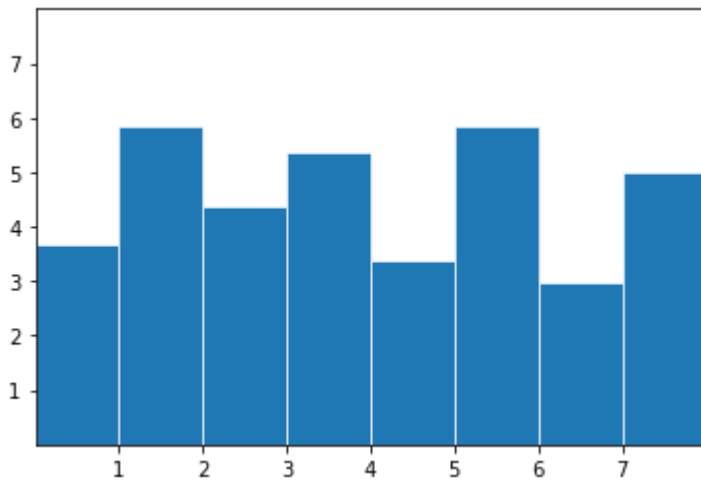
# 사용 데이터 설정
x = 0.5 + np.arange(8)
y = np.random.uniform(2, 7, len(x))

# 방법1
plt.bar(x, y, width=1, edgecolor="white", linewidth=0.7)

# 방법2
fig, ax = plt.subplots()
ax.bar(x, y, width=1, edgecolor="white", linewidth=0.7)

plt.show()

```



hist(x)

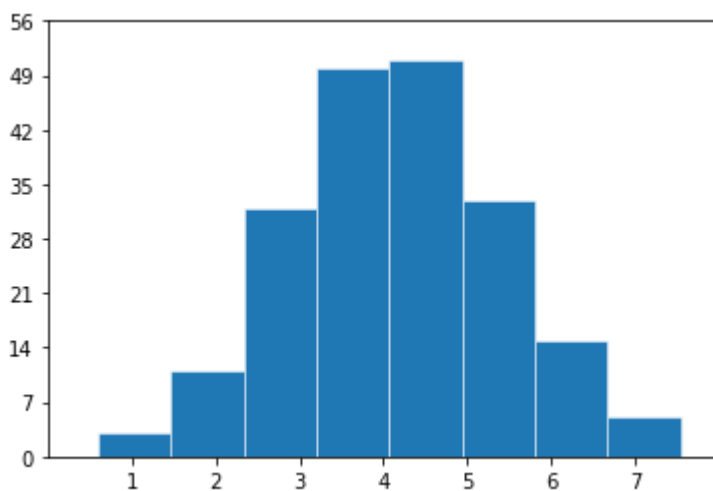
```
import matplotlib.pyplot as plt
import numpy as np

# 사용 데이터 설정
x = 4 + np.random.normal(0, 1.5, 200)

# 방법1
plt.hist(x, bins=8, linewidth=0.5, edgecolor="white")

# 방법2
fig, ax = plt.subplots()
ax.hist(x, bins=8, linewidth=0.5, edgecolor="white")

plt.show()
```



boxplot(X)

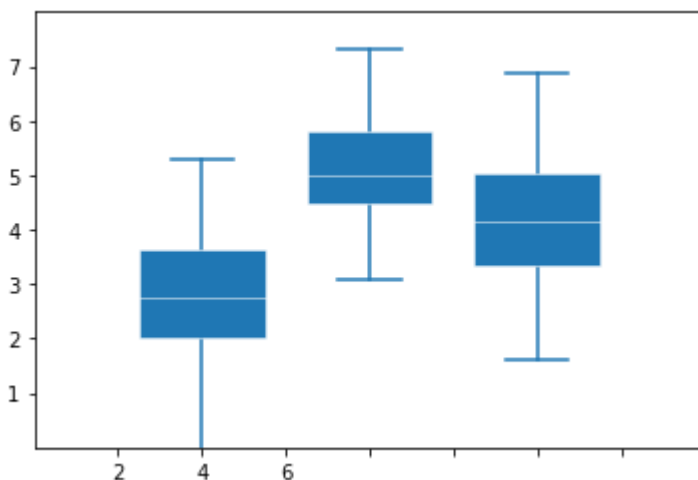
```
import matplotlib.pyplot as plt
import numpy as np
```

```
# 사용 데이터 설정
D = np.random.normal((3, 5, 4), (1.25, 1.00, 1.25), (100, 3))

# 방법1
plt.boxplot(D, positions=[2, 4, 6], widths=1.5, patch_artist=True,
            showmeans=False, showfliers=False,
            medianprops={"color": "white", "linewidth": 0.5},
            boxprops={"facecolor": "C0", "edgecolor": "white",
                    "linewidth": 0.5},
            whiskerprops={"color": "C0", "linewidth": 1.5},
            capprops={"color": "C0", "linewidth": 1.5})

# 방법2
fig, ax = plt.subplots()
VP = ax.boxplot(D, positions=[2, 4, 6], widths=1.5, patch_artist=True,
                showmeans=False, showfliers=False,
                medianprops={"color": "white", "linewidth": 0.5},
                boxprops={"facecolor": "C0", "edgecolor": "white",
                        "linewidth": 0.5},
                whiskerprops={"color": "C0", "linewidth": 1.5},
                capprops={"color": "C0", "linewidth": 1.5})

plt.show()
```



- 참고 :
 - https://matplotlib.org/stable/plot_types/index.html
 - <https://engineer-mole.tistory.com/218>