

데이터 시각화 (2024)

데이터과학부 정진명

(jmjung@suwon.ac.kr, 글로벌경상관 918호)

12 주차

Contents

- legend 위치선정
- 부그림

legend 위치선정

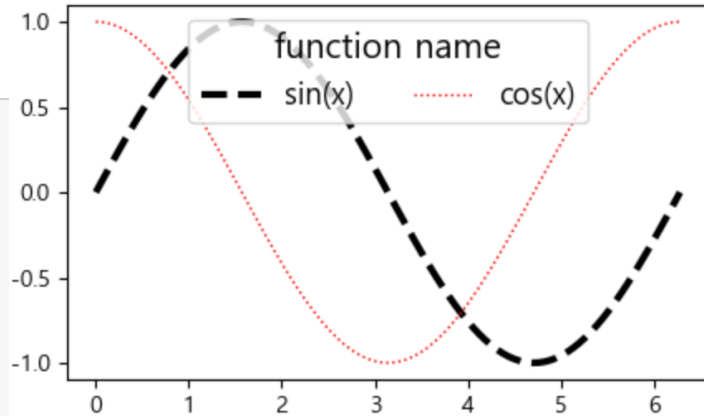
legend 다른 parameters

- title: 제목
- loc: legend 위치 (default: best)
- ncol: legend의 column의 개수
- fontsize
- title_fontsize
- bbox_to_anchor: legend 위치의 기준을 정함
- handles
- labels
- ...

```
fig=plt.figure(figsize=(5,3), dpi=100)
ax1=fig.subplots()
```

```
X=np.linspace(0,2*np.pi, 100)
Y1=np.sin(X)
Y2=np.cos(X)
```

```
_ =ax1.plot(X,Y1, c='k', lw=3, ls='--',label='sin(x)')
_ =ax1.plot(X,Y2, c='r', lw=1, ls=':.',label='cos(x)')
_ =ax1.legend(title='function name', loc='upper center', ncol=2, fontsize=13, title_fontsize=15)
```



| Location String | Location Code |
|-----------------|---------------|
| 'best' | 0 |
| 'upper right' | 1 |
| 'upper left' | 2 |
| 'lower left' | 3 |
| 'lower right' | 4 |
| 'right' | 5 |
| 'center left' | 6 |
| 'center right' | 7 |
| 'lower center' | 8 |
| 'upper center' | 9 |
| 'center' | 10 |

legend의 위치 선정: bbox_to_anchor

```
fig=plt.figure(figsize=(10,6), dpi=100)
axs=fig.subplots(2,2).flatten()
```

```
X=np.linspace(0,2*np.pi, 100)
Y1=np.sin(X)
```

```
_=axs[0].plot(X,Y1, c='k', lw=3, ls='--',label='sin(x)')
```

```
_=axs[0].legend(title='func name', loc='lower center', ncol=2, fontsize=12, title_fontsize=12, bbox_to_anchor=(0.5,1.0))
```

```
_=axs[1].plot(X,Y1, c='k', lw=3, ls='--',label='sin(x)')
```

```
_=axs[1].legend(title='func name', loc='upper left', ncol=2, fontsize=12, title_fontsize=12, bbox_to_anchor=(1.0,1.0))
```

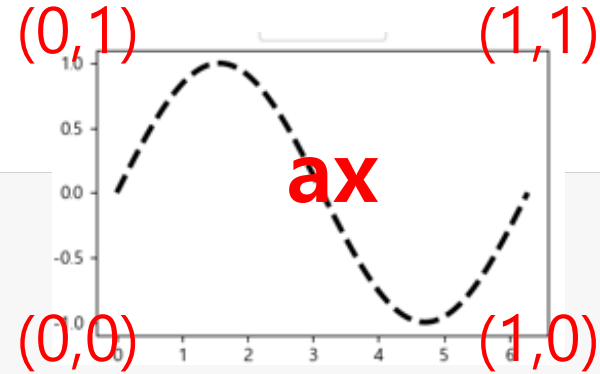
```
_=axs[2].plot(X,Y1, c='k', lw=3, ls='--',label='sin(x)')
```

```
_=axs[2].legend(title='func name', loc='upper left', ncol=2, fontsize=12, title_fontsize=12)
```

```
_=axs[3].plot(X,Y1, c='k', lw=3, ls='--',label='sin(x)')
```

```
_=axs[3].legend(title='func name', loc='upper right', ncol=2, fontsize=12, title_fontsize=12, bbox_to_anchor=(0,1.0))
```

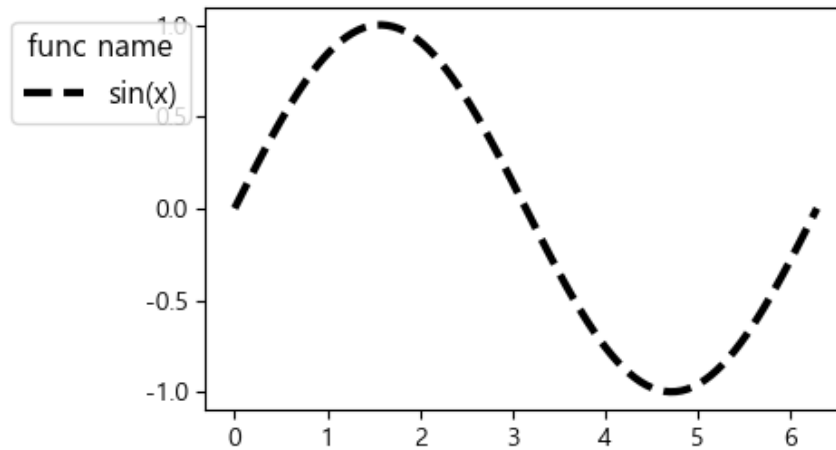
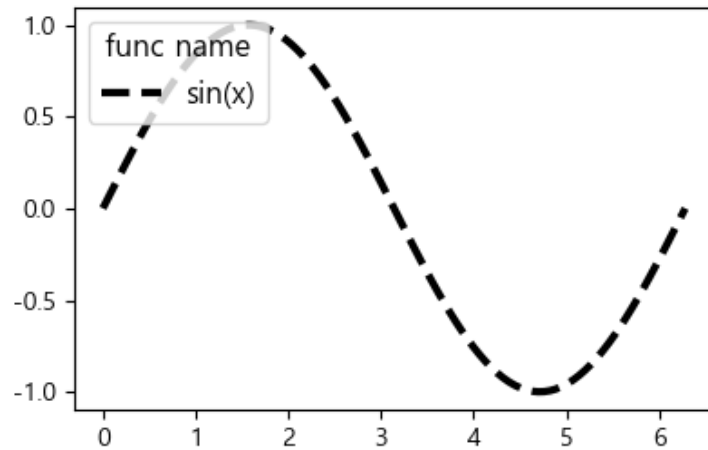
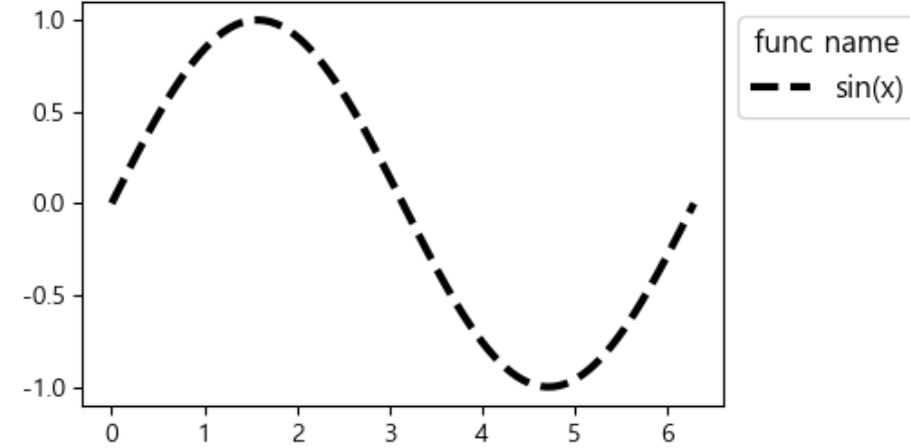
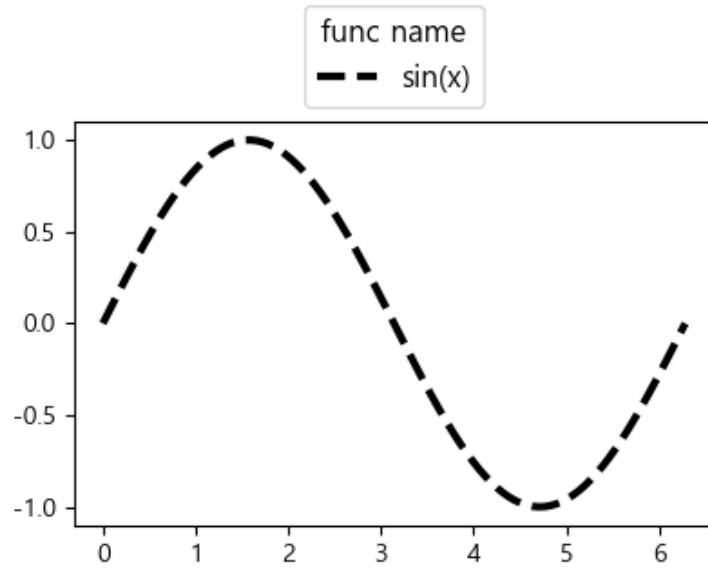
```
fig.tight_layout()
```



ax의 (0.5,1) 점에 legend box의 lower center를 맞춘다.

- **bbox_to_anchor 사용 안할 시:** loc은 legend box의 ax상의 절대적 위치를 말한다.
- **bbox_to_anchor 사용 시:** loc은 text처럼 동작하며, legend box의 loc 부분을 bbox_to_anchor 에 위치 시킨다.

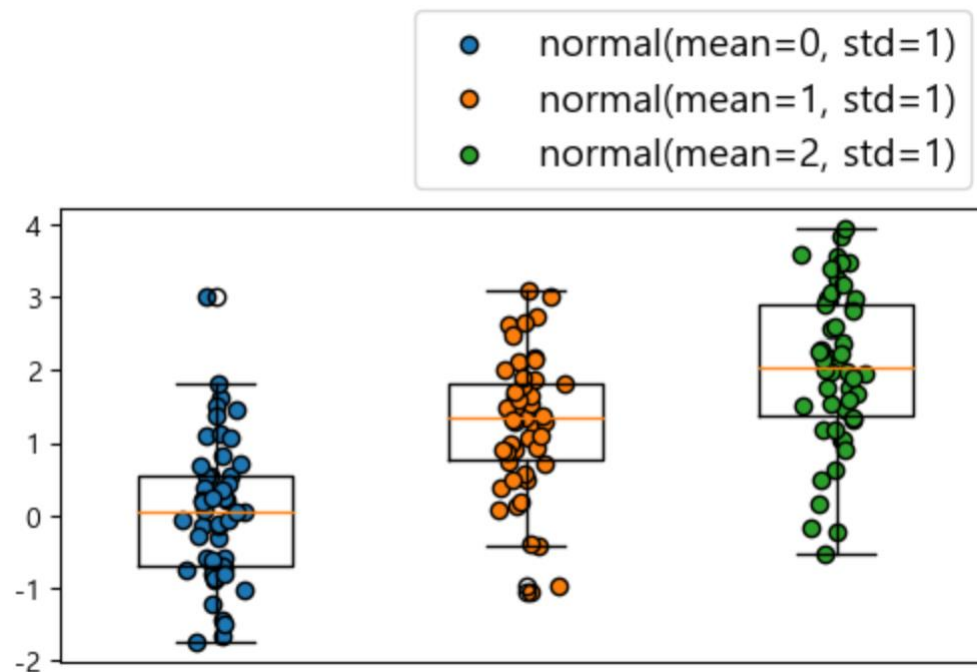
legend의 위치 선정: bbox_to_anchor



실습1

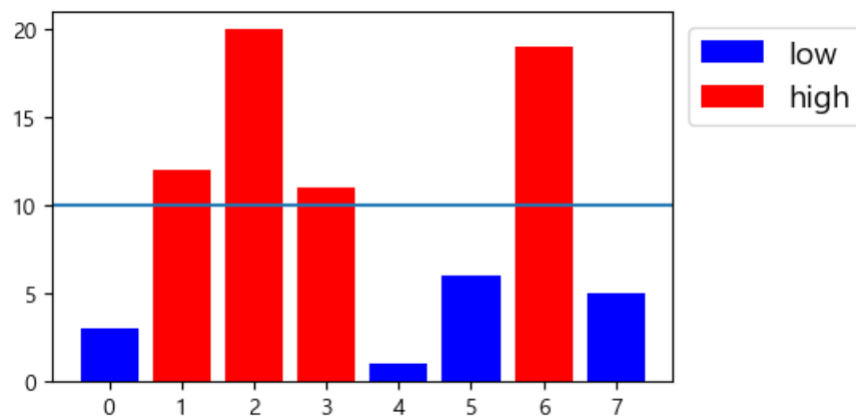
생성한 d1,d2,d3를 가지고 아래와 같은 그림을 시각화 하시오

1. boxplot
2. scatter with random variation
3. legend



실습2

```
# 랜덤하게 생성되는 8개의 정수로 series d1을 만들었다.  
# 값이 10이하이면 파란색 bar를, 10초과이면 빨간색 bar를 시각화하고  
# 각각 low, high로 legend를 추가하시오
```



부그림 넣기

부그림 넣기 (inset_axes)

```
fig=plt.figure(figsize=(6,4), dpi=100)
```

```
ax1=fig.subplots()
```

```
X=np.linspace(-15,15, 1000)
```

```
Y=np.sinc(X)
```

```
ax1.plot(X,Y)
```

```
ins_ax=ax1.inset_axes([0.7,0.7, 0.25,0.25])
```

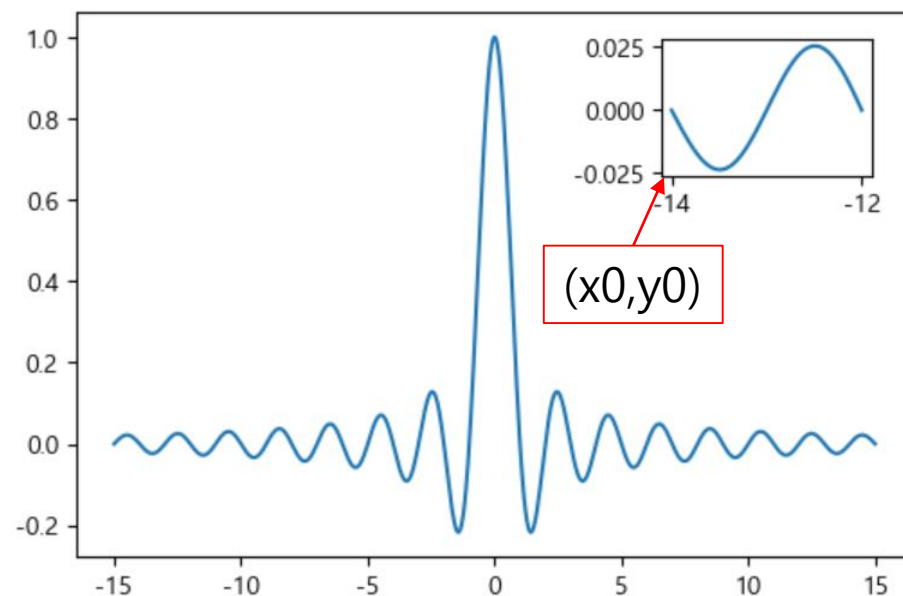
```
Xins=np.linspace(-14,-12,500)
```

```
Yins=np.sinc(Xins)
```

```
ins_ax.plot(Xins, Yins)
```

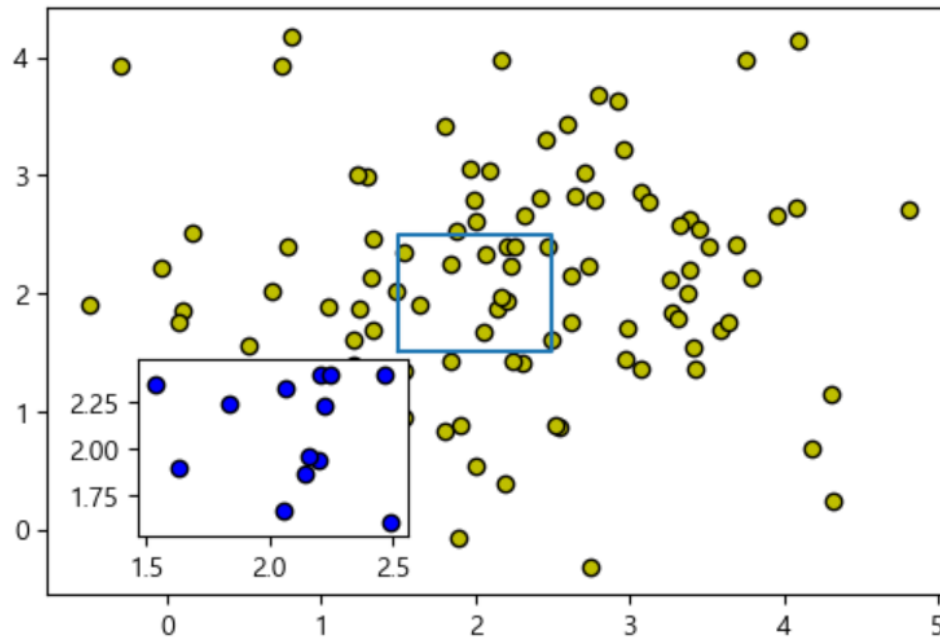
[x0, y0, width, height]

- 위 네 숫자 모두 axes 전체 사이즈를 1,1로 했을 때의 값
- x0, y0는 부그림의 왼쪽 아래의 상대적 좌표



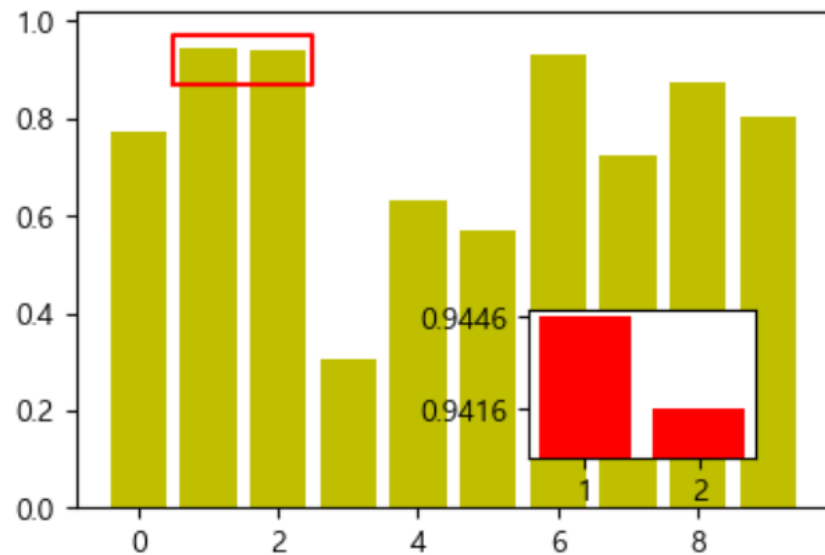
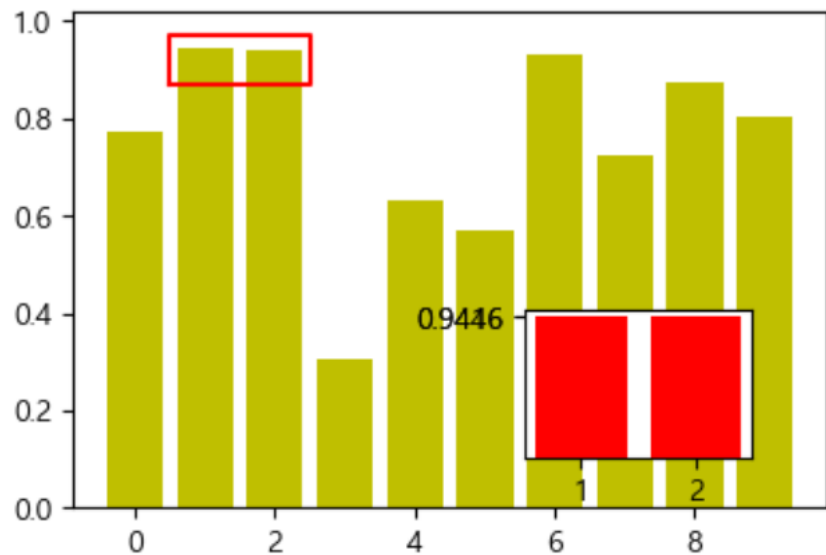
실습6

1. 평균이 2, 표준편차 1인 정규분포에서 shape (100,2) 샘플링 한 후
2. 첫번째 열을 x로 두번째 열을 y로 하여 **메인 그림에** 그림
3. 메인 그림의 $1.5 < x < 2.5$, $1.5 < y < 2.5$ 범위를 확대하여 **부그림에** 그림



실습7

- 두번째,세번째 bar 확대해서 부그림에 넣기
- ax1에 넣었더니 두 bar 가 잘 구분이 되질 않았다.
- 코드를 조금 수정하여 ax2에 두 bar가 잘 구분될수 있도록 하시오 (부그림의 y_tick 주의)



Q & A

Thank you