데이터 시각화 (2024)

데이터과학부 정진명
(jmjung@suwon.ac.kr, 글로벌경상관 918호)

1 주차

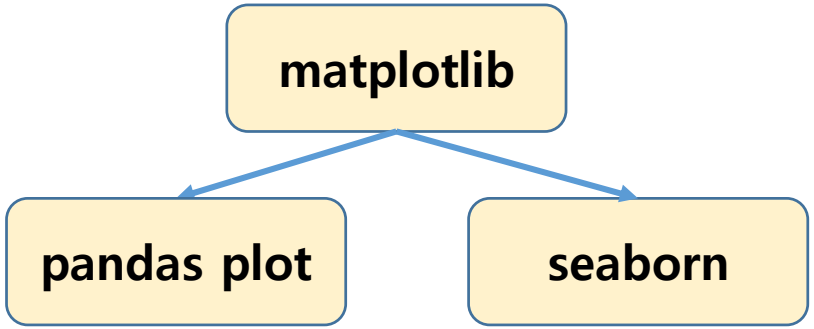
강의 개요

강의 목표

❖ **파이썬 패키지를** 활용하여, 데이터를 다양한 방법으로 **시각화 하는 능력** 개발

❖ 수업에서 다룰 시각화 패키지

- **Matplotlib (메인)** (low-level package)
- pandas plot, seaborn (high-level package)



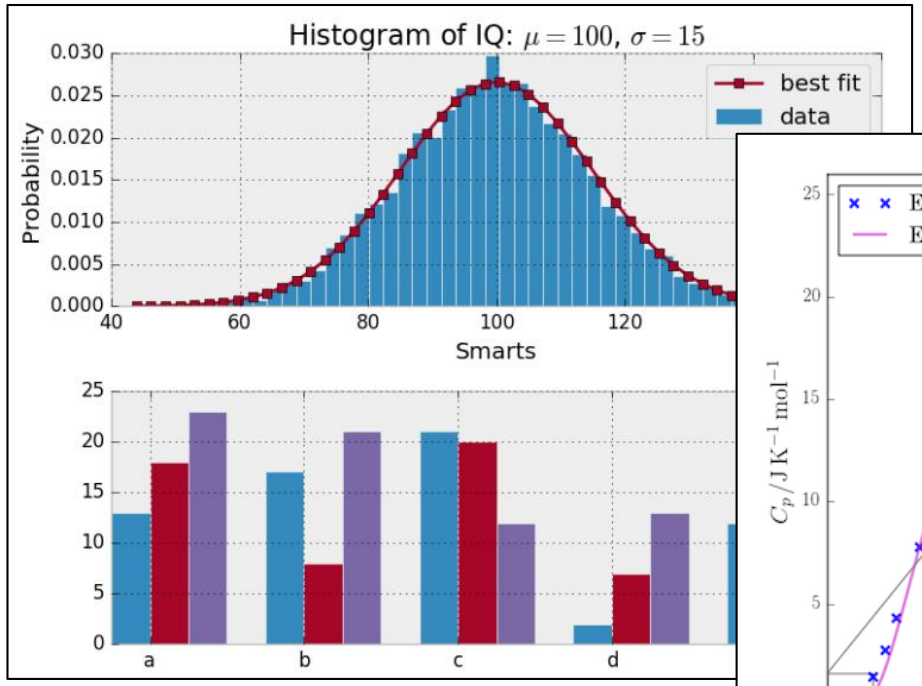
❖ 시각화하기 수행하기 위해서는 **데이터 핸들링** 능력이 필요

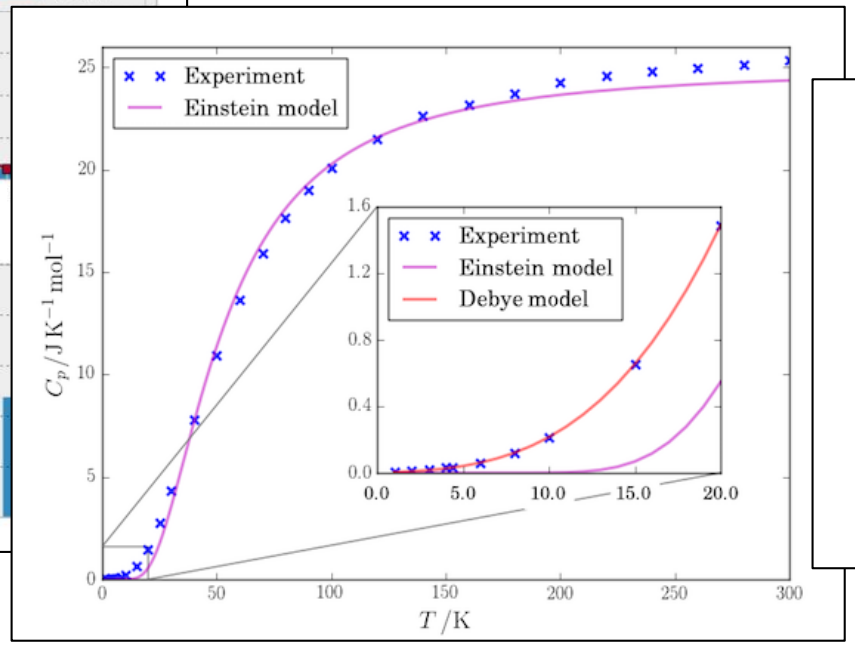
- 수업내용: **데이터 핸들링** (50%) + **시각화** (50%)
- 데이터 핸들링: 기초파이썬, numpy, pandas 활용

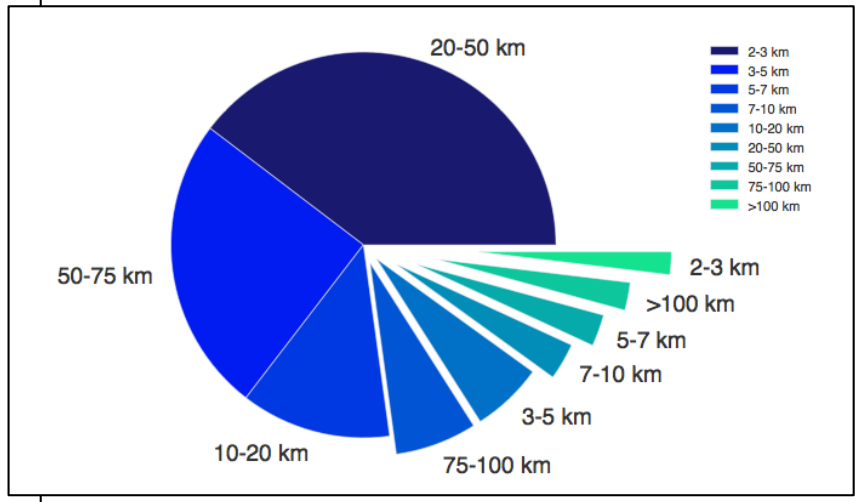
  (필요한 설명은 하지만, 자세한 설명 및 연습은 생략)

matplotlib

- **matplotlib**
  - 파이썬에서 자료를 차트(chart)나 플롯(plot)으로 시각화(visulaization)하는 라이브러리
  - 다양한 종류의 plot tool을 제공

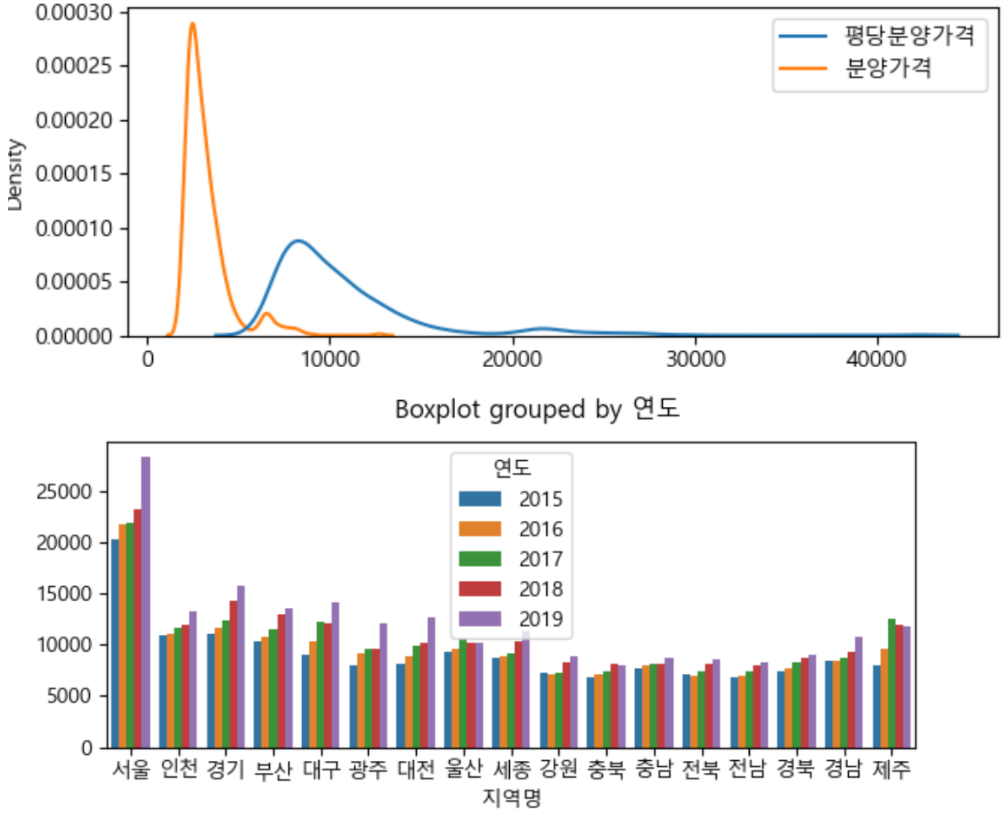


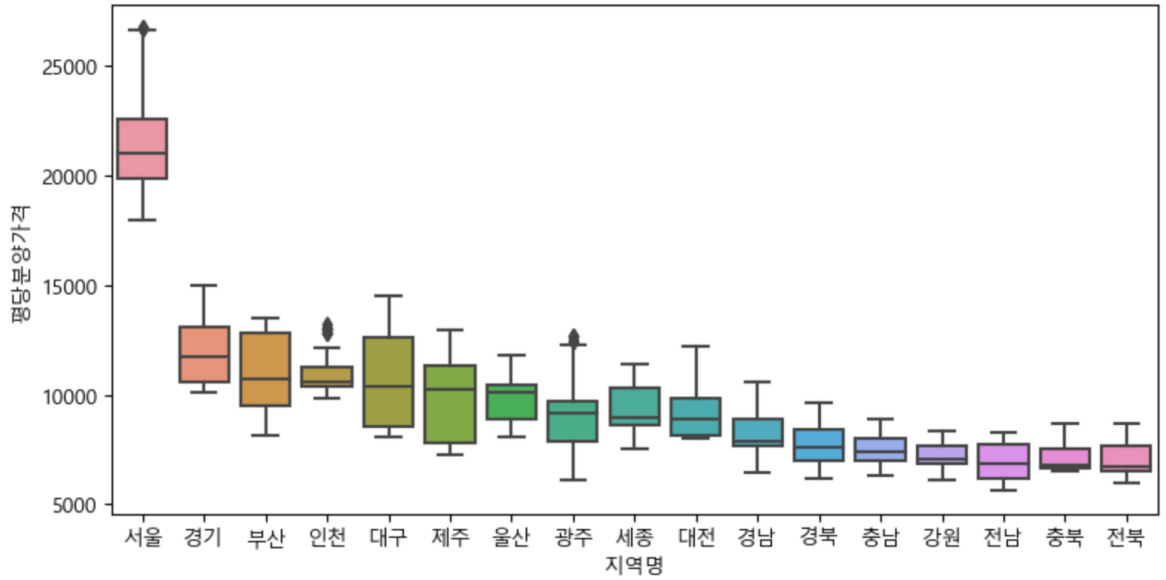




pandas plot, seaborn

- matplotlib 패키지를 모체로 하는 high level 패키지들로, 간단한 코드로 복잡한 시각화 가능
- 디테일한 조작이 필요할 때는 matplotlib의 함수들을 직접 사용해야 할 경우가 많음
- 몇몇 예제에 대하여 matplotlib 구현과 pandas plot, seaborn 구현 동시 수행하여 그 결과와 코드의 복잡도를 비교





시각화 예제

- ❖ 학년 별 남녀 각 평균 수면시간 시각화 (bar)
- ❖ 학년 별 수면시간 분포 시각화 (histogram, box)

학년	성별	수면시간
1	남	7
3	여	8
2	남	4
4	여	6
4	남	8
2	여	6
…	…	…

- ❖ 나이 대 (10대, 20대, …) 별로 키와 몸무게의 상관관계 시각화 (scatter)

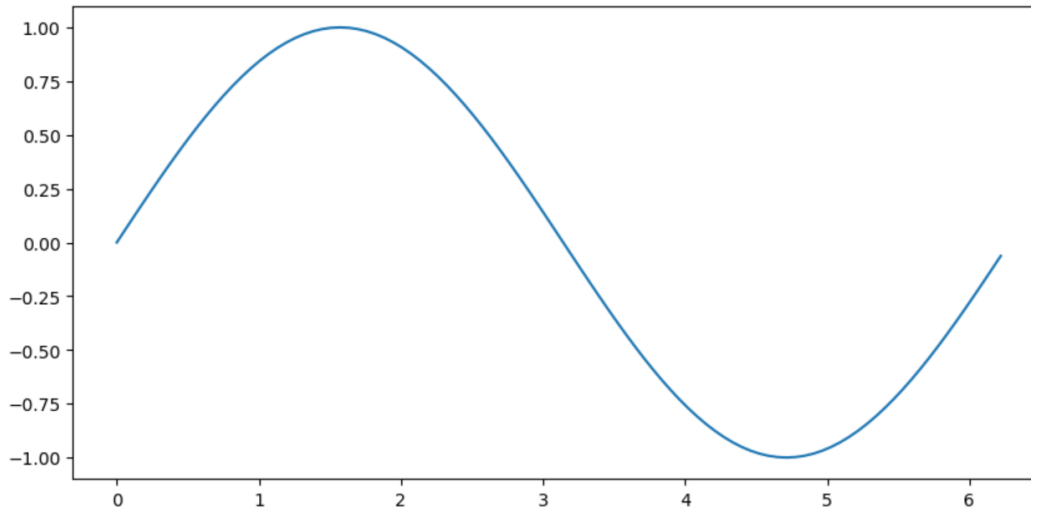
나이	키	몸무게
12	180	78
53	170	76
57	167	45
32	186	85
29	156	75
	…	…

- 필요한 데이터 핸들링은?
- 자동화 정도에 따라 프로그램 복잡도가 달라짐

수업 내용 예제

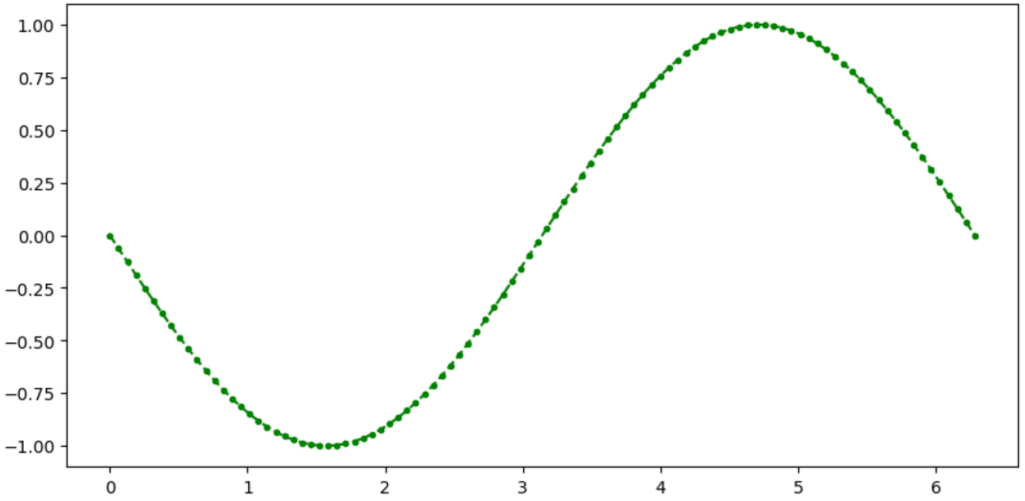
```
fig=plt.figure(figsize=(10,5), dpi=100)
ax=fig.add_subplot(1,1,1)

T=range(100)
X=[(2*math.pi*t)/len(T) for t in T]
Y=[math.sin(value) for value in X]

ax.plot(X,Y)
fig.show()
```

list comprehension




```
fig=plt.figure(figsize=(10,5), dpi=100)
ax=fig.add_subplot(1,1,1)

X=np.linspace(0,2*np.pi, 100)
Y=np.sin(X-np.pi)

ax.plot(X,Y,'.--g')
fig.show()
```

numpy package 함수

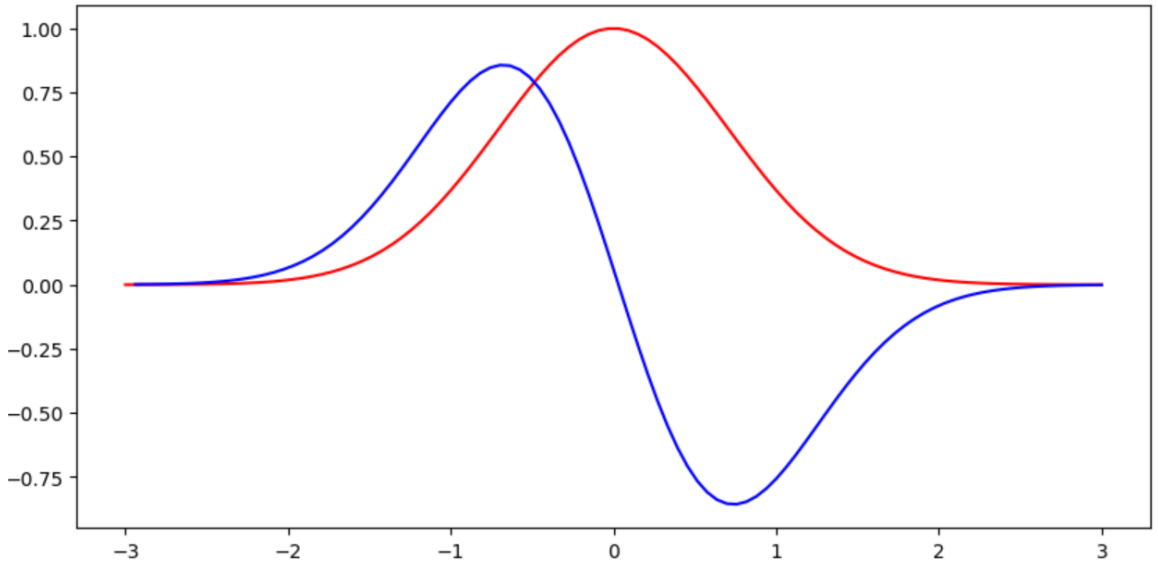


수업 내용 예제

```
def plot_slope(X,Y,ax):
    Xs=X[1:]-X[:-1]
    Ys=Y[1:]-Y[:-1]
    ax.plot(X[1:], Ys/Xs, 'b')

fig=plt.figure(figsize=(10,5), dpi=100)
ax=fig.add_subplot(1,1,1)

X=np.linspace(-3,3,100)
Y=np.exp(-X**2)

ax.plot(X,Y,'r')
plot_slope(X,Y,ax)
```

함수 정의 (기울기 구하는 함수)

함수 호출

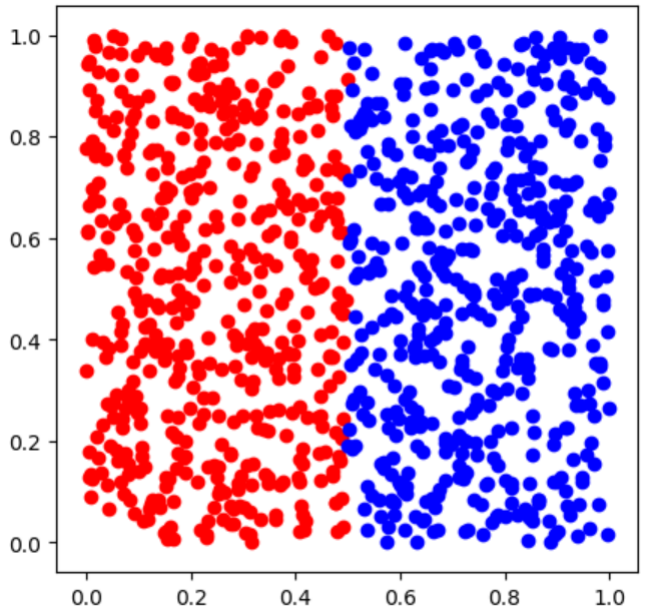


수업 내용 예제

```
fig=plt.figure(figsize=(5,5), dpi=100)
ax=fig.add_subplot(1,1,1)

data1=np.random.rand(500,2)
ax.scatter(data1[:,0]/2, data1[:,1],c='r')

data2=np.random.rand(500,2)
ax.scatter(0.5+data2[:,0]/2, data2[:,1],c='b')
fig.show()
```

numpy random 함수

numpy broadcasting

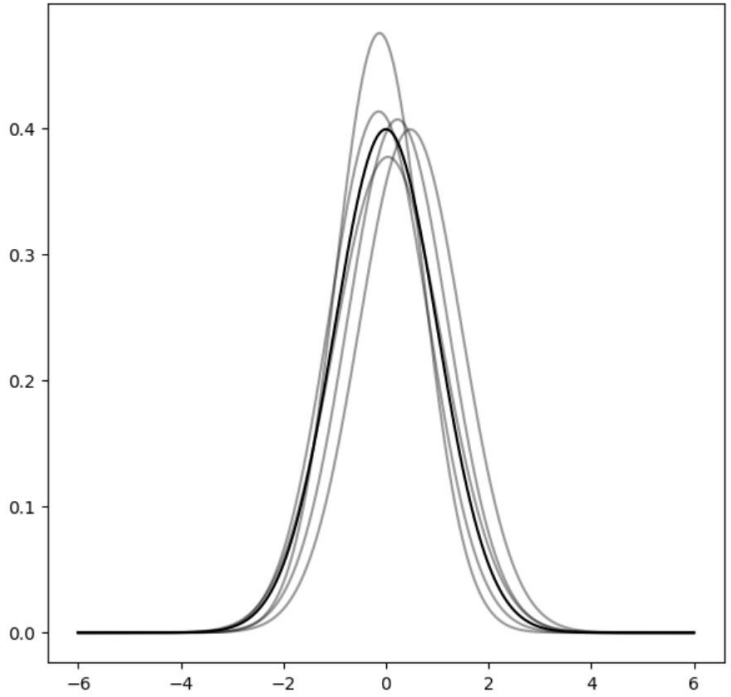


수업 내용 예제

```
def pdf(X, mu, sigma):
    a = 1/(sigma * np.sqrt(2*np.pi))
    b = -1/(2*(sigma**2))
    return a * np.exp(b * ((X - mu)** 2))

X = np.linspace(-6, 6, 1024)

for i in range(5):
    samples = np.random.standard_normal(50)
    mu, sigma = np.mean(samples), np.std(samples)
    ax.plot(X, pdf(X, mu, sigma), color = 'k', alpha=0.4)

ax.plot(X, pdf(X, 0., 1.), color = 'k')
```

함수 정의 (주어진 평균과 분산가지고 normal distribution 그리는 함수)

함수 호출

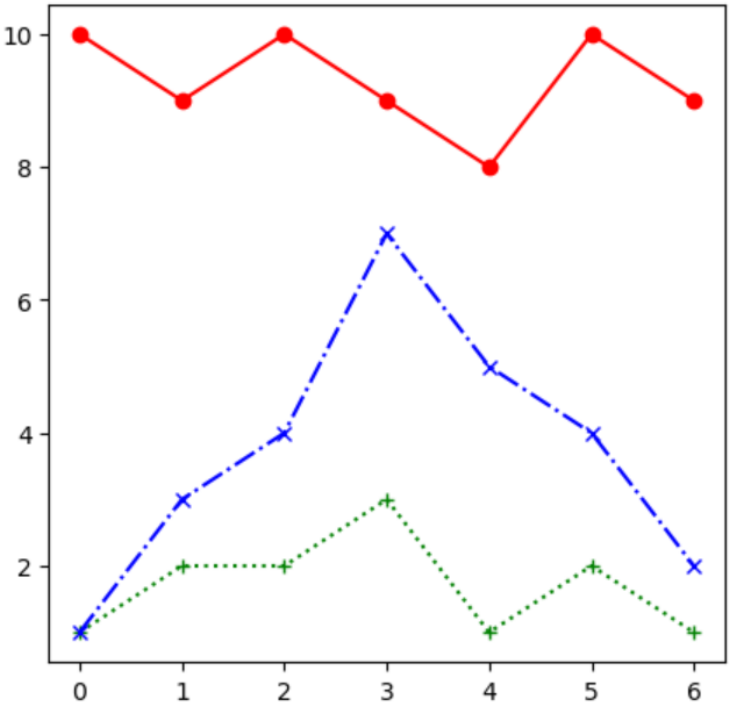


수업 내용 예제

```
## with numpy
fig=plt.figure(figsize=(5,5), dpi=100)
ax=fig.add_subplot(1,1,1)

data=np.loadtxt('data_p41.txt')
mrk=['o','+','x']
lin=['-',':','-.']
col=['r','g','b']

ind=0
for column in data.T[1:]:
    ax.plot(data.T[0], column, mrk[ind]+lin[ind]+col[ind])
    ind+=1

fig.show()
```

numpy 파일 읽기

for 문을 통한 표식 제어



수업 내용 예제

- PCA (principal component analysis)

	sepal length	sepal width	petal length	petal width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

Standardization →

	sepal length	sepal width	petal length	petal width
0	-0.900681	1.032057	-1.341272	-1.312977
1	-1.143017	-0.124958	-1.341272	-1.312977
2	-1.385353	0.337848	-1.398138	-1.312977
3	-1.506521	0.106445	-1.284407	-1.312977
4	-1.021849	1.263460	-1.341272	-1.312977

PCA (2 components) →

	principal component 1	princial component 2
0	-2.264542	0.505704
1	-2.086426	-0.655405
2	-2.367950	-0.318477
3	-2.304197	-0.575368
4	-2.388777	0.674767

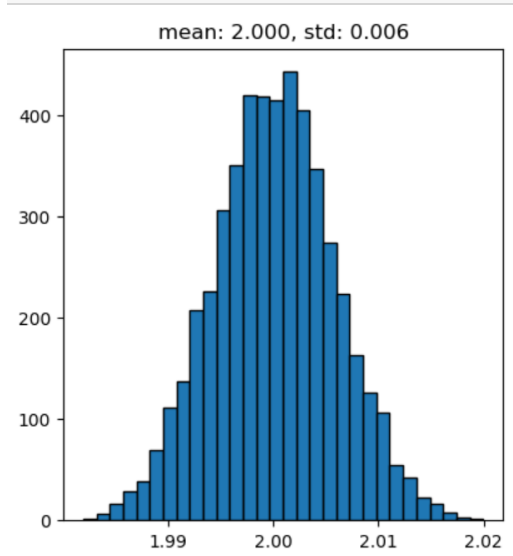
	target
0	Iris-setosa
1	Iris-setosa
2	Iris-setosa
3	Iris-setosa
4	Iris-setosa

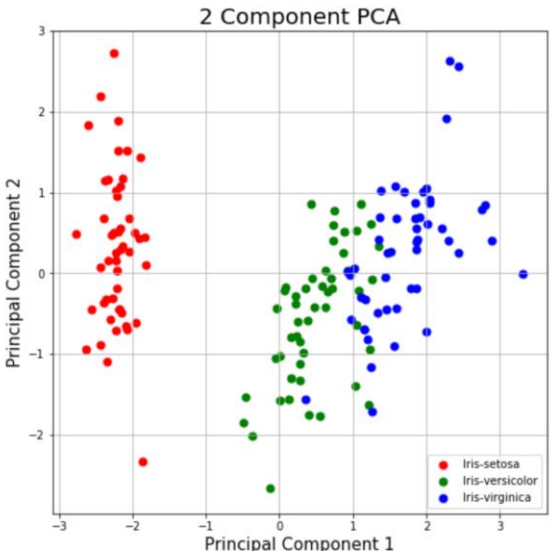
- 중심극한정리

matrix (two dimension)

```
data2=np.random.uniform(1,3,size=(10000,5000))
data2=data2.mean(axis=0)
plot_dist(data2, ax2)
```

numpy axis based operation



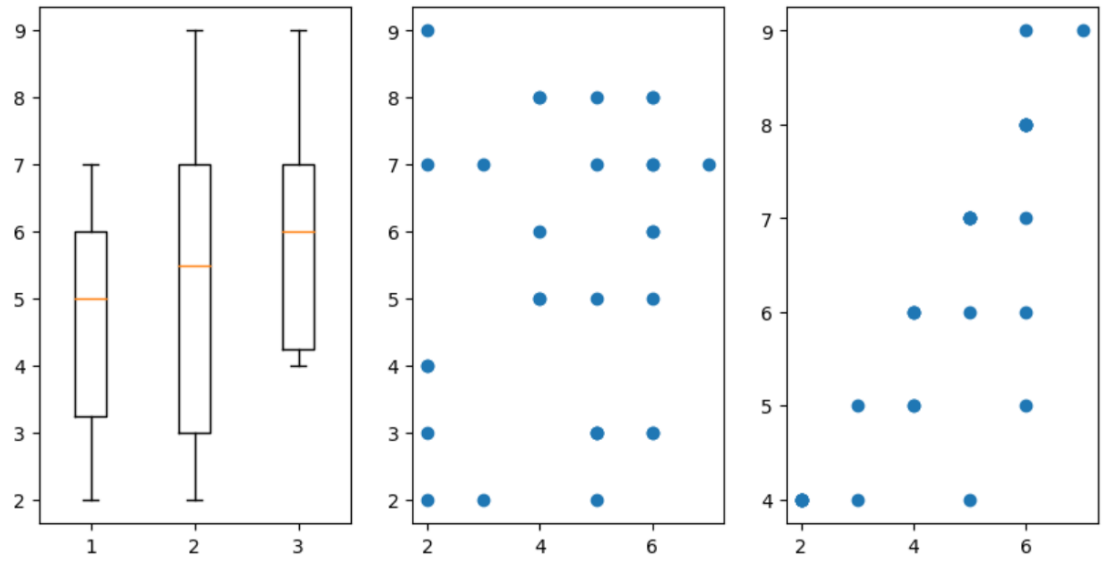


수업 내용 예제

pandas

```
data1=pd.read_table('data_p60.txt', sep='\t', engine='python')

_=ax1.boxplot(data1.as_matrix())
_=ax2.scatter(data1['groupA'], data1['groupB'])
_=ax3.scatter(data1['groupA'], data1['groupC'])
```

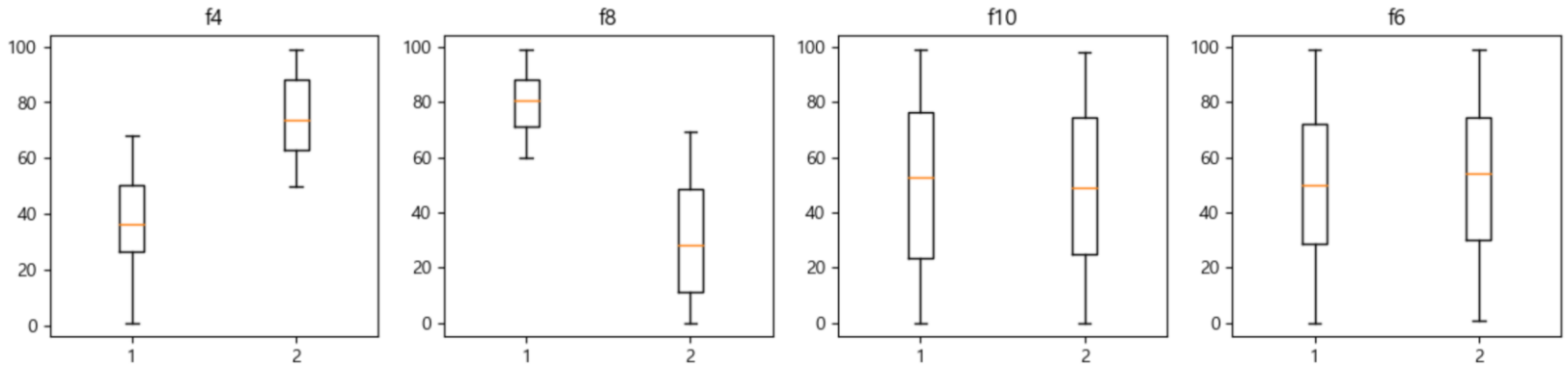
pandas column가져오기



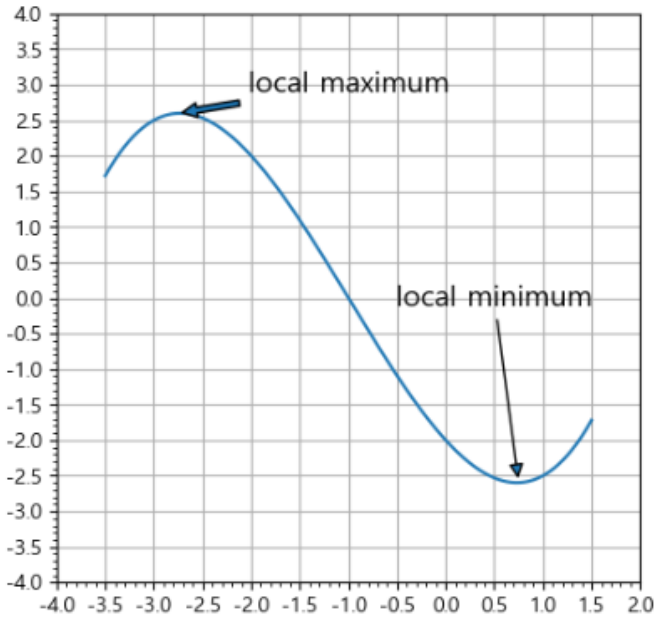
수업 내용 예제

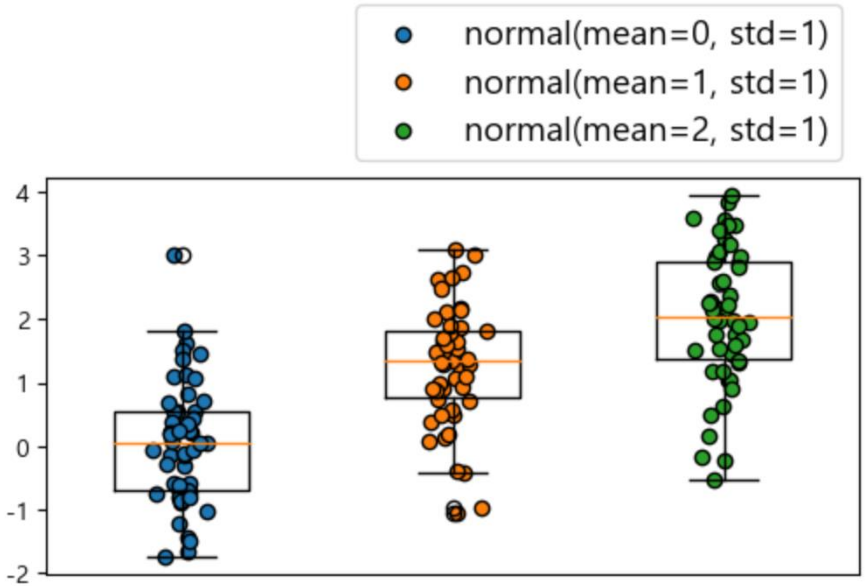
```
## data load
data = pd.read_table('data/dat_class.txt', sep='Wt')
data
data['class'].value_counts()

## data seperation
data0=data.loc[data['class']==0]
data0=data0.drop(columns=['class'])
data0
data1=data.loc[data['class']==1]
data1=data1.drop(columns=['class'])
data1

## get significant features
mean_diff=abs(data0.mean(axis=0)-data1.mean(axis=0))
mean_diff_sorted=mean_diff.sort_values()
g1,g2=mean_diff_sorted.index[-2:]
b1,b2=mean_diff_sorted.index[:2]
g1,g2

## draw boxplot
fig=plt.figure(figsize=(15,3), dpi=100)
axs=fig.subplots(1,4)

for ii,f in enumerate([g1,g2,b1,b2]):
    _=axs[ii].boxplot([data0[f], data1[f]])
    _=axs[ii].set_title(f)
```

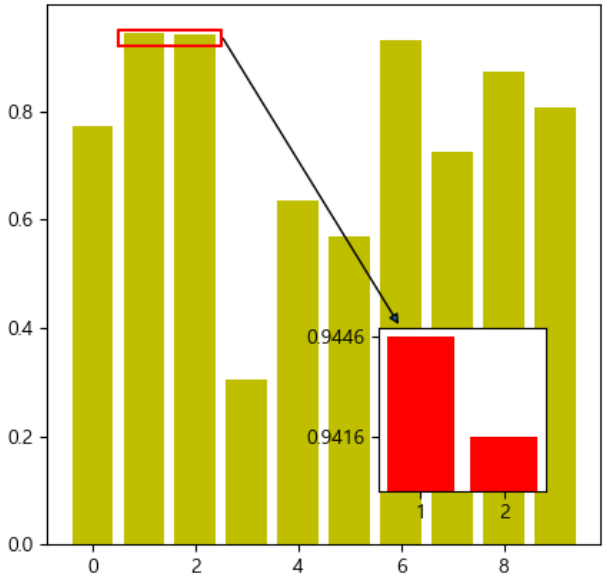
	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	class
0	20	23	89	35	63	50	53	71	16	8	0
1	16	89	86	97	49	93	61	6	43	77	1
2	35	58	20	3	93	48	6	61	75	33	0
3	62	35	39	64	67	36	89	4	42	64	1
4	69	68	31	50	72	51	95	51	43	41	1
...	...	...	...	...	...	...	...	...	...	...	...
195	96	32	16	16	78	7	19	95	13	85	0
196	2	78	84	97	3	47	10	16	37	35	1
197	45	7	99	85	83	72	38	20	25	77	1
198	45	28	91	69	80	70	1	38	50	52	1
199	33	80	23	54	20	48	10	22	15	41	1

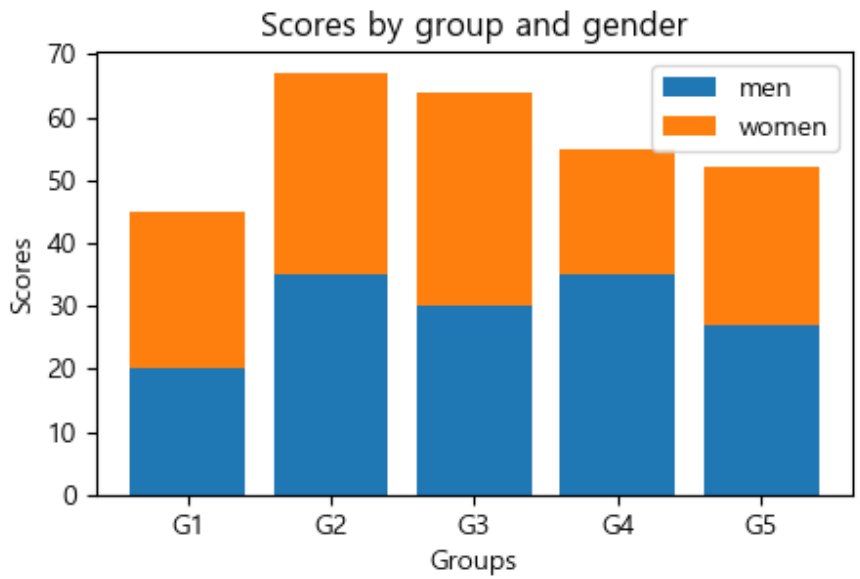


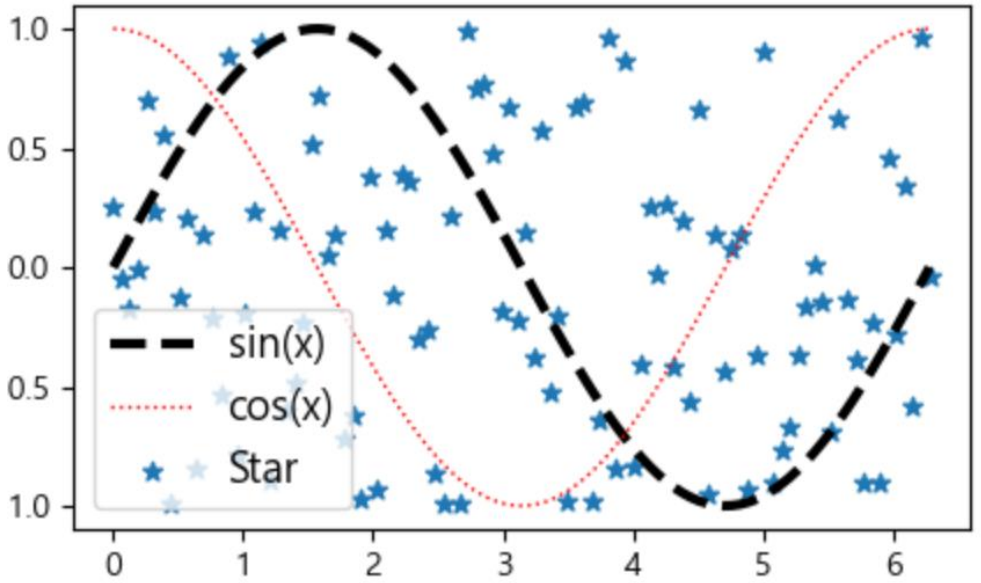
수업 내용 예제


local maximum
local minimum


normal(mean=0, std=1)
normal(mean=1, std=1)
normal(mean=2, std=1)


0.9446
0.9416


Scores by group and gender
men
women
Scores
Groups
G1
G2
G3
G4
G5


sin(x)
cos(x)
Star

강의 방식

- **출석**
  - 수업 시작 시 출석 확인 (보통 전자출결)
  - 출석 총점: 15점 (1회 결석 시 1점 감점, 3회 지각 시 1회 결석)
  - 출석이 제대로 체크 되지 않은 경우, 해당일에 연락 해야함 (해당 수업 일이 지나면 증빙자료 필요)

- **주요 공지사항:**
  - 캔버스 공지사항
  - 수업 오픈채팅방 (추후 캔버스에 오픈채팅방 주소 및 비번 공지)

- **강의자료 업로드 (캔버스)**
  - 업로드 강의 자료: **1)** PPT 자료 **2)** python code 및 데이터
  - 업로드 시간: 매 주 수업시간 전

- **과제 출제 (2~3번)**
  - 과제 제출 방법: 작성한 ipynb 파일을 **캔버스** 해당 '**과제**' 란에 제출
  - 기한: 과제 출제 후 1주일

- **강의시간**: 1시간 10분 강의 + 20분 휴식 + 1시간 10분 강의

평가 방법

출석: 15점
과제: 15점
기말시험: 35점 (10 ~ 12 문제)
중간시험: 35점 (10 ~ 12 문제)

- 중간, 기말고사:
  1) 과제, 이전 중간,기말고사와 유사한 형태
  2) 오픈북 (ppt, ipynb 파일을 포함한 모든 자료 사용 가능)
  3) 단, 인터넷 사용금지
  4) 부정행위 (chatgpt, 카카오톡, 인터넷검색 사용 등) 방지를 위한 시험 모니터 녹화 후 제출

- 성적 산정 후에, 개인적인 이유로 학점 변경 불가

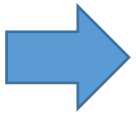
python interpreter 및 package 버전 확인하기

mission

1. jupyter notebook을 실행시킨후
2. test_code.ipynb를 열고
3. 첫번째 cell을 실행하시오

```
import sys
print('python', sys.version)

import numpy as np
print('numpy', np.__version__)

import pandas as pd
print('pandas', pd.__version__)

import matplotlib as mpl
print('matplotlib', mpl.__version__)

import matplotlib.pyplot as plt

import seaborn as sns
print('seaborn', sns.__version__)

# # 결과 확인을 용이하게 하기 위한 코드
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = 'all'

plt.rc("font", family="Malgun Gothic") # 한글표시 (window)
plt.rc("axes", unicode_minus=False) # x,y축 (-)부호 표시
```

현재 교수자 컴퓨터에 설치되어 있는 python interpreter와 그 외 package 버전 (동일할 필요는 없음)



```
python 3.11.7 | packaged by Anaconda, Inc. | (main, Dec 15 2023, 18:05:47) [MSC v.1916 64 bit (AMD64)]
numpy 1.26.4
pandas 2.1.4
matplotlib 3.8.0
seaborn 0.13.2
```

Q & A

Thank you