

13. 데이터 시각화(matplotlib & seaborn) #src: 8_Python/ch13_데이터시각화 참고**1) 시각화 개요;** 원본 데이터 또는 분석된 결과 데이터를 그래프로 표현

- 시각화 라이브러리: **matplotlib**, **seaborn**, **folum(지도 시각화)**, **plotnine**, **poly.ly**, **pycharts** 등

(1) **matplotlib**: 파이썬에서 가장 많이 사용하는 시각화 라이브러리. #판다스에서도 사용 多

(2) **seaborn**: matplotlib을 기반으로 다양한 색, 테마, 차트 기능을 추가한 라이브러리

2) Matplotlib

#https://matplotlib.org/stable/api/pyplot_summary.html

(1) matplotlib으로 그래프 그리는 단계

① 데이터 준비 ② 그래프(공간) 생성 ③ 그래프 그리기(함수) ④ 커스터마이징 ⑤ 출력&저장

%matplotlib inline; 주피터 노트북을 실행한 브라우저에서 시각화

%config InlineBackend.figure_format = 'retina'; 그래프 해상도 높임 #retina, 'jpeg', 'svg', 'pdf' 등

(2) 그래프 객체 matplotlib.pyplot.figure(num=None, figsize=None, ...) #figsize=(너비, 높이) inch

① plt.figure(); 그래프를 그리기 위한 객체 #import matplotlib.pyplot as plt

(3) 그래프 영역 나누기; 도화지(figure)를 plt.subplots()로 분할해 각 부분에 그래프를 그림

① plt.subplot(nrow, ncols, index); 현재 figure객체에 서브플롯 추가 #, 없이 써도 동일 ex) (211)

② fig, axes = plt.subplots(nrow=1, ncols=1, sharex=False, sharey=False, ...); 서브플롯 집합 추가
axes[0,0].plot(x, y, fmt, ...) #nrow와 ncol의 기본값은 1, sharex, sharey는 축 공유 여부

(4) 다양한 그래프; 판다스 데이터프레임을 이용해서 그래프를 그릴 수 있음 #pyplot 함수들

df.plot(x=None, y=None, kind='line', figsize=None, title=None, grid=None, xlim, ylim, fmt, ...)

① kind: line, scatter, bar, barh(수평바), hist, box, density=kde, area, pie, hexbin 등

② xlim, ylim: list/tuple

③ figsize: tuple(인치 단위)

③ fmt = '[marker][line][color]' or '[color][marker][line]' #ex) 'o--g' or 'go--'

(5) 그래프 함수들

① scatter(); 산점도. 데이터를 이용해 점의 크기(s), 색(c), 투명도(alpha) 설정 가능

② bar(), barh(), axvline(), axhline(); 수직 막대그래프, 수평 막대그래프, 수평선, 수직선

③ hist(), boxplot(), violinplot(); 히스토그램(값도 반환), 박스 플롯, 바이올린 플롯

(6) 그래프 커스터마이징

① 선 스타일(linestyle='-', 'dotted', '-' 등), 선 너비(linewidth), 선 색깔(color) 등 설정 가능

② text(), annotate(); 그래프에 text삽입, 강조할 값 표시(가리키기 등) 가능 #annotate

- ③ 그래프 제목과 축 제목: `suptitle("Main Title")`, `set_title("Title1")`
- ④ 축(Axis), 눈금(Tick): `xlim()`; 범위, `xlabel()`; 레이블, `xticks()`; 눈금, `xticklabels()`; 눈금 레이블
`set(ylim=[], yticks=[], yticklabels=[], ylabel=[], title=' ');` 한번에 설정 `#set_ylim()`; 각각 설정
- ⑤ `legend()`; `plot()` 함수가 label 속성을 가질 경우 함수의 호출만으로 범례 표시
- ⑥ 수학기호; 정규표현식 이용 [#https://matplotlib.org/2.0.2/users/mathtext.html](https://matplotlib.org/2.0.2/users/mathtext.html) 참고

(7) 그래프 저장: `savefig('파일명', ...)` `#plt.savefig("파일명.png")`

3) Seaborn: matplotlib 기반의 고수준 그래프 라이브러리 [#https://seaborn.pydata.org/api.html](https://seaborn.pydata.org/api.html)

(1) seaborn으로 그래프 그리는 단계

- ① 데이터 준비 ② 미적 속성 설정 ③ 함수를 이용하여 그래프 그리기 ④ 그래프 출력&저장

(2) 미적 속성 설정

```
seaborn.set(context='notebook', style='darkgrid', palette='deep',
            font='sans-serif', font_scale=1, color_codes=True, rc=None)
```

- ① context: `notebook`(기본값: 1배율), `paper(0.8)`, `talk(1.3)`, `poster(1.6)` `#set_context()`로도 설정 가능
- ② style: `darkgrid`, `whitegrid`, `dark`, `white`, `ticks` `#set_style()`
- ③ palette: 색깔 팔레트 지정 `deep`, `muted`, `bright`, `pastel`, `dark`, `colorblind` 등 `#set_palette()`

(3) seaborn으로 그래프 그리기

- ① 관계형 그래프(Relational plots): `scatterplot()`, `lineplot()`, `relplot()`; `lineplot`을 서브플롯으로 표현
- ② 범주형 그래프(Categorical plots): `catplot()`, `boxplot()`, `violinplot()`, `pointplot()`, `countplot()` 등
 - `stripplot()`, `swarmplot()`; 변수가 범주형인 산점도 그래프 `#swarmplot()`; 점들이 중첩X
 - `barplot()`; 포인트 추정치와 신뢰 구간을 막대그래프로 표시
 - `countplot()`; 막대를 사용하여 각 범주 구간의 관측 수를 표시
 - `pointplot()`; 산점도 그림문자(markers)를 이용하여 점 추정치 및 신뢰 구간 표시
 - `boxplot()`; 사분위 그래프 `boxenplot()`; 향상된 상자 플롯(큰 데이터셋에서 사용)
- ③ 쌍관계 그래프: `pairplot()`
- ④ 회귀 그래프(Regression plots): `lmpoint()`; 데이터를 이용, `regplot()`; 두 변수를 이용
- ⑤ 행렬 그래프(Matrix plots): `heatmap()`; 데이터를 색으로 인코딩 된 직사각형 행렬로 표시
`#annot=True`; 값 표시

(4) 다중 그래프를 위한 FacetGrid

- ① `FacetGrid(data, row, col, hue, col_warp, ...)`; 조건부 관계를 그리기 위한 다중 플롯 그리드
- ② `FacetGrid.map()`; 해당 영역에 그래프 함수를 이용해서 그래프 그리기 위해 `map()` 이용