빅데이터 실습

9주차 1차시

데이터 시각화 - matplotlib 라이브러리 이해하기

데이터 시각화
matplotlib 라이브러리 이해하기

학습개요

1/ 데이터 시각화의 개념 및 중요성

2/ Matplotlib 라이브러리 소개

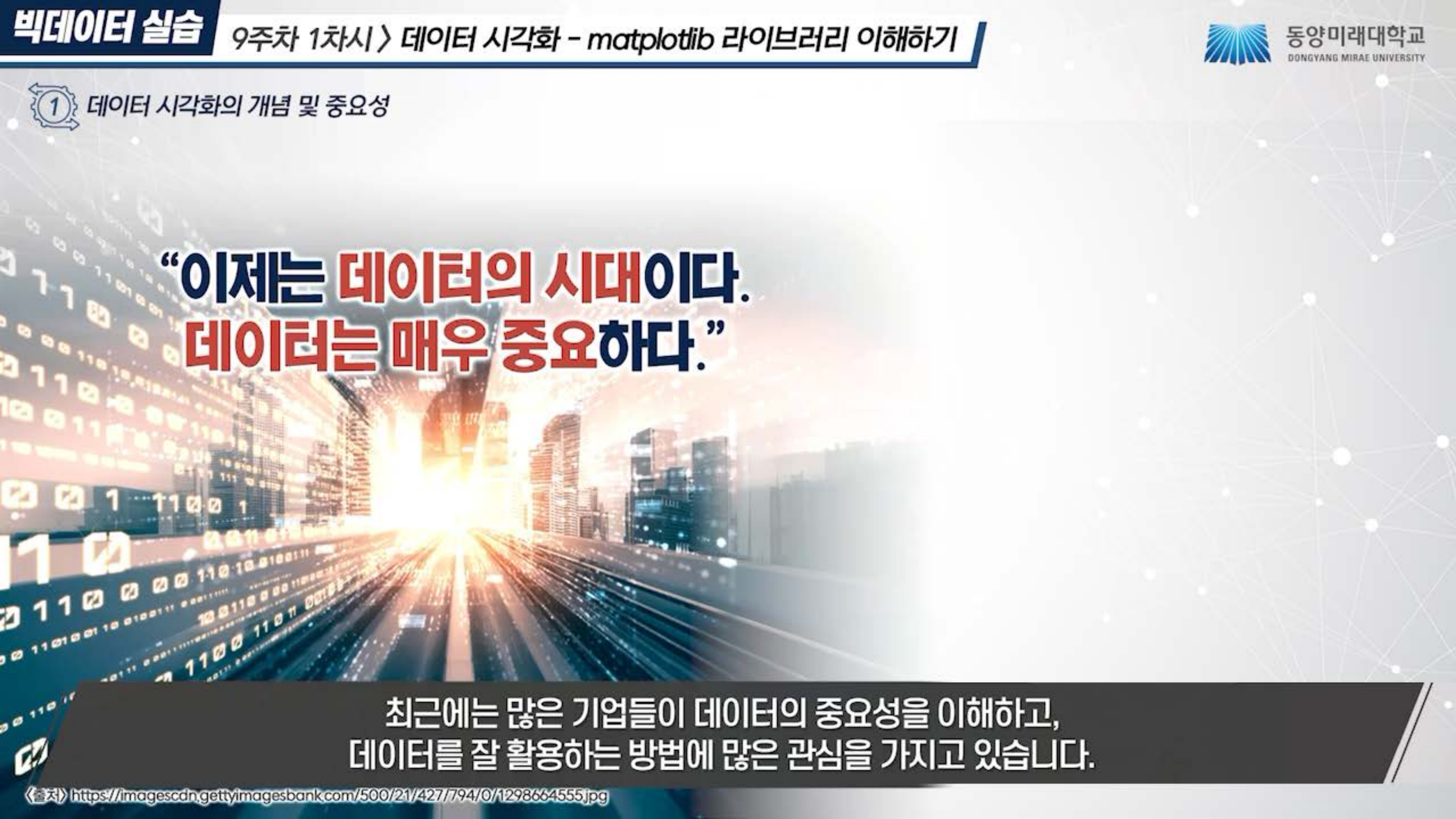
3/ Matplotlib이 제공하는 주요 기능(차트)

4/ 데이터, 상황별 차트 선택 가이드라인

01

데이터 시각화의 개념 및 중요성

① 데이터 시각화의 개념 및 중요성



최근에는 많은 기업들이 데이터의 중요성을 이해하고,
데이터를 잘 활용하는 방법에 많은 관심을 가지고 있습니다.

〈출처〉 https://imagescdn.gettyimagesbank.com/500/21/427/794/0/1298664555.jpg

① 데이터 시각화의 개념 및 중요성

✓ 기업

- ◉ 데이터를 활용한 사용자 맞춤형 서비스 제공

✓ 국가

- ◉ 데이터를 통합적으로 관리하고 구축하려는 시도
- ◉ 복지의 사각지대를 해소, 교통 체계를 구축하는 등
  국민들의 삶의 질을 높이기 위한 서비스에 활용

✓ 개인

- ◉ 가계부 작성, 주식이나 부동산과 같은 재테크를 목적으로
  경제 상황을 이해하기 위한 데이터 분석

① 데이터 시각화의 개념 및 중요성

① 데이터 시각화의 개념 및 중요성

어떻게 하면 데이터를 잘 활용할 수 있을까?

〈출처〉 https://imagescdn.gettyimagesbank.com/500/20/955/539/0/1289426915.jpg

① 데이터 시각화의 개념 및 중요성

시각화의 장점

1 많은 양의 데이터를 한눈에 볼 수 있음

행정구역별(읍면동)	2017 총인구 (명)
서울특별시	9,741,871
부산광역시	3,416,918
대구광역시	2,453,041
인천광역시	2,925,967
광주광역시	1,496,172
대전광역시	1,525,849
울산광역시	1,157,077
세종특별자치시	276,589
경기도	12,851,601
강원도	1,521,386
충청북도	1,611,009
충청남도	2,162,426
전라북도	1,826,174
전라남도	1,792,319
경상북도	2,677,058
경상남도	3,345,293
제주특별자치도	641,757

〈출처〉 통계청

① 데이터 시각화의 개념 및 중요성

빅데이터 분석을 잘 하려면?

✓ 데이터 자체의 특성을 이해해야 함

데이터 자체의 특성은 데이터 시각화를 통해 효과적으로 이해할 수 있습니다.

〈출처〉 https://imagescdn.gettyimagesbank.com/500/21/611/177/0/1294991015.jpg

① 데이터 시각화의 개념 및 중요성

시각화의 장점

② 효과적인 분석 결과를 공유하여 사용자들이 데이터 기반의 의사결정을 할 수 있음

◉ 데이터 분석을 통해서 얻은 인사이트를 공유할 때

여러 차트와 표 등으로 구성된 보드.
주로 중요한 데이터의 지표
(KPI, Key Performance Indicator)를
표현하는데 많이 활용

〈출처〉 https://imagescdn.gettyimagesbank.com/500/20/127/314/0/1289709424.jpg

① 데이터 시각화의 개념 및 중요성

시각화의 장점

2 효과적인 분석 결과를 공유하여 사용자들이 데이터 기반의 의사결정을 할 수 있음

예

카카오톡 회사의 중요한 데이터 지표는?

총 사용자수, 현재 이용 중인 사용자수,
메시지의 개수나 크기, 실시간 광고 수익

일목요연하게 볼 수 있도록 대시보드로 구성

① 데이터 시각화의 개념 및 중요성

시각화의 장점

② 효과적인 분석 결과를 공유하여 사용자들이 데이터 기반의 의사결정을 할 수 있음

만들어진 대시보드는 구성원에게 공유되고,
다른 관점에서 인사이트를 도출할 수 있게 도와줌

↓

새로운 기회를 찾고,
데이터를 근거로 중요한 의사결정을 내릴 수 있음

① 데이터 시각화의 개념 및 중요성

시각화의 장점

✓ 데이터 시각화

◉ 데이터를 효과적으로 차트나 그래프를 이용해서
사용자들이 이해하기 쉽게 만들어 주는 과정

➡ 복잡한 데이터를 쉽게 이해

➡ 분석 결과를 효과적으로 공유하여
데이터 기반 의사결정을 가능하게 함

02

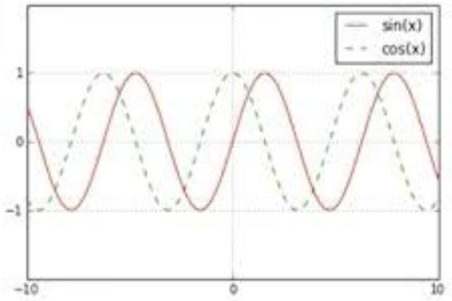
Matplotlib 라이브러리 소개

② Matplotlib 라이브러리 소개

IPython

```
In [14]: %pylab inline
Populating the interactive namespace from numpy and matplotlib

In [15]: plot(x, sin(x), "-r", label = "sin(x)")
    ...: plot(x, cos(x), "--g", label = "cos(x)")
    ...: xticks([-10, 0, 10])
    ...: yticks([-1, 0, 1])
    ...: ylim(-2, 2)
    ...: legend()
    ...: grid()
    ...:
```




```
In [16]:
```

matplotlib

〈출처〉 https://www.flickr.com/photos/schockwellenreiter/17734848325/

② Matplotlib 라이브러리 소개

matplotlib

- ✓ 파이썬에서 자료를 차트(chart)나 플롯(plot)으로 시각화(visulaization)하는 패키지
- ✓ 데이터 시각화를 위해 가장 많이 활용되는 패키지
- ✓ 2002년 파이썬에서 MATLAB과 유사한 인터페이스를 지원하고자 프로젝트가 시작됨
- ✓ 최근 bokeh, seaborn, plotly과 같은 새로운 시각화 패키지들도 많이 활용

② Matplotlib 라이브러리 소개

matplotlib

✓ 라인 플롯, 바 차트, 파이차트, 히스토그램 등
정형화된 차트나 플롯 이외에도
저수준 api를 사용한 다양한 시각화 기능 제공

사용자 입장에서는 난해하지만,
더 정교한 작업이 가능

② Matplotlib 라이브러리 소개

✓ Matplotlib

◉ 전체 패키지

✓ Pyplot

◉ matplotlib에 있는 최상위 모듈

➡ Pyplot에 있는 모든 함수는
현재 Figure의 현재 Axes에 수행됨

figure와 axes의 개념은 matplotlib의 함수의 구조를 이해하는데
중요하기 때문에 이후에 다시 설명하겠습니다.

② Matplotlib 라이브러리 소개

✓ pylab

◉ pyplot와 numpy를 하나의 네임스페이스로 임포트한 모듈

➡ Deprecated

➡ 대신 pyplot을 사용

➡ 예제가 pylab으로 되어 있다면
pyplot으로 변경해서 사용하거나 또는 pyplot으로
만들어진 예제를 찾아서 활용하는 것이 좋음

② Matplotlib 라이브러리 소개

Matplotlib.pyplot

- ✓ matplotlib의 핵심 모듈
- ✓ 데이터를 시각화하기 위한 대부분의 기능을 제공

② Matplotlib 라이브러리 소개

Matplotlib.pyplot

✓ figure

- ◉ 그래프를 그리기 위한 하얀 도화지

✓ axes (subplot)

- ◉ figure 안에 그리는 그래프
- ◉ 하나의 figure 안에 여러 개의 axes를 그릴 수 있으나,
  하나의 axes는 여러 개의 figure에 속할 수 없음
- ◉ axes는 2개의 축(x, y축)을 가짐

② Matplotlib 라이브러리 소개

figure와 axes의 차이점

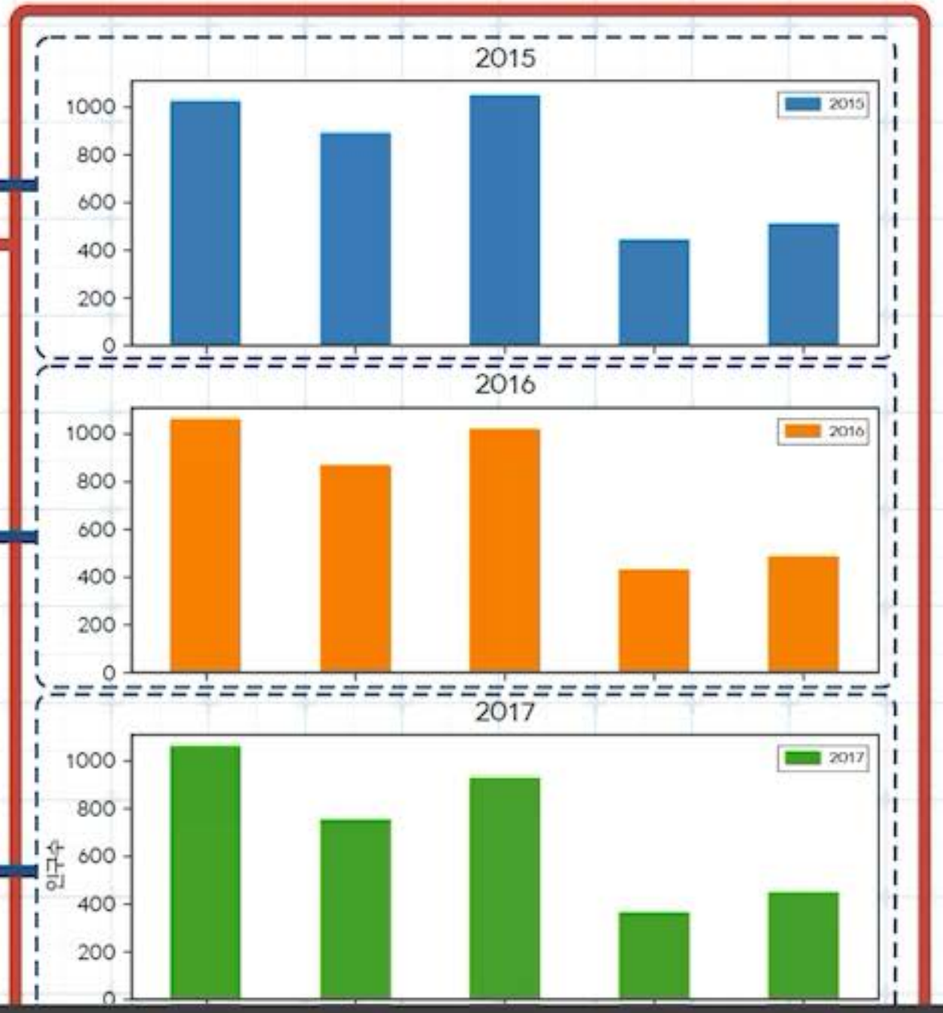
[Figure] 그래프가 그려지는 캔버스

· 모든 그림은 Figure 객체에 포함됨 (Matplotlib, figure)

· 원래는 figure( ) 함수를 통해 객체를 생성하지만
plot( ) 함수는 자동으로 Figure생성

[Axes] Figure 안에 있는 각각의 그래프

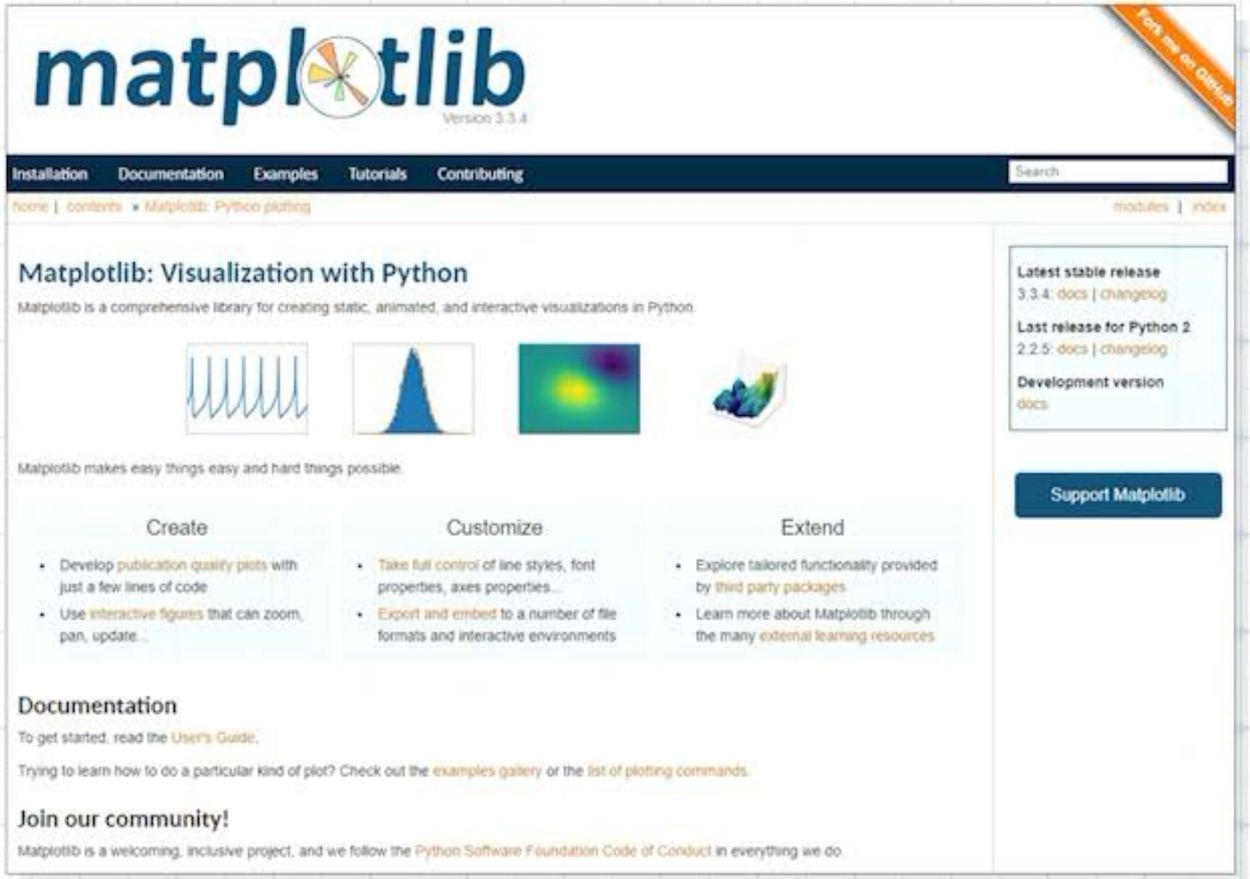
Figure와 마찬가지로,
원래는 subplot( )으로 생성해야 하지만,
plot( ) 함수에서 자동으로 생성



Figure와 axes 객체의 차이점을 이해하고,
각 객체에서 제공해주는 함수들을 적절하게 활용할 수 있어야 합니다.

2 Matplotlib 라이브러리 소개

7 Matplotlib 공식 홈페이지



〈출처〉 https://matplotlib.org/stable/index.html

② Matplotlib 라이브러리 소개

Matplotlib 〉 Example

✓ Matplotlib을 사용해서
데이터를 시각화한
여러 가지 샘플 예제를 제공

〈출처〉 https://matplotlib.org/stable/index.html

② Matplotlib 라이브러리 소개

Matplotlib 〉 Example 〉 plotting categorical variables

Plotting categorical variables ¶

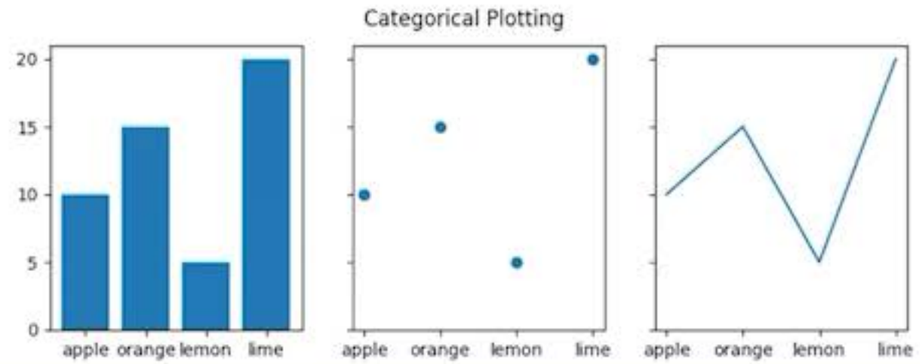
How to use categorical variables in Matplotlib.

Many times you want to create a plot that uses categorical variables in Matplotlib. Matplotlib allows you to pass categorical variables directly to many plotting functions, which we demonstrate below.

```
import matplotlib.pyplot as plt

data = {'apple': 10, 'orange': 15, 'lemon': 5, 'lime': 20}
names = list(data.keys())
values = list(data.values())

fig, axs = plt.subplots(1, 3, figsize=(9, 3), sharey=True)
axs[0].bar(names, values)
axs[1].scatter(names, values)
axs[2].plot(names, values)
fig.suptitle('Categorical Plotting')
```



Out

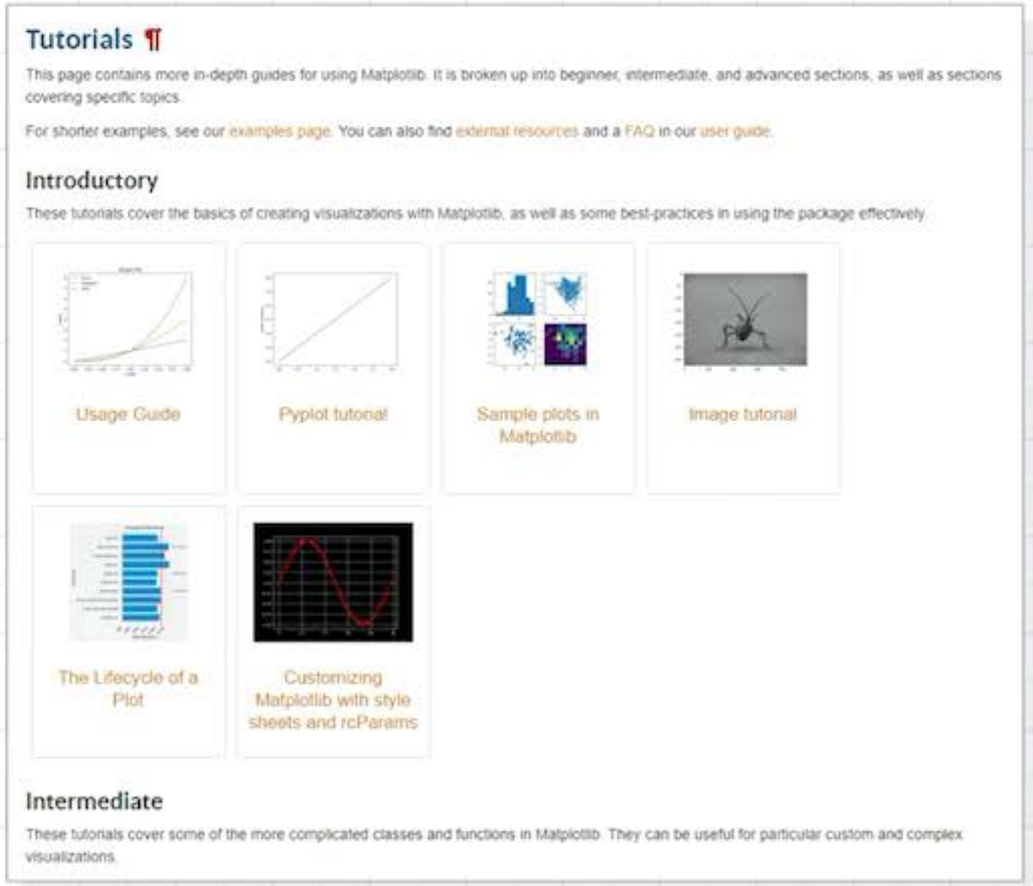
```
Text(0.5, 0.98, 'Categorical Plotting')
```

✓ Example의 코드를 보고
내가 원하는 형태로
수정하는 작업이 필요함

〈출처〉 https://matplotlib.org/stable/index.html

② Matplotlib 라이브러리 소개

Matplotlib Tutorial



- ✓ 레벨에 따른 튜토리얼 제공
  - Introductory
  - Intermediate
  - Advanced
  - Colors
  - Provisional
  - · · ·

〈출처〉 https://matplotlib.org/stable/index.html

03

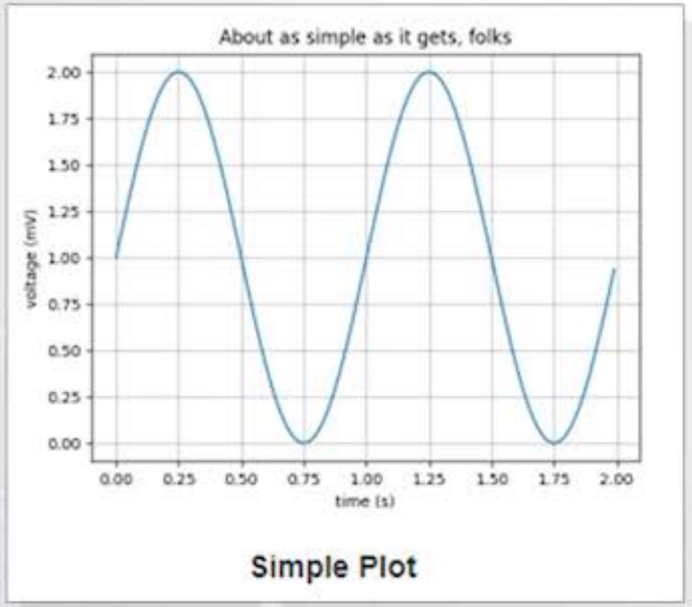
Matplotlib이 제공하는 주요기능 (차트)

③ Matplotlib이 제공하는 주요 기능 (차트)

7 Line Plot (선 그래프)

✓ Plot( )

◉ x축이 연속형 값인 경우 주로 활용
(e.g. 날짜, 년도와 같은 시계열 데이터)



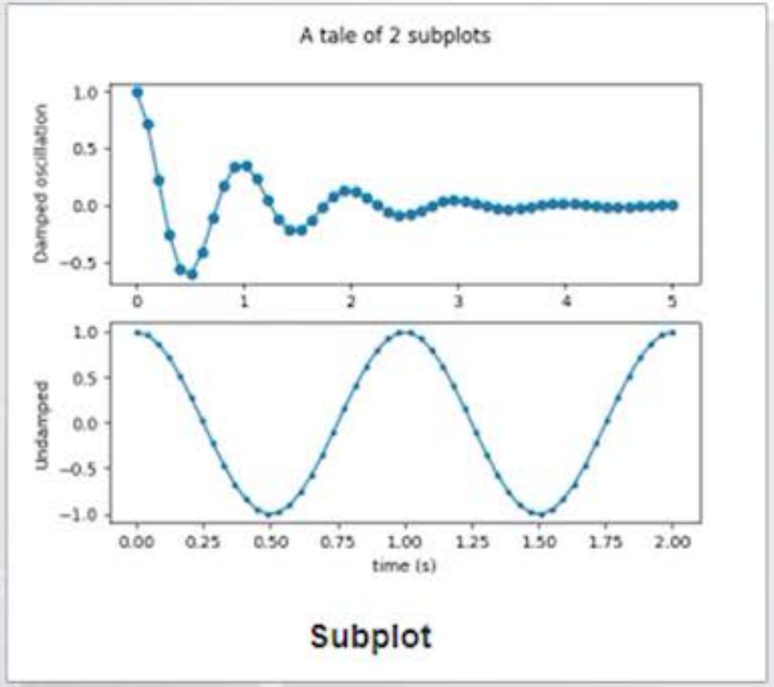
Simple Plot

③ Matplotlib이 제공하는 주요 기능 (차트)

7 Line Plot (선 그래프)

✓ subplot( )

◉ 하나의 figure에 여러 개의 axes(subplot) 생성

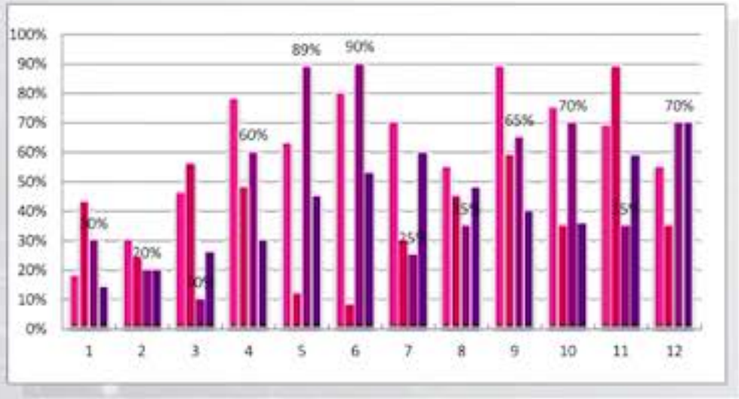


③ Matplotlib이 제공하는 주요 기능 (차트)

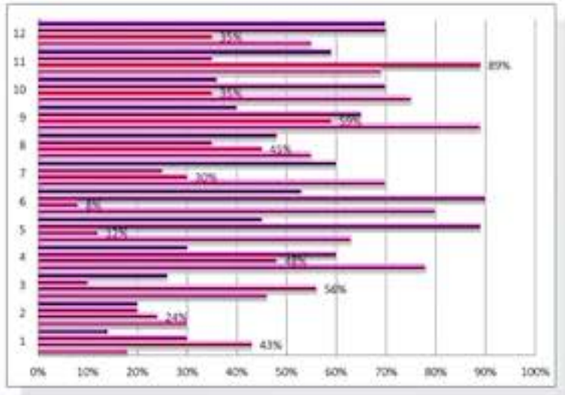
7 Bar plot (막대 그래프)

✓ x축의 값이 주로
범주형 변수(categorical variable)인
경우에 주로 활용 (e.g. 성별, 기기유형, …)

bar( )



barh( )



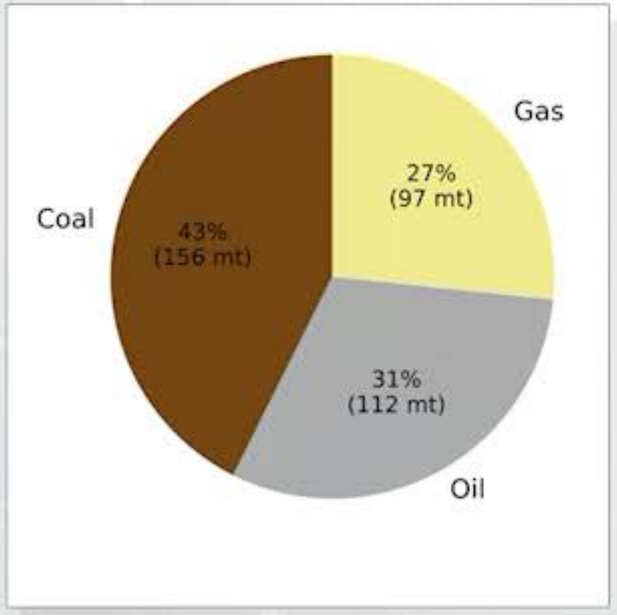
〈출처〉 https://imagescdn.gettyimagesbank.com/500/20/366/529/0/1270591014.jpg

③ Matplotlib이 제공하는 주요 기능 (차트)

7 Pie plot (파이 차트)

✓ pie( )

◉ 점유율과 같이
y값의 합이 100인 데이터 시각화에 주로 활용



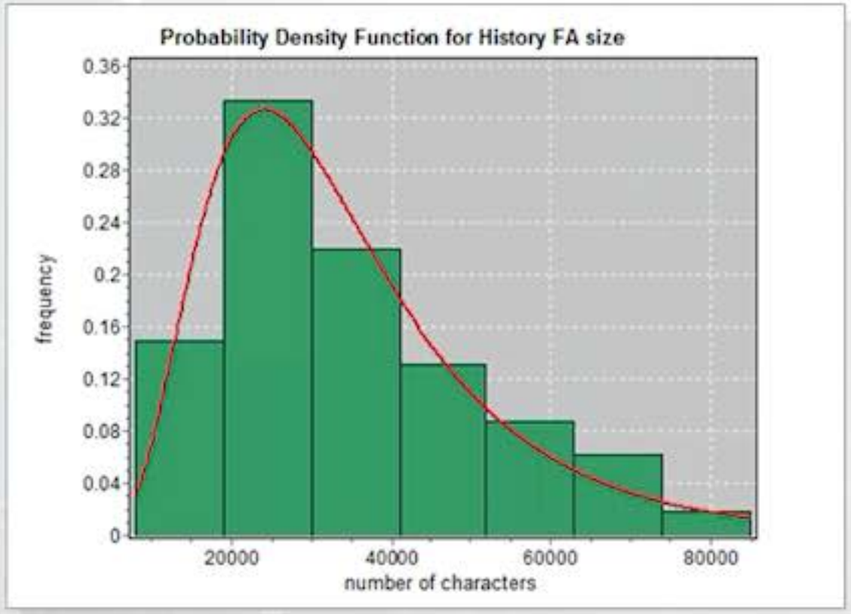
〈출처〉 https://commons.wikimedia.org/wiki/File:Co2_by_fuel_pie_chart.svg

③ Matplotlib이 제공하는 주요 기능 (차트)

ㄱ 히스토그램

✓ 히스토그램

◉ 숫자 데이터를 동일한 폭의 통(bucket)으로 그룹화하여 자료를 표현하는 방식



〈출처〉 https://commons.wikimedia.org/wiki/File:Fa_history_histogram.png

③ Matplotlib이 제공하는 주요 기능 (차트)

데이터에서 연령대별 명 수를 그리는 방법

1 히스토그램

이름	나이
A	13
B	38
C	21
D	25
E	42
F	35
G	27

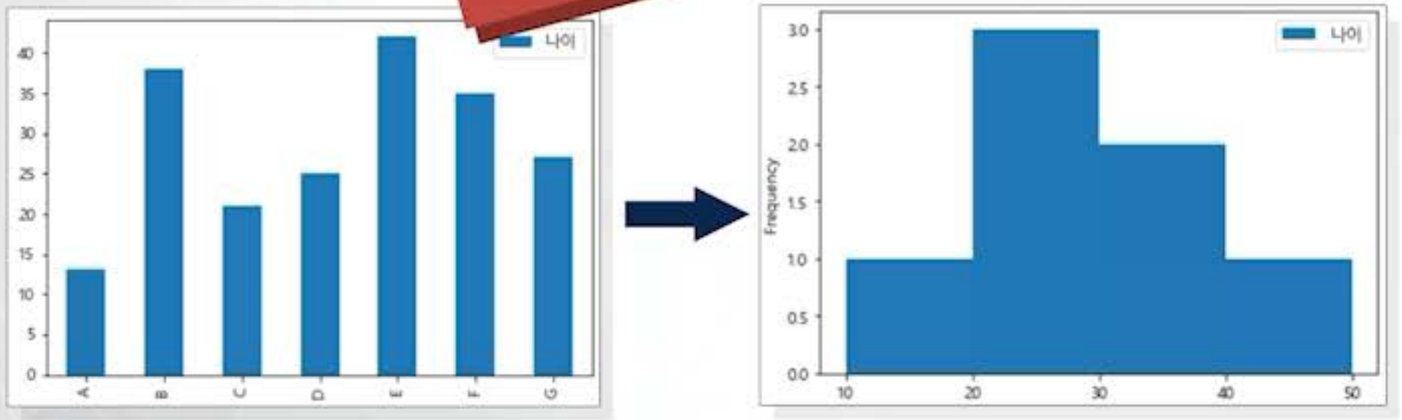
✓ 그룹집계(groupby)도 가능하지만 나이대 컬럼을 새로 추가해야 함

3 Matplotlib이 제공하는 주요 기능 (차트)

데이터에서 연령대별 명 수를 그리는 방법

1 히스토그램

hist( )



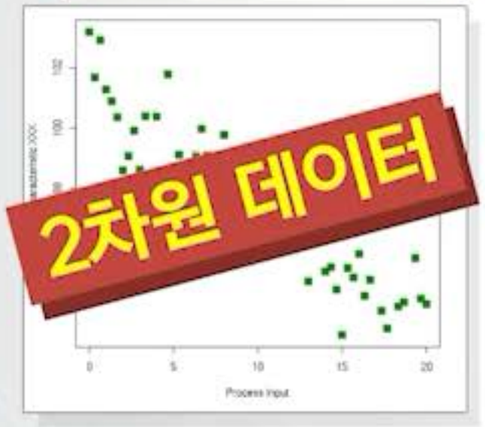
단순 연령 정보가 아닌 연령대별 분포도를 그린 것을 히스토그램이라고 합니다.

③ Matplotlib이 제공하는 주요 기능 (차트)

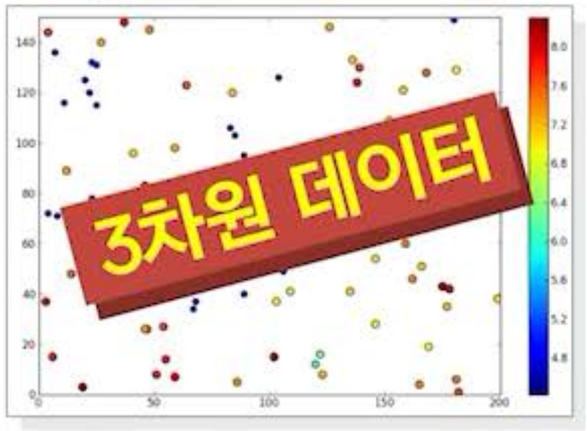
2 scatter( ) : 산점도 그래프

✓ 데이터의 분포를 볼 때 주로 사용

스캐터 차트



버블 차트



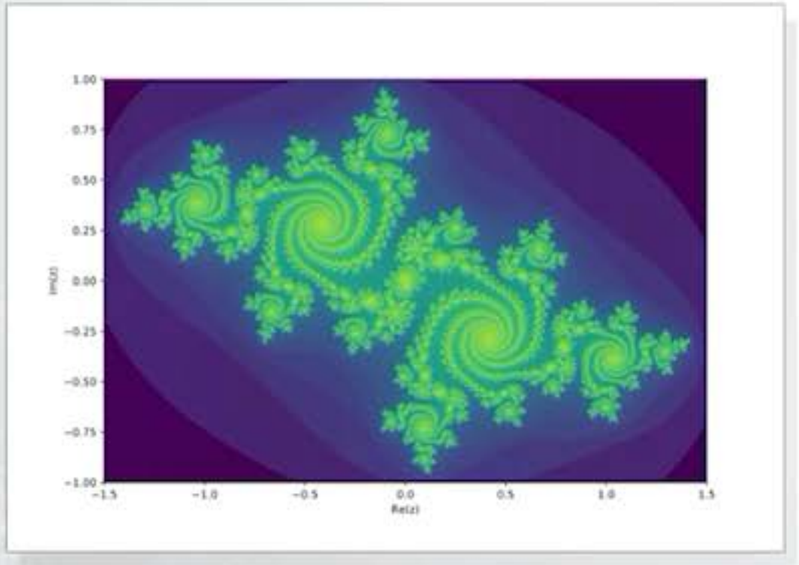
〈출처〉 (좌) https://commons.wikimedia.org/wiki/File:Scatter_diagram_for_quality_characteristic_XXX.svg
(우) https://commons.wikimedia.org/wiki/File:Sparse_data.png

③ Matplotlib이 제공하는 주요 기능 (차트)

7 데이터에서 연령대별 명 수를 그리는 방법

✓ imshow( )

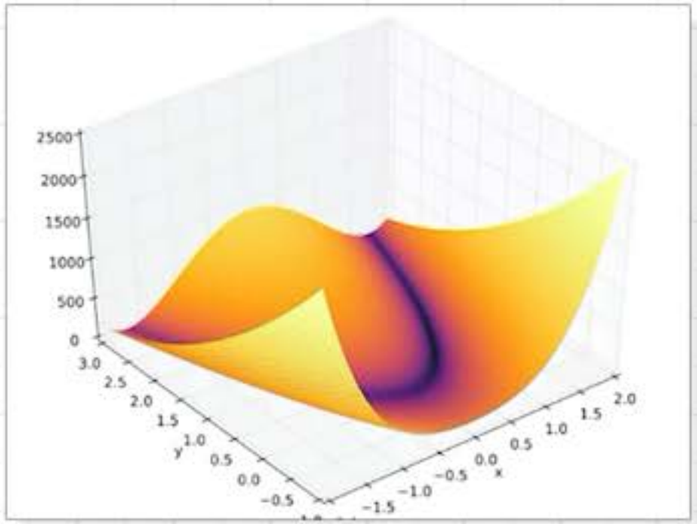
◉ 이미지를 그릴 때 사용

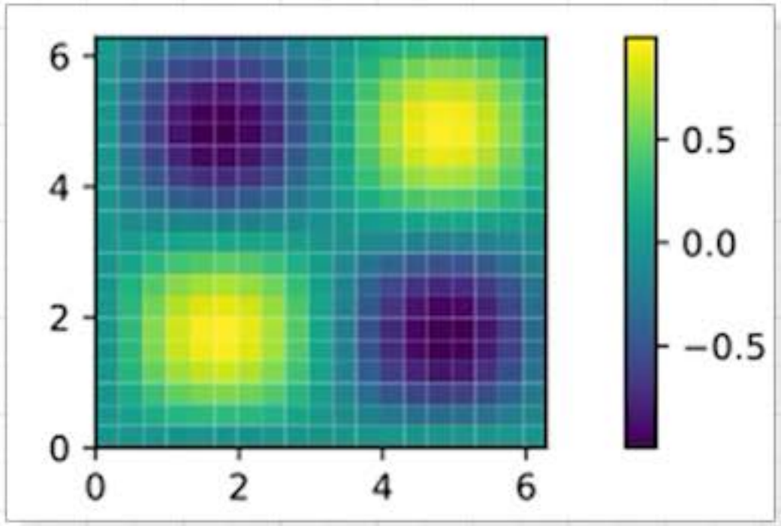


〈출처〉 https://commons.wikimedia.org/wiki/File:Julia_set_plotted_with_Matplotlib.svg

3 Matplotlib이 제공하는 주요 기능 (차트)

그 외 plot들





✓ 다양한 형태의 plot 타입을 제공하므로 Matplotlib을 통해 구현 가능

✓ 각각의 파라미터들을 잘 지정해야 원하는 결과를 얻을 수 있음

〈출처〉 (좌) https://commons.wikimedia.org/wiki/File:Cartes_couleur_sinFoisSin_python_matplotlib.svg (우) https://commons.wikimedia.org/wiki/File:Mpl_example_Rosenbrock_function.svg

04

데이터, 상황별 차트 선택 가이드라인

④ 데이터, 상황별 차트 선택 가이드라인

"차트들을 통해서 그래프를 그리는 이유는?"

분석한 결과나 데이터를 잘 이해하기 위해서

"데이터의 특성, 분석 결과, 데이터 상황별로
어떤 차트를 사용해야 더 효과적으로
데이터를 시각화할 수 있을지 고민해야 함"

4 데이터, 상황별 차트 선택 가이드라인

그래프 선택 가이드

- ✓ Dr. Andrew Ablela가 제안한 차트 선택 방법('2009)

④ 데이터, 상황별 차트 선택 가이드라인

4가지 분류

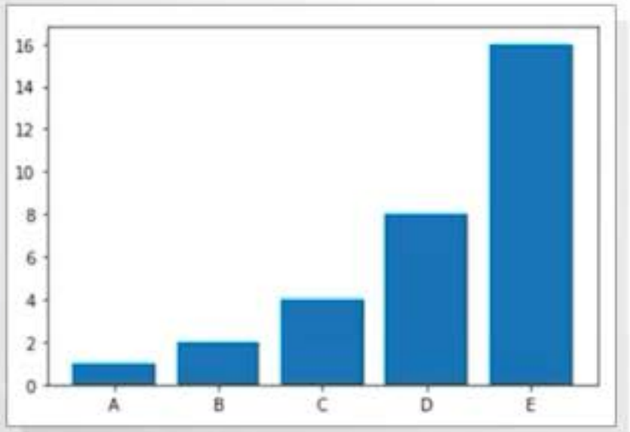
1. 비교 (Comparison)
2. 관계 (Relationship)
3. 분산 (Distribution)
4. 구성요소 (composition)

④ 데이터, 상황별 차트 선택 가이드라인

4가지 분류

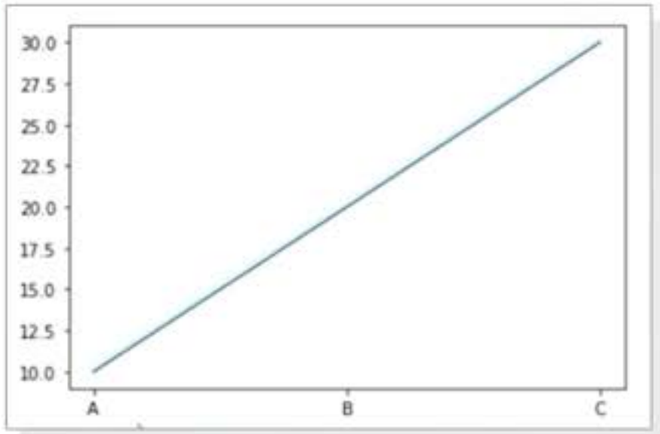
1 비교 (Comparison)

막대 그래프



✓ 범주형 데이터 비교

라인 그래프



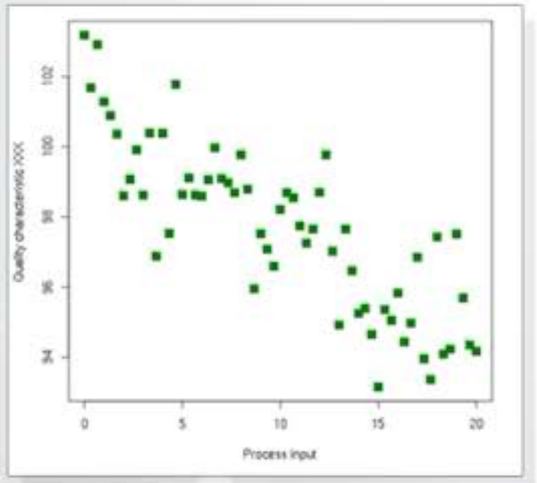
✓ 시계열 데이터 비교

④ 데이터, 상황별 차트 선택 가이드라인

4가지 분류

② 관계 (Relationship)

스캐터 차트



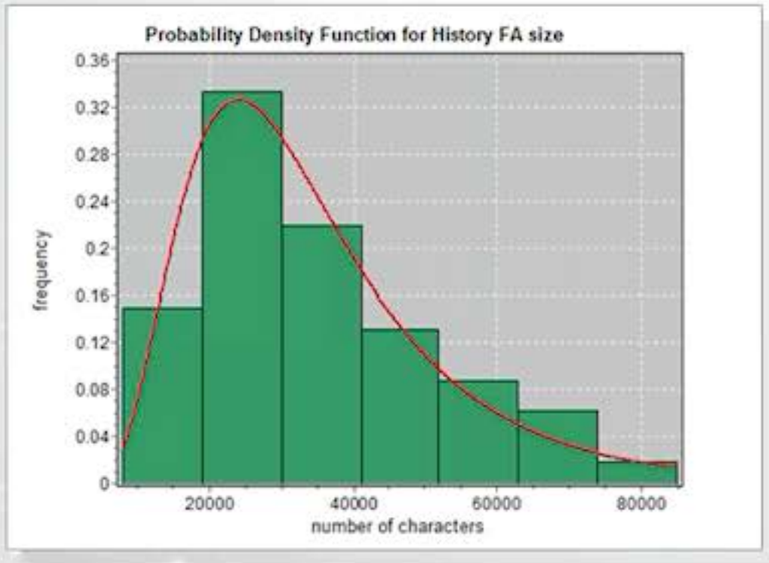
〈출처〉 (좌) https://commons.wikimedia.org/wiki/File:Scatter_diagram_for_quality_characteristic_XXX.svg

④ 데이터, 상황별 차트 선택 가이드라인

4가지 분류

③ 분산 (Distribution)

히스토그램



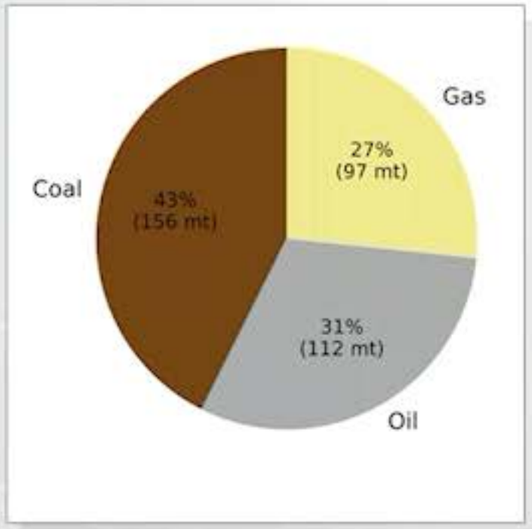
〈출처〉 https://commons.wikimedia.org/wiki/File:Fa_history_histogram.png

④ 데이터, 상황별 차트 선택 가이드라인

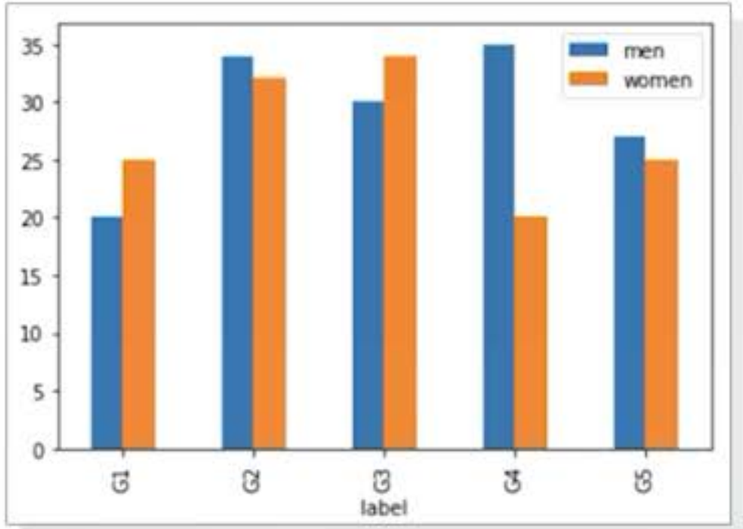
4가지 분류

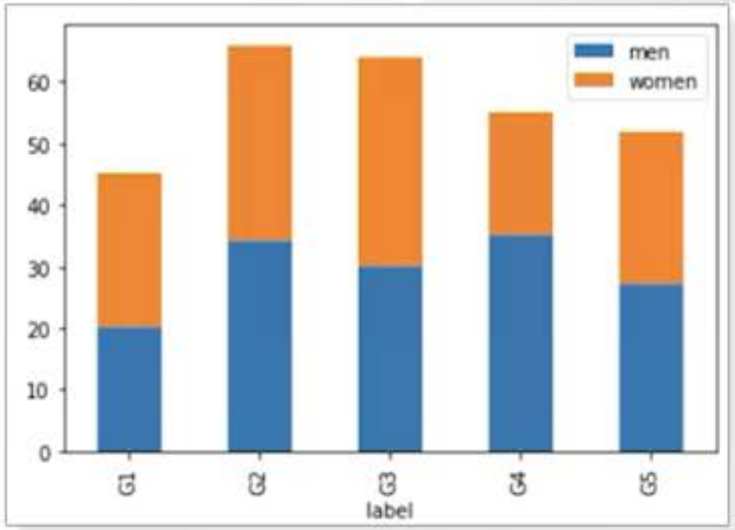
④ 구성요소 (composition)

파이 차트



막대 그래프





〈출처〉 교수자 제공 이미지

학습완료

1/ 데이터 시각화의 개념 및 중요성

2/ Matplotlib 라이브러리 소개

3/ Matplotlib이 제공하는 주요 기능(차트)

4/ 데이터, 상황별 차트 선택 가이드라인