





- ✓ 데이터 시각화 패키지 matplotlib
- ✓ 수리 연산 패키지 numpy
- ▼ 패키지 matplotlib를 사용한 선 그리기





- ☑ 데이터 시각화 패키지 matplotlib의 홈페이지를 활용해 시각화 개념를 이해할 수 있다.
- 😈 패키지 matplotlib를 사용해 2차원 평면에 그래프를 그릴 수 있다.

LESSON 01

# matplotlib 라이브러리소개





- → matplotlib 라이브러리란? (1/4)
- ❤️ 파이썬으로 데이터를 시각화할 때 가장 많이 사용하는 라이브러리 중 하나
- ❤️ 주로 2D 형태의 그래프, 이미지 등을 그릴 때 사용
- ❷ 실제 과학 컴퓨팅 연구 분야나 인공지능 연구 분야에서도 많이 활용



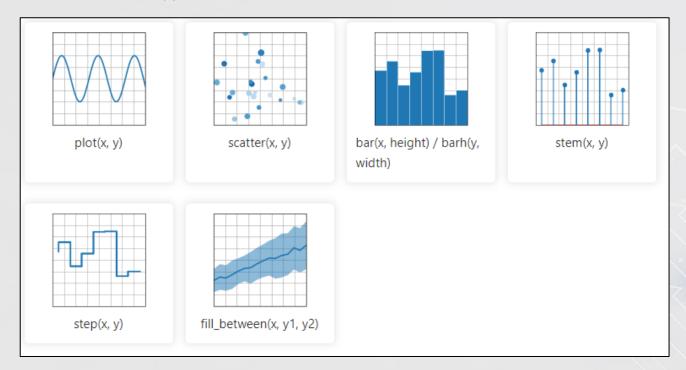
#### [전] matplotlib 라이브러리 소개



# → matplotlib 라이브러리란? (2/4)

## ☑ 어떤 그래프들을 그릴 수 있나요?

#### Basic Plot Types



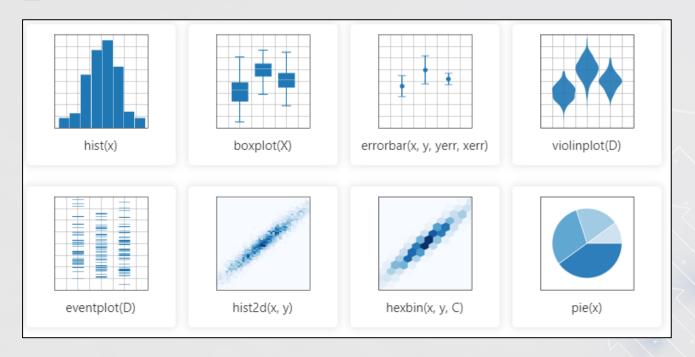
#### M matplotlib 라이브러리 소개



# → matplotlib 라이브러리란? (3/4)

# ☑ 어떤 그래프들을 그릴 수 있나요?

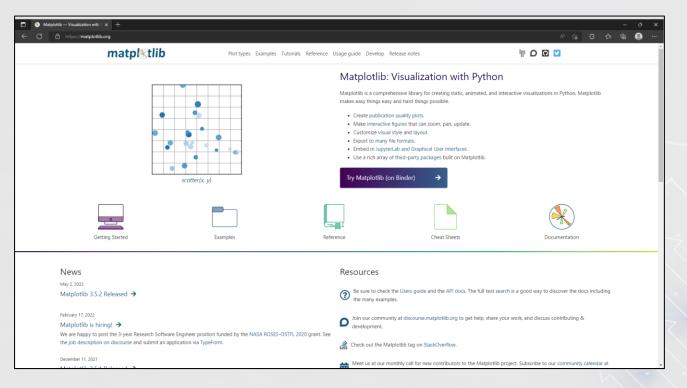
#### Statistics Plots





## → matplotlib 라이브러리란? (4/4)

# → matplotlib 書酬の以: https://matplotlib.org/



#### M matplotlib 라이브러리 소개



- → 2002년 존 헌터가 시작, 2D 그래프를 위한 데스크탑 패키지
- ❤️ 파이썬에서 자료를 차트(chart)나 플롯(plot)으로 시각화(visualization)하는 패키지
- ❷ 정형화된 차트나 플롯 이외에도 저수준 API를 사용한 다양한 시각화 기능을 제공
  - ☑ 라인 플롯(line plot)
  - ✓ 스캐터 플롯(scatter plot)
  - ☑ 바 차트(bar chart)
  - ☑ 히스토그램(histogram)
  - 박스 플롯(box plot)

#### M matplotlib 라이브러리 소개



# → matplotlib 라이브러리 사용

**→** matplotlib의 pyplot 모듈 불어오기

import matplotlib.pyplot

from matplotlib import pyplot

# ■ matplotlib 라이브러리의 pyplot 모듈을 'plt'라는 별명으로 부르기 (alias)

import matplotlib.pyplot as plt

from matplotlib import pyplot as plt

LESSON 02

# 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리

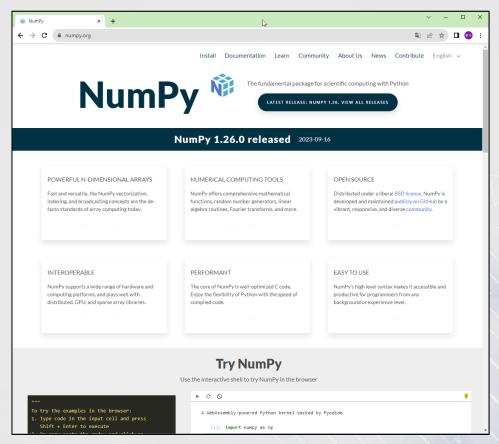


#### ☑근, 숫자 데이터를 쉽게 다루도록 돕는 numpy 관이브러긴



### → Numpy 홈페이지

- numpy.org
  - ☑ 수식 관련 유용한 자료 제공
  - ☑ 난수 관련 유용한 자료 제공
  - ☑ 자료 분포 관련 유용한 자료 제공



#### ☑2. 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리



#### → Numpy 설치 점검



# 🤪 명령 pip show numpy

vamun work aid

Name: numby Version: 1.24.3

Summary: Fundamental package for array computing in Python

Home-page: https://www.numpy.org Author: Travis E. Oliphant et al.

Author-email:

License: BSD-3-Clause

Location: C: #ProgramData #anaconda3-23-09 #Lib #site-packages

Requires:

Required-by: astropy, bokeh, Bottleneck, contourpy, daal4py, datasets, datashader, datashape, gensim, h5py, holoviews, hyplot, imagecodecs, imageio, imbalanced-lear n, matplotlib, mkl-fft, mkl-random, numba, numexpr, pandas, patsy, pyarrow, pyerf a. PyWavelets, scikit-image, scikit-learn, scipy, seaborn, statsmodels, tables, ti fffile, transformers, xarray

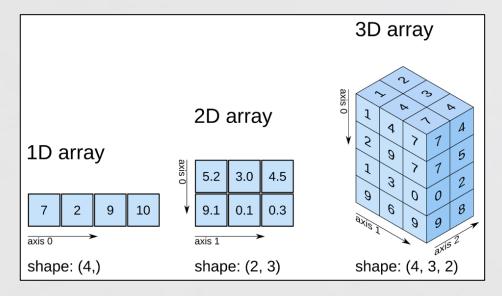
Note: you may need to restart the kernel to use updated packages.

#### ☑근 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리



# → NumPy quickstart

- 🤪 주묘 객체
  - 🔽 ndarray
    - ◆ 다차원 배열
  - https://numpy.org/devdocs/user/quickstart.html



#### ☑근, 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리



- → Ndarray 유형을 쉽게 만드는 메소드
- np.array()
- np.arange()

```
import numpy as np
a = np.array([6, 7, 8])
а
array([6, 7, 8])
type(a)
numpy.ndarray
b = np.arange(15)
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14])
type(b)
numpy.ndarray
```

#### ☑근, 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리



# ⊸ 배열의 형태를 수정



```
c = np.arange(15).reshape(3, 5)
array([[ 0, 1, 2, 3, 4],
      [5, 6, 7, 8, 9],
[10, 11, 12, 13, 14]])
c.shape
(3.5)
c.ndim
c.dtype.name
'int32'
c.itemsize # 각 원소의 메모리 바이트 크기
c.size
15
type(c)
numpy.ndarray
```

#### ☑2 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리



- ⊸ 등분하는 수의 배열 생성
- np.linspace(start, stop, num)
  - ☑ start에서 stop사이를 num개의 수로 등분하는 수의 배열
  - ☑ 결국 start와 stop을 포함해서 num-1 등분하는 수의 배열

```
# O(start)에서 10(stop)사이를 11(num)개의 수로 등분.
# 결국 0과 10을 포함해서 10(11-1)등분하는 수의 배열
np.linspace(0, 10, 11)
array([0., 1., 2., 3., 4., 5., 6., 7., 8., 9., 10.])
np.linspace(0, 10, 10)
array([ 0. , 1.11111111, 2.22222222, 3.33333333, 4.44444444,
       5.5555556. 6.66666667. 7.77777778. 8.88888889. 10.
# (stop - start) / (num - 1)
(10-0) / (10-1)
1.11111111111111112
```

#### ☑근, 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리



## → 일정한 수의 나열을 원할 때 자주 사용

# **➢ Numpy.linspace()**

```
np.linspace(0, 5, 6)
array([0., 1., 2., 3., 4., 5.])

np.linspace(0, 5, 5)
array([0. , 1.25, 2.5 , 3.75, 5. ])

# (stop - start) / (num - 1)
(5-0) / (5-1)
1.25
```

#### ☑2. 숫자 데이터를 쉽게 다루도록 돕는 numpy 라이브러리



## → np.linspace() 도움말



```
In [28]: help(np.linspace)
        Help on function linspace in module numby
        linspace(start, stop, num=50, endpoint=True, retstep=False, dtype=None, axis=0)
            Return evenly spaced numbers over a specified interval.
            Returns `num` evenly spaced samples, calculated over the
                                                                             print(np.linspace(0, 2, 3))
            interval ['start', 'stop'].
                                                                             • 0에서 2까지, 일정한 간격이
            The endpoint of the interval can optionally be excluded
            .. versionchanged:: 1.16.0
                                                                               되도록 숫자 3개를 반환
               Non-scalar `start` and `stop` are now supported.
                                                                            • 결국 (3-1) 등분
            .. versionchanged:: 1.20.0
               Values are rounded towards ``-inf`` instead of ``O`` when an
               integer ``dtype`` is specified. The old behavior can
               still be obtained with ``np.linspace(start. stop. num).astype(int)`
            Parameters
            start : array like
               The starting value of the sequence.
            stop : array_like
               The end value of the sequence, unless `endpoint` is set to False
               In that case, the sequence consists of all but the last of ``num + 1`
               evenly spaced samples, so that 'stop' is excluded. Note that the step
               size changes when `endpoint` is False.
            num : int. optional
               Number of samples to generate. Default is 50. Must be non-negative.
            endpoint : bool, optional
               If True, `stop` is the last sample. Otherwise, it is not included.
               Default is True.
            retstep : bool, optional
               If True, return ('samples', 'step'), where 'step' is the spacing
               between samples.
```

LESSON 03

# matplotlib. pyplot의plot 함수로 그래프 그리기





#### ⊸ 설치 점검



## 🤪 서드 파티 모듈 matplotlib과 numpy 설치 점검

#### #설치 점검

pip show matplotlib

Name: matplotlib Version: 3.7.2

Summary: Python plotting package Home-page: https://matplotlib.org

Author: John D. Hunter, Michael Droettboom Author-email: matplotlib-users@python.org

License: PSF

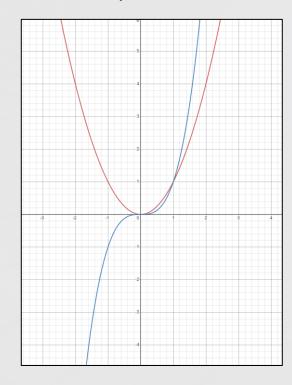
Location: C: #ProgramData#anaconda3-23-09#Lib#site-packages

Requires: contourpy, cycler, fonttools, kiwisolver, numpy, packaging, pillow, pypa

rsing, python-dateutil Required-by: seaborn



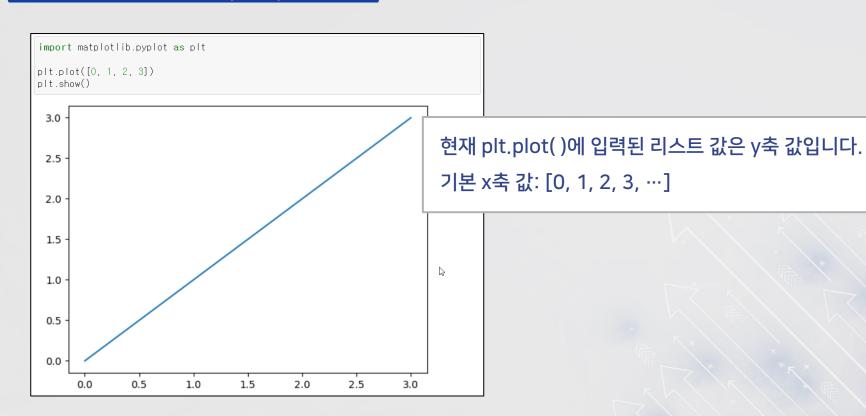
- → 선 그래프 그리기: plt.plot()
- **→** 가장 간단한 플롯은 선을 그리는 라인 플롯(line plot)
- ❷ 라인 플롯은 데이터가 시간, 순서 등에 따라 어떻게 변화하는지 보여주기 위해 사용



## 🛂 plot 함수로 선 그래프 그리기



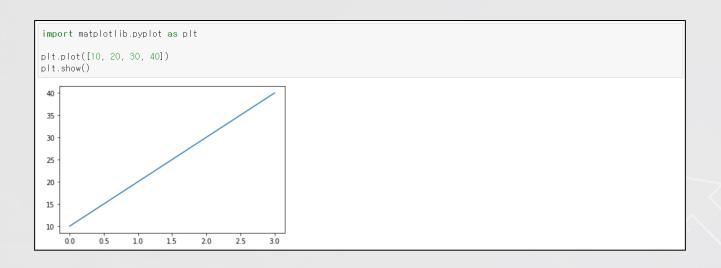
# → 선 그래프 그리기: plt.plot()



#### ☑글 plot 함수로 선 그래프 그리기



# → 선 그래프 그리기: plt.plot([y축 데이터])

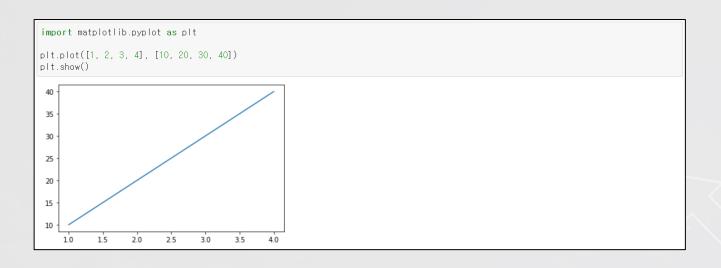


현재 plt.plot()에 입력된 리스트 값은 y축 값입니다. x축 값도 입력하고 싶다면 어떻게 해야 할까요?

#### ☑글, plot 함수로 선 그래프 그리기



# → 션 그래프 그리기: plt.plot([x축 데이터], [y축 데이터])

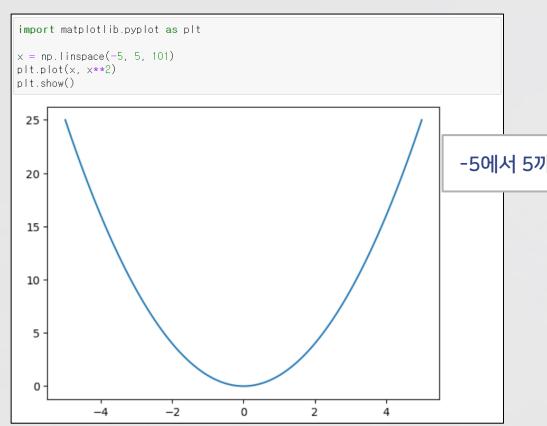


첫 번째 리스트 값은 x축 두 번째 리스트 값은 y축을 나타냅니다.

#### ☑3. plot 함수로 선 그래프 그리기



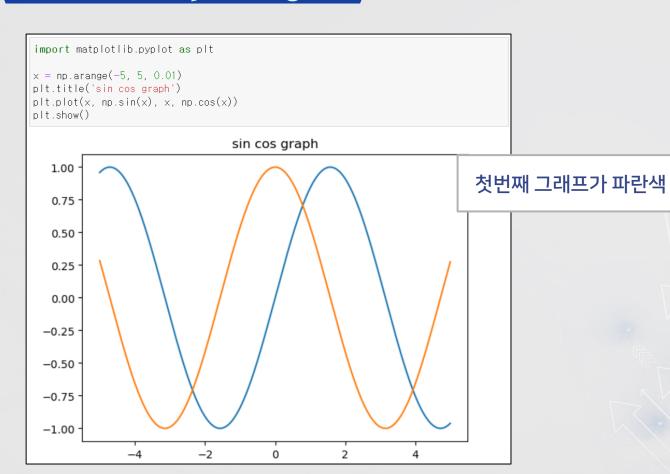
# ⊸ x의 값 지점: np.linspace()



-5에서 5까지 100등분하는 수의 나열



# ⊸ x의 값 지점: np.arange()



# SUMMARY

# 학습정긴





•••

#### 🧑 데이터 시각화 패키지 matplotlib

matplotlib.org

### 🧑 수리 연산 패키지 numpy

- numpy.org
- 가다차원배열 ndarry
- > 배열을 만드는 함수 np.array()
- > 수의 나열인 배열을 만드는 함수
  - > np.arange(start, stop, step), np.linspace(start, stop, num)







•••

## 🧿 패키지 matplotlib를 사용한 선 그리기

- >> plt.plot(ys), x=[0, 1, 2, ...]
- >> plt.plot(xs, ys)
- >> plt.plot(x1s, y1s, x2s, y2s, ···)
- >> plt.title('title')
- >> plt.show()



