의료인공지능 부트캠프 1일차 둘째시간

Jupyter와 pandas를 이용한 데이터 분석 기초

2022. 10. 07. 박주희

INSILICOGEN

www.insilicogen.com

본 문서의 모든 콘텐츠는 저작권법의 보호를 받는 저작물로 별도의 저작권 표시 또는 다른 출처를 명시한 경우를 제외하고는 (주)인실리코젠에 저작권이 있습니다.
저작권 표시 또는 기타 소유권 표시를 삭제해서도 안되며, 당사와의 협의 또는 허락없이 무단 복제, 변경, 배포를 금지합니다.
저작권 관련 문의사항이 있으시면 ix@insilicogen.com으로 연락바랍니다.
© 2022 **INSILICOGEN, INC.** ALL RIGHTS RESERVED.

INDEX

01

AI 파트, 어떤 일을 할까요?


AI Part
Data Analysis
AI Modeling

다양한 데이터를 접합니다.

오믹스 데이터

- WGS
- RNA-seq, Methyl-seq
- Microbiome
- Liquid biopsy

EMR 데이터

- EMR 차트 데이터
- Inbody 데이터

시계열 데이터

- 웨어러블 기기 데이터
- 수산물 성장 데이터

X-ray 데이터

- Chest PA
- Spine X-ray

게임 로그 데이터

- 게임 컨텐츠 데이터
- 게임 User 데이터

자연어 데이터

- 개체명 데이터
- 문장 관계 데이터

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

다양한 데이터들, 어떻게 활용 할 수 있을까?

- 갯수는 몇개일까?
- 특징은 어떤 것이 있을까?
- 어떤 분포로 구성되어 있을까?
- 특징 간 연관성이 있을까?
- 데이터를 활용하여 어떤 연구를 할 수 있을까?

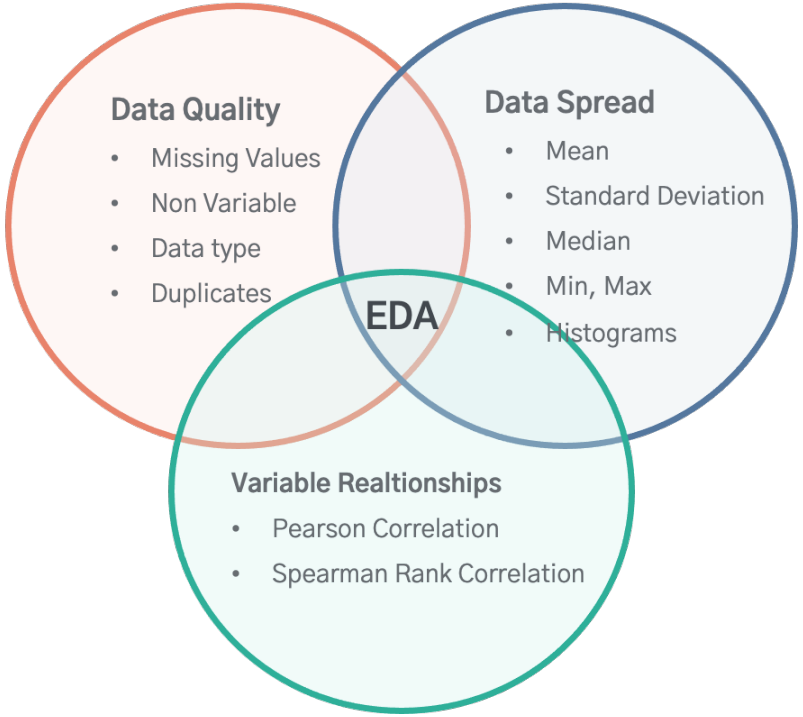
데이터를 탐색해봐야지!

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

가장 먼저 EDA, 데이터를 탐색해 봅시다

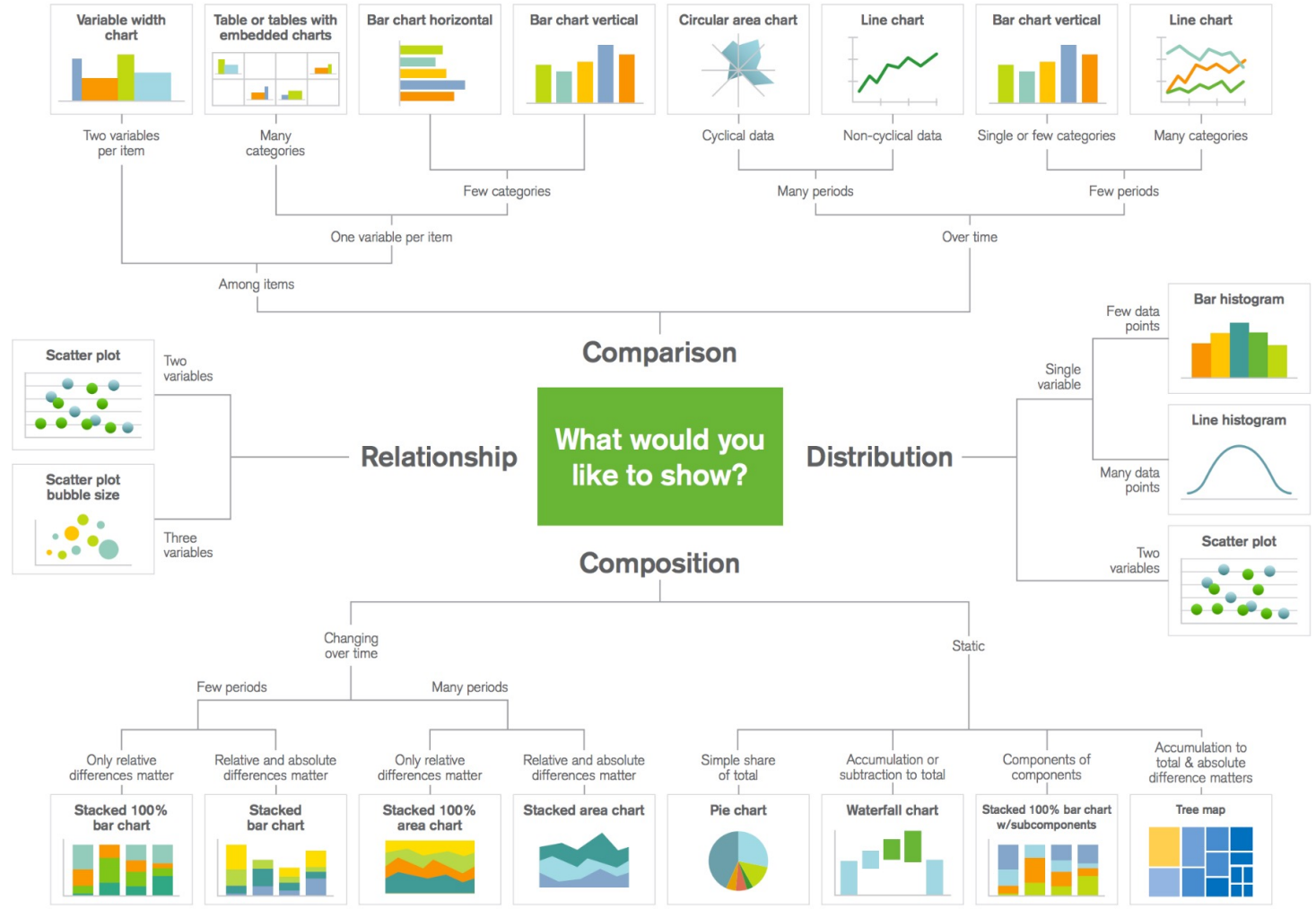
EDA란?

- Exploratory data analysis (탐색적 데이터 분석)
- 미국의 통계학자, 존 튜키가 창안한 데이터 분석 방법론으로 주어진 데이터만을 가지고도 충분한 정보를 찾을 수 있도록 한 데이터 분석 방법
- 다양한 시각화와 요약 통계 등을 활용하여 전체적인 데이터를 살펴보고 속성의 값들을 관찰하는 방법



© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

가장 먼저 EDA, 데이터를 탐색해 봅시다



Source: ©A. Abela, 2010. www.ExtremePresentation.com

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

EDA in python, 어떤 라이브러리를 활용 할까요?

데이터 핸들링

- python에서 사용하는 데이터 분석 라이브러리
- DataFrame, Series 를 다룰 수 있음
- R과 유사한 기능을 수행

- python에서 사용하는 고성능 수치계산 라이브러리
- 벡터, 행렬 등 수치 연산을 수행함.
- 내부 코드는 C언어로 구성, 빠른 연산 가능

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

EDA in python, 어떤 라이브러리를 활용 할까요?

데이터 시각화

- python에서 사용하는 데이터 시각화 라이브러리
- Lineplot, Piechart, Histogram, Scatter plot 등 다양한 차트를 지원

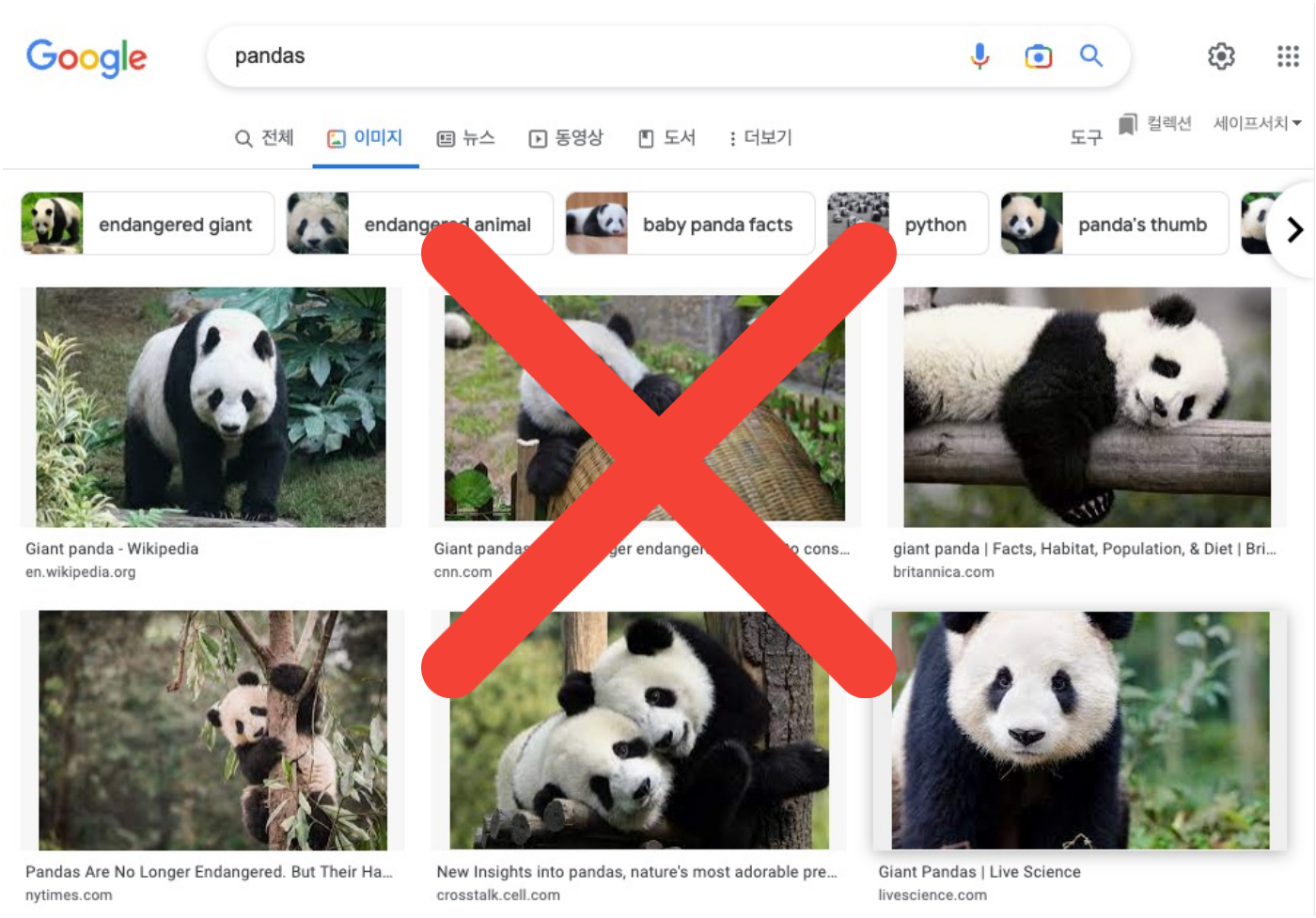
- Matplotlib, Statsmodels 라이브러리 기반
- 다양한 색상테마와 통계 차트 등 기능을 추가한 데이터 시각화 라이브러리

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

02

pandas를 배워봅시다!

pandas란?

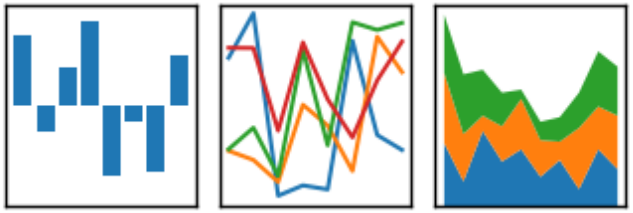


© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

pandas란?

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Python Data Analysis Library

데이터 분석에 특화된 파이썬 라이브러리

http://pandas.pydata.org/

latest version 1.5.0

```
$ conda install pandas

$ pip install pandas

>>> import pandas as pd
```

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

pandas에서 활용 가능한 자료 구조

Series
1차원 배열 자료 구조

```
total_bill      16.99
tip              1.01
sex            Female
smoker             No
day               Sun
time           Dinner
size                2
Name: 0, dtype: object
```

DataFrame
2차원 자료 구조 (엑셀시트와 비슷)

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Mon	Dinner	3
2	21.01	3.50	Male	Yes	Mon	Lunch	3
3	23.68	3.31	Male	No	Thu	Lunch	2
4	24.59	3.61	Female	No	The	Dinner	4
5	25.29	4.71	Male	Yes	Wed	Dinner	4
6	8.77	2.00	Male	Yes	Wed	Dinner	2
7	26.88	3.12	Male	No	Sat	Dinner	4
8	15.04	1.96	Male	No	Sun	Dinner	2
9	14.78	3.23	Male	No	Sun	Dinner	2

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 은 어떻게 만들 수 있을까?

pd.DataFrame(data) 메소드로 DataFrame을 직접 선언 할 수 있다.

```
In [10]: df2 = pd.DataFrame({ 'A' : 1.,
    ....:                     'B' : pd.Timestamp('20130102'),
    ....:                     'C' : pd.Series(1,index=list(range(4)),dtype='float32
    ....:                     'D' : np.array([3] * 4,dtype='int32'),
    ....:                     'E' : pd.Categorical(["test","train","test","train"])
    ....:                     'F' : 'foo' })
    ....:

In [11]: df2
Out[11]:
   A          B  C  D      E    F
0  1 2013-01-02  1  3   test  foo
1  1 2013-01-02  1  3  train  foo
2  1 2013-01-02  1  3   test  foo
3  1 2013-01-02  1  3  train  foo
```

그 밖에도,

- DataFrame.from_dict
- DataFrame.from_items
- DataFrame.from_records

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 은 어떻게 만들 수 있을까?

Excel 혹은 csv 파일을 읽어 DataFrame을 생성할 수 있다

The default for *read_csv* is to create a DataFrame with simple numbered rows:

```
In [2]: pd.read_csv('foo.csv')
Out[2]:
       date  A  B  C
0  20090101  a  1  2
1  20090102  b  3  4
2  20090103  c  4  5
```

In the case of indexed data, you can pass the column number or column name you wish to use as the index:

```
In [3]: pd.read_csv('foo.csv', index_col=0)
Out[3]:
          A  B  C
date
20090101  a  1  2
20090102  b  3  4
20090103  c  4  5
```

read_csv 말고도,

- read_excel
- read_pickle
- read_json
- read_html

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 직접 생성해 볼까요?

파이썬 Dictionary 자료형을 이용하여 DataFrame 객체를 생성하는 방법은 다음과 같다.

```
#Pandas DataFrame 생성방법
import pandas
data = {
    'col1': [1, 2, 3, 4],
    'col2': [5, 6, 7, 8],
    'col3': [9, 10, 11, 12],
}
df = pandas.DataFrame(data, index=['A', 'B', 'C', 'D'])
df
```

Last executed 2017-02-10 15:51:56 in 14ms

	col1	col2	col3
A	1	5	9
B	2	6	10
C	3	7	11
D	4	8	12

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED

DataFrame 구조

이렇게 생성된 DataFrame 객체의 내부 구조는 다음과 같다.
'col1', 'col2', 'col3'라는 세 개의 Series 객체로 구성된다.

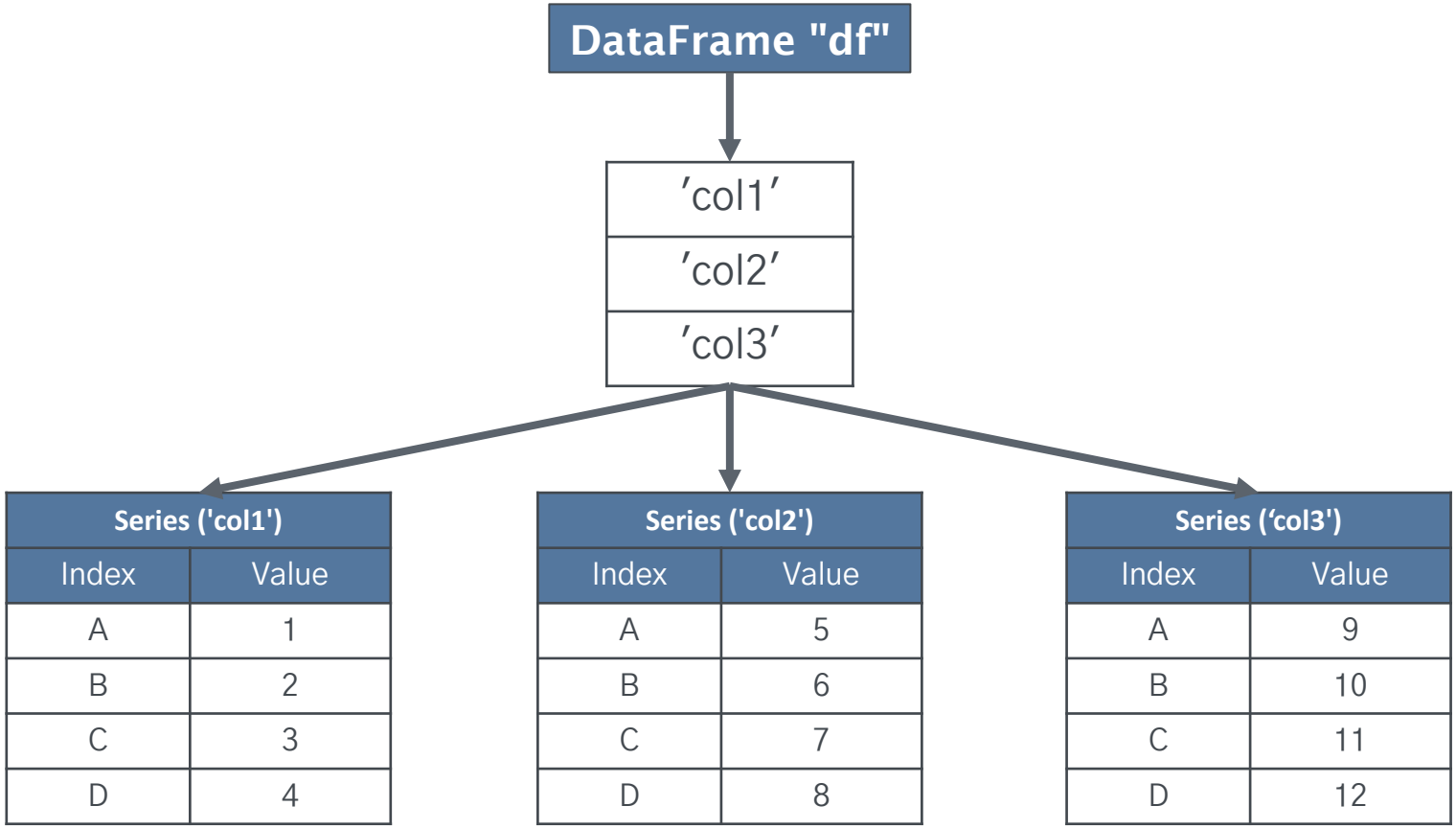
DataFrame "df"			
	Series ('col1')	Series('col2')	Series('col3')
Index	Value	Value	Value
A	1	5	9
B	2	6	10
C	3	7	11
D	4	8	12

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED

DataFrame 접근

DataFrame 객체는 세 개의 Series객체로 구성되어 있으므로

'col1', 'col2', 'col3'라는 key를 통해 value에 해당하는 Series객체에 접근할 수 있다.



Series ('col1')	
Index	Value
A	1
B	2
C	3
D	4

Series ('col2')	
Index	Value
A	5
B	6
C	7
D	8

Series ('col3')	
Index	Value
A	9
B	10
C	11
D	12

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 인덱싱 – column

DataFrame 의 특정 컬럼의 값은 Dictionary와 비슷하게 "df[colum명]" 으로 가져올 수 있다.

	col1	col2	col3
A	1	5	9
B	2	6	10
C	3	7	11
D	4	8	12

```
print(type(df['col1']))
df['col1']
```

executed in 5ms, finished 14:12:11 2022-10-06

```
<class 'pandas.core.series.Series'>

A    1
B    2
C    3
D    4
Name: col1, dtype: int64
```

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 인덱싱 – row

DataFrame의 특정 row의 값을 가져오기 위해서는 .iloc, loc 속성을 사용할 수 있다.

속성	설명
.iloc	integer position을 통해 행의 값을 찾을 수 있다.
.loc	label을 통해 행의 값을 찾을 수 있다.

	col1	col2	col3
A	1	5	9
B	2	6	10
C	3	7	11
D	4	8	12

```
df.iloc[0]
df.loc['A']
```

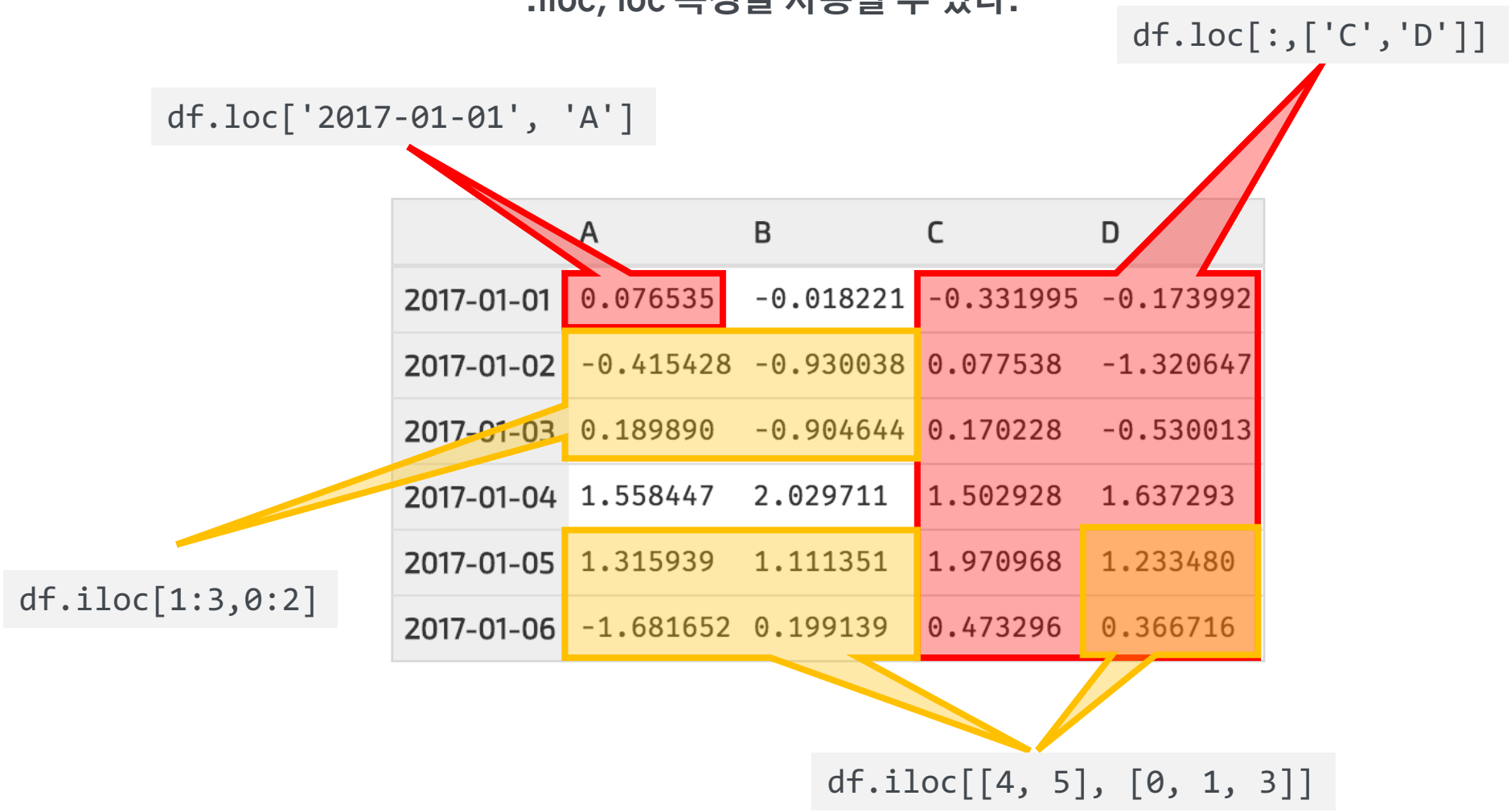
executed in 5ms, finished 14:16:10 2022-10-06

```
col1    1
col2    5
col3    9
Name: A, dtype: int64
```

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 인덱싱 – 특정 영역 가져오기

DataFrame의 특정 영역의 값을 가져오기 위해서는
.iloc, loc 속성을 사용할 수 있다.



	A	B	C	D
2017-01-01	0.076535	-0.018221	-0.331995	-0.173992
2017-01-02	-0.415428	-0.930038	0.077538	-1.320647
2017-01-03	0.189890	-0.904644	0.170228	-0.530013
2017-01-04	1.558447	2.029711	1.502928	1.637293
2017-01-05	1.315939	1.111351	1.970968	1.233480
2017-01-06	-1.681652	0.199139	0.473296	0.366716

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 인덱싱 – Boolean indexing

DataFrame에 필터 조건을 주면, 조건에 부합하는 값들을 인덱싱 할 수 있다.

df 전체 기준 > 2

```
df[df > 2]
```

executed in 11ms, finished 15:13:24 2022-10-06

	col1	col2	col3
A	NaN	5	9
B	NaN	6	10
C	3.0	7	11
D	4.0	8	12

df ['col1']기준 > 2

```
df[df['col1'] > 2]
```

executed in 6ms, finished 15:13:00 2022-10-06

	col1	col2	col3
C	3	7	11
D	4	8	12

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 핸들링 – 행, 열 생성

행, 열에 해당하는 값들을 선언해 줌으로써 DataFrame 내 행, 열을 생성할 수 있다.

‘E’ 행 생성

```
df.loc['E']=[2, 4, 6, 8]
df
```

executed in 9ms, finished 23:05:21 2022-10-06

	col1	col2	col3	col4
A	1	5	9	1
B	2	6	10	3
C	3	7	11	5
D	4	8	12	7
E	2	4	6	8

‘col4’열 생성

```
df['col4'] = [1, 3, 5, 7]
df
```

executed in 7ms, finished 23:06:54 2022-1(

	col1	col2	col3	col4
A	1	5	9	1
B	2	6	10	3
C	3	7	11	5
D	4	8	12	7

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 핸들링 – 행, 열 삭제

drop() 메소드를 활용하여 행과 열을 삭제 할 수 있다.

‘A’ 행 삭제

```
df.drop('A', axis=0)
```

executed in 9ms, finished 15:39:21 2022-10-06

	col1	col2	col3
B	2	6	10
C	3	7	11
D	4	8	12

‘col1’열 삭제

```
df.drop('col1', axis=1)
```

executed in 7ms, finished 15:39:38 2022-10-06

	col2	col3
A	5	9
B	6	10
C	7	11
D	8	12

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 핸들링 – 데이터 프레임 통합

서로 다른 데이터 프레임을 통합하려면 어떻게 해야 할까?

df1

	col1	col2	col3
A	1	5	9
B	2	6	10
C	3	7	11
D	4	8	12

+

df2

	col1	col4
A	1	사과
B	2	배
C	3	감
D	5	딸기

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 핸들링 – 데이터 프레임 통합 (concat, merge)

Concat()과 merge() 메소드를 활용하여 데이터 프레임을 통합할 수 있다.

concat()

데이터프레임을 연결

```
pd.concat([df1,  df2], axis=1)
```

executed in 8ms, finished 18:26:02 2022-10-06

	col1	col2	col3	col1	col4
A	1	5	9	1	사과
B	2	6	10	2	배
C	3	7	11	3	감
D	4	8	12	5	딸기

merge()

두 데이터프레임에 존재하는
고유값을 기준으로 병합

```
pd.merge(df1, df2, on='col1', how='outer')
```

executed in 11ms, finished 18:28:41 2022-10-06

	col1	col2	col3	col4
0	1	5.0	9.0	사과
1	2	6.0	10.0	배
2	3	7.0	11.0	감
3	4	8.0	12.0	NaN
4	5	NaN	NaN	딸기

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 메소드 – 요약, 통계

Pandas는 DataFrame을 다룰 수 있는 다양한 메소드들을 제공한다.

.head()	상위 5개의 행 출력	**.describe()**	데이터의 컬럼별 요약 통계량 확인 (count, mean, std, min, max, IQR)
.tail()	하위 5개의 행 출력	**.min(),** **.max(),** **.mean(),** **.median(),** **...**	데이터의 통계량 계산
.shape	데이터 차원 출력 (행, 열 크기 확인)		
.columns	데이터의 컬럼명 확인	**.value_counts()**	데이터의 개별 컬럼 내 값의 개수
.index	데이터의 인덱스 확인	**.groupby()**	group 별로 집계하는 함수 mean(), sum(), count() 집계 가능
.info()	데이터의 전반적인 정보 확인 (shape, columns, count, dtype)	**.corr()**	데이터의 컬럼 간의 상관 계수 확인

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 메소드 – missing, duplicated

Pandas는 DataFrame을 다룰 수 있는 다양한 메소드들을 제공한다.

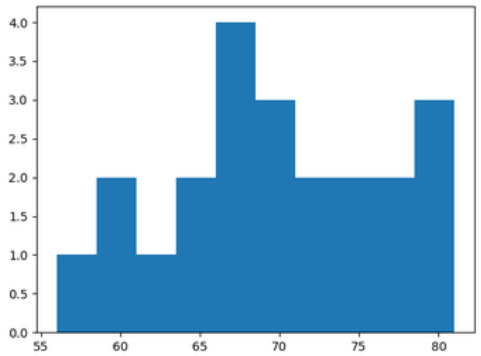
.isna(), .isnull()	결측인 컬럼을 확인하여 bool 형식으로 반환 (결측값 : True, 정상값 : False)	**.duplicated()**	중복되는 값이 있는 컬럼을 확인하여 bool 형식으로 변환
.notna() .notnull()	결측이 아닌 컬럼을 확인하여 bool 형식으로 반환 (정상값 : True, 결측값 : False)	**.drop_duplicates()**	중복되는 값 삭제 (keep = first/last/false)
.dropna()	결측 값이 있는 행 삭제 (**axis=0**)	**.replace(A, B)**	A를 B로 대치하는 함수
.fillna()	결측 값을 원하는 값으로 변경		

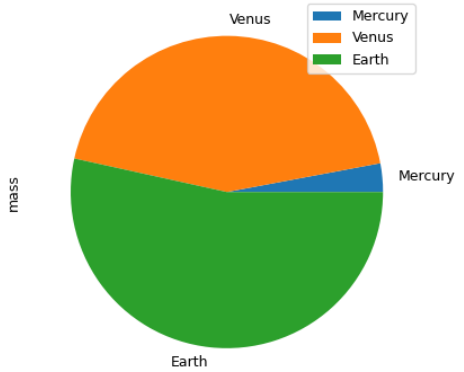
© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

DataFrame 메소드 – 시각화

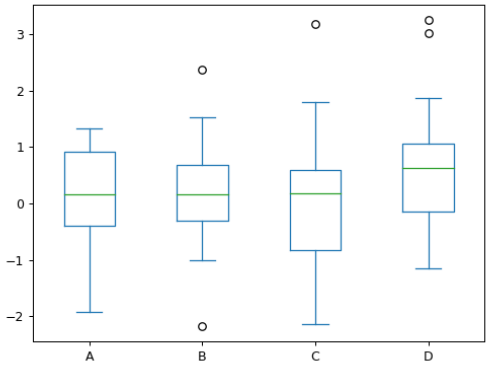
Pandas는 DataFrame에는 matplotlib을 기반으로한 시각화 메소드를 내장하고 있다.

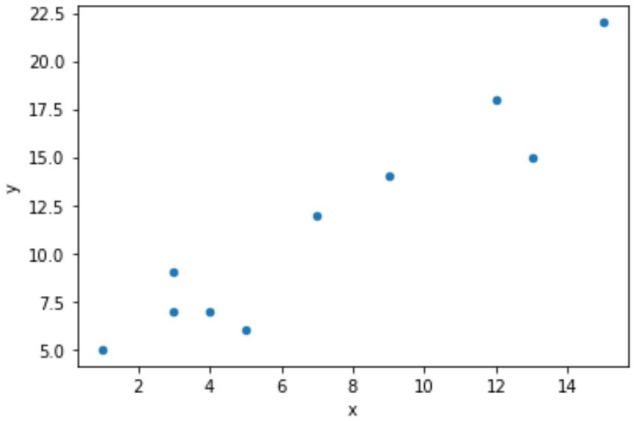
.plot.hist()	Histogram 시각화	.plot.pie()	Pie chart 시각화





.plot(kind='box')	Boxplot 시각화	.plot(kind='scatter')	Scatterplot 시각화





RIGHTS RESERVED.

03

matplotlib 배워봅시다!

matplotlib 이란?

Visualization with Python

시각화에 특화된 파이썬 라이브러리

https://matplotlib.org/stable/

latest version 3.6.0

```
$ conda install matplotlib

$ pip install matplotlib

>>> import matplotlib.pyplot as plt
```

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

직접 plot을 그려 볼까요?

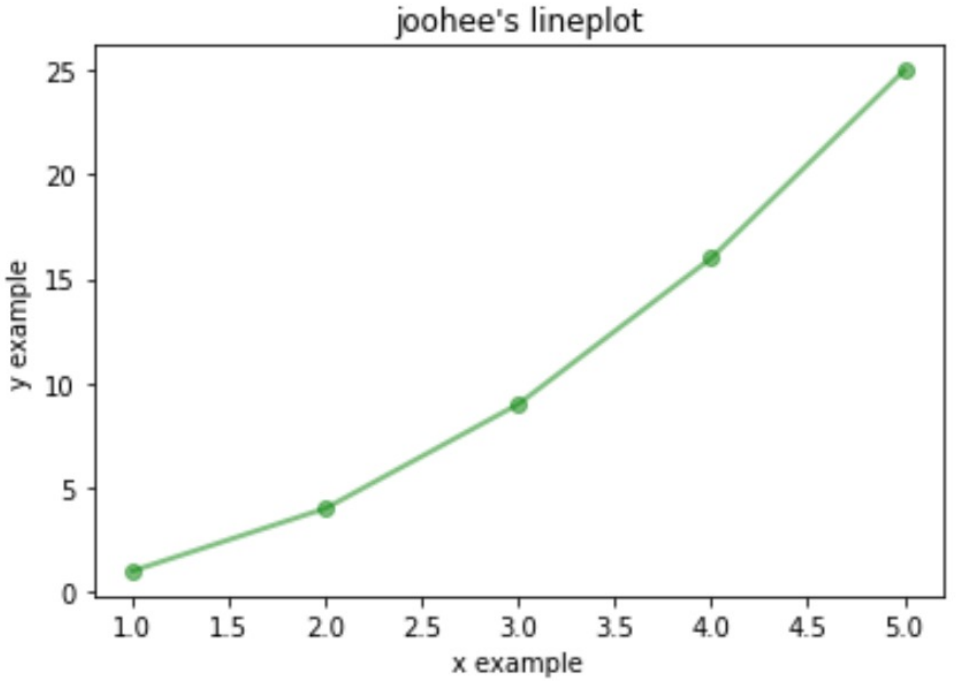
파이썬 matplotlib을 활용하여 lineplot을 그리는 방법은 다음과 같다.

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]
y = [1, 4, 9, 16, 25]
plt.plot(x, y, color='green', marker='o', alpha=0.5, linewidth=2)

plt.title("joohee's lineplot")
plt.xlabel("x example")
plt.ylabel("y example")
plt.show()
```

executed in 119ms, finished 23:44:45 2022-10-06



© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

Plot을 그리기에 앞서,

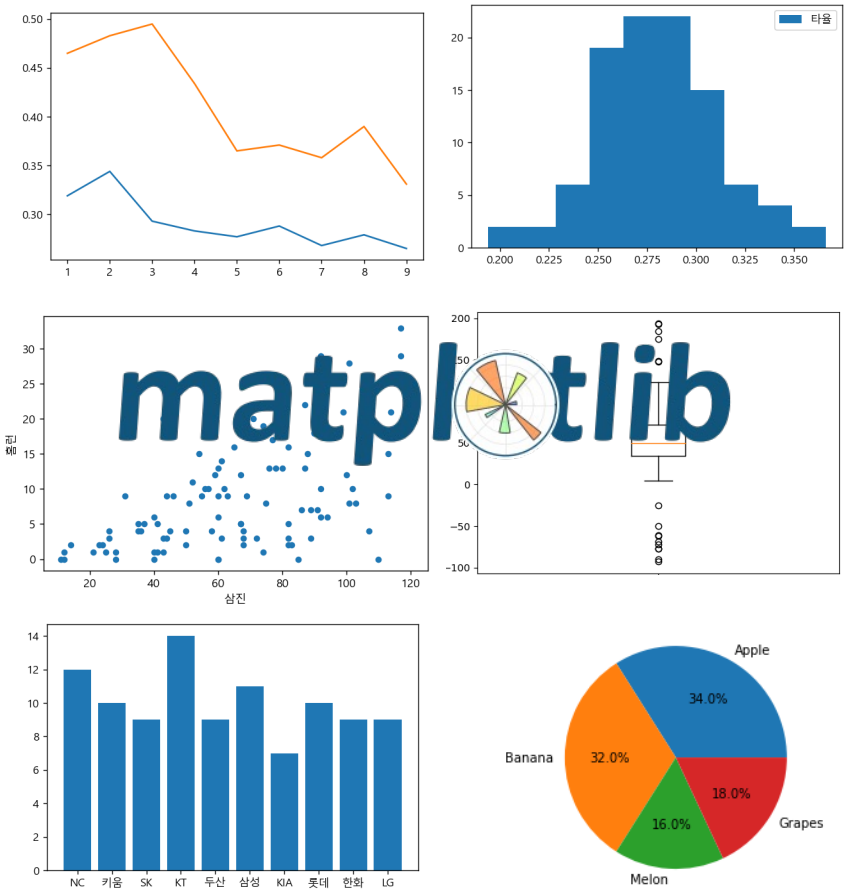
Plot을 그리기에 앞서, 생각해야 할 것

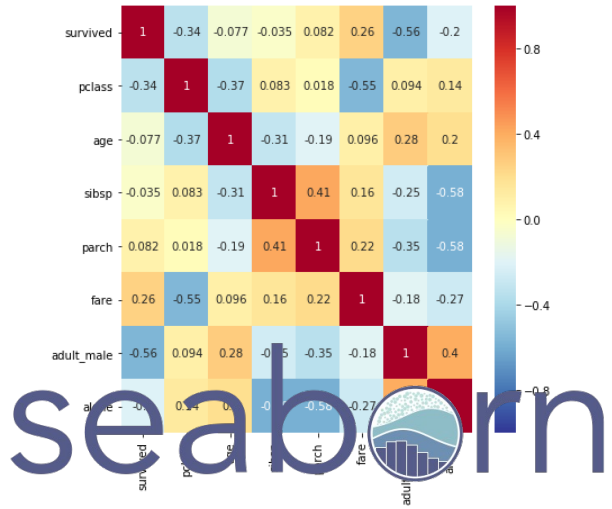
1. 어떤 종류의 plot을 그릴 것인지?
2. 어떤 구조로 그릴 것인지?
3. 옵션은 어떻게 할 것인지?

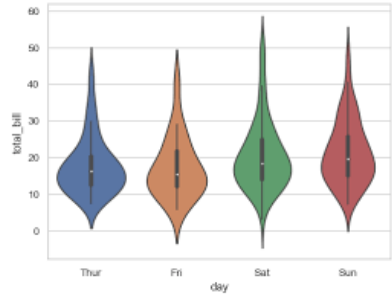
© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

plot의 종류

데이터나, 분석 방법에 따라 활용할 plot을 선정한다.







© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

plot의 구조

Plot을 그릴 때 데이터를 어떤 구조로 표현할 것인지를 선택한다.

1. 한 그래프 내에 여러 데이터를 그린다.

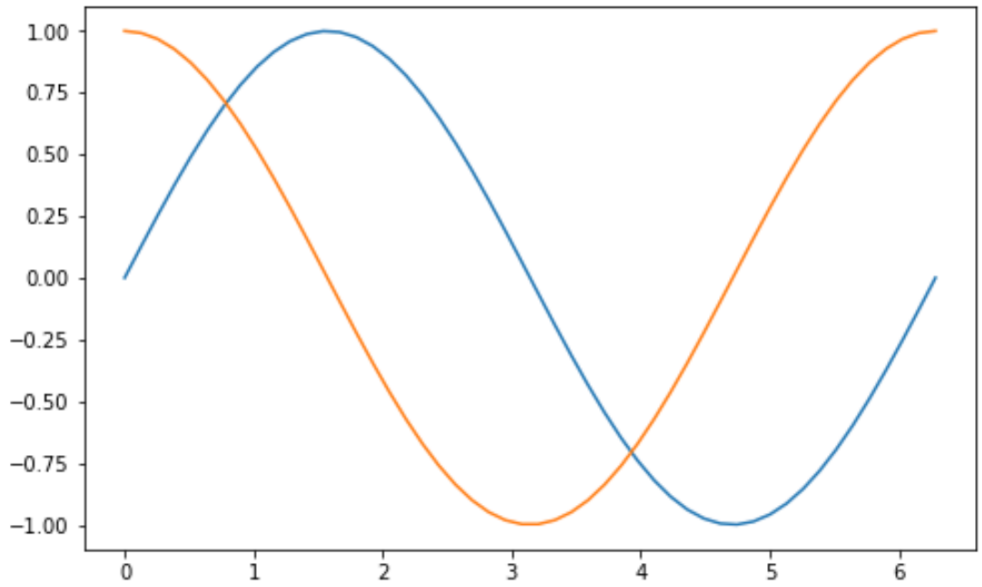
```
x = np.linspace(0, 2*np.pi, 50)

fig = plt.figure(figsize=(8, 5))

# plt.plot(x, np.sin(x), x, np.cos(x)) # 한 plot에 데이터를 추가해도 됨
plt.plot(x, np.sin(x))
plt.plot(x, np.cos(x))

plt.show()
```

executed in 133ms, finished 02:08:18 2022-10-07



© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

plot의 구조

Plot을 그릴 때 데이터를 어떤 구조로 표현할 것인지를 선택한다.

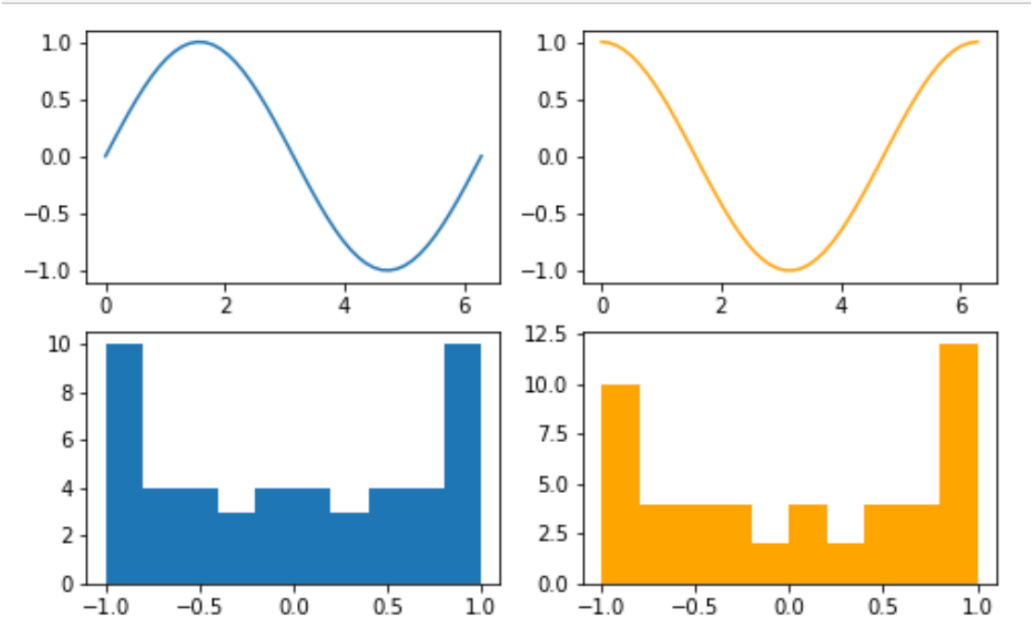
2. subplot을 활용한다.

```
fig = plt.figure(figsize=(8, 5))

ax1 = fig.add_subplot(2, 2, 1)
ax1.plot(x, np.sin(x))
ax2 = fig.add_subplot(2, 2, 2)
ax2.plot(x, np.cos(x), color='orange')
ax3 = fig.add_subplot(2, 2, 3)
ax3.hist(np.sin(x))
ax4 = fig.add_subplot(2, 2, 4)
ax4.hist(np.cos(x), color='orange')

plt.show()
```

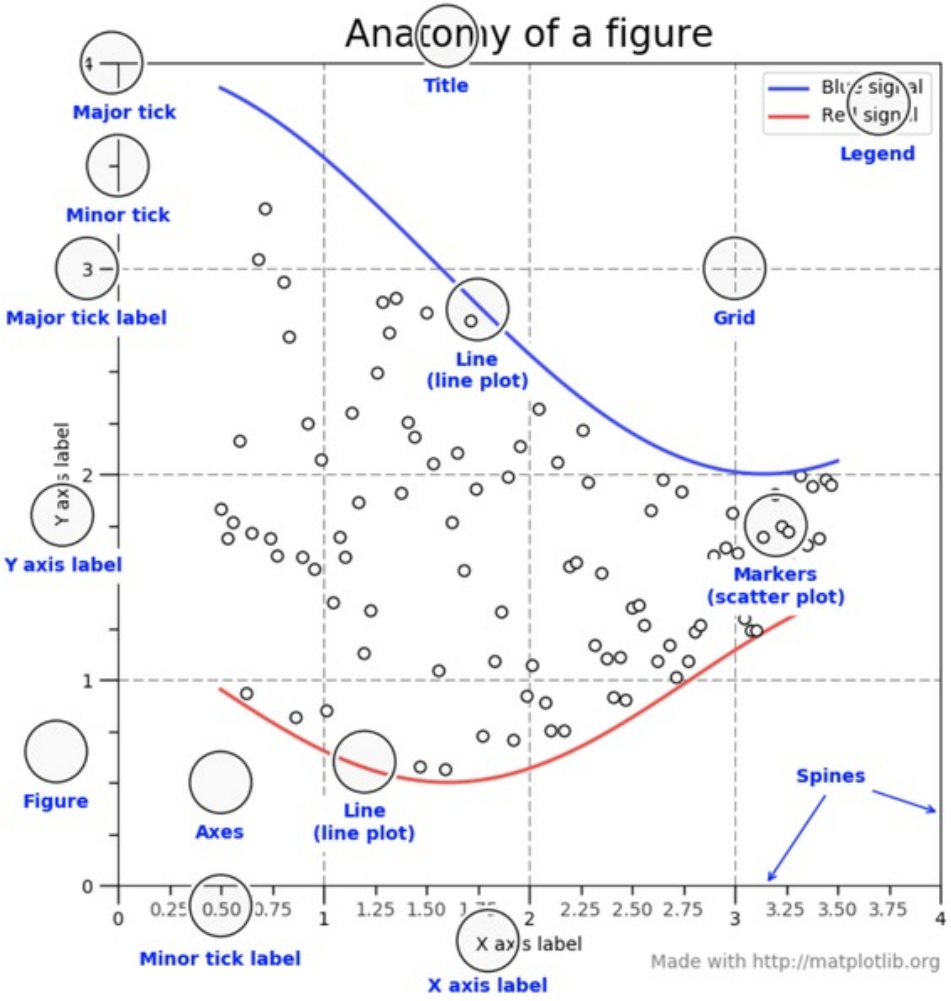
executed in 295ms, finished 03:06:11 2022-10-07



© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

plot의 옵션

Plot의 옵션을 설정한다. 아래의 명칭만 잘 알면 검색을 통해 쉽게 설정할 수 있다.



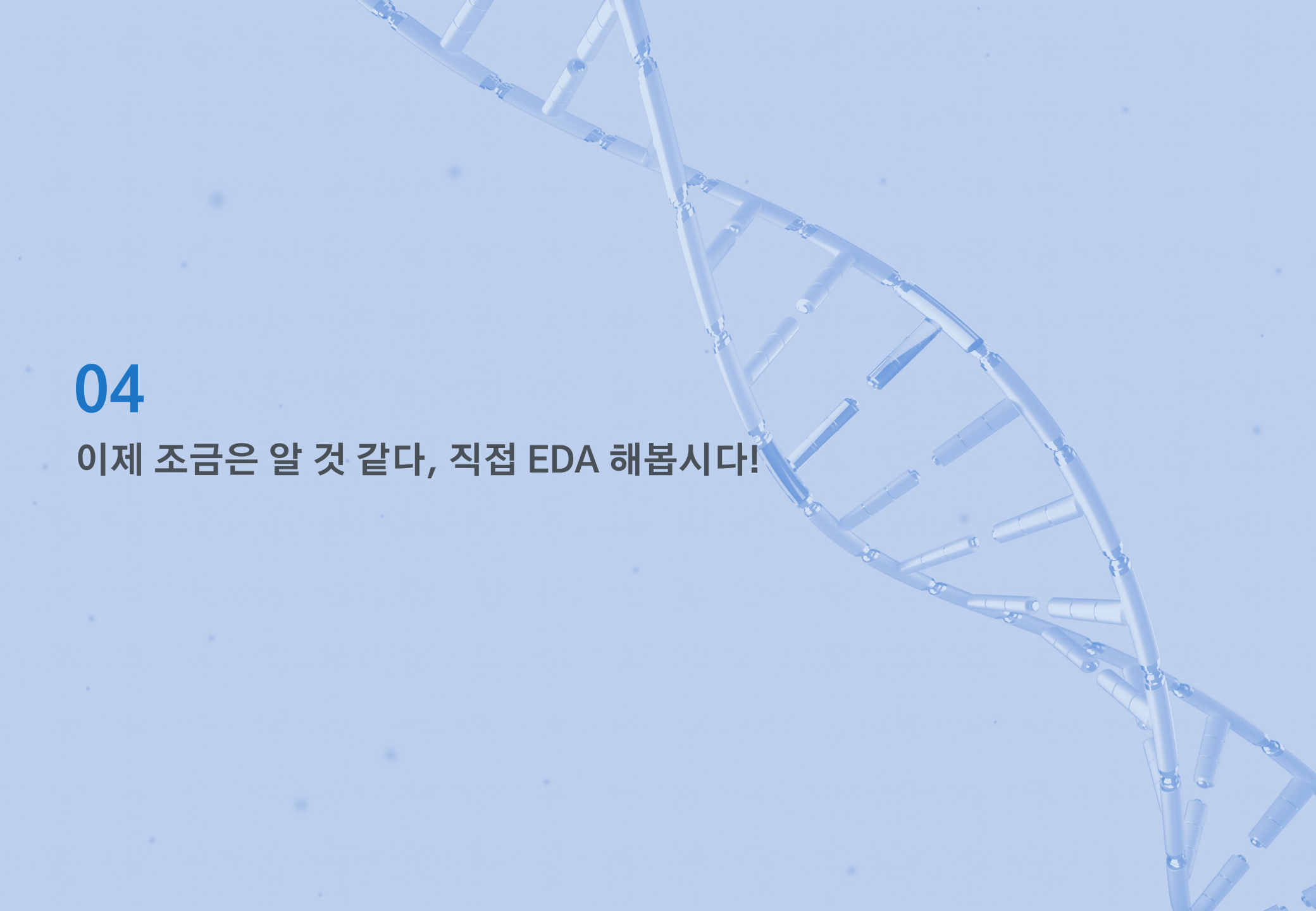
Google

- matplotlib **Legend** 위치 변경
- matplotlib **Title** 설정
- matplotlib scatter plot **markers** 변경
- seaborn heatmap **X axis tick** 간격 변경
- seaborn **Grid** 설정

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

04

이제 조금은 알 것 같다, 직접 EDA 해봅시다!



이제 조금은 알 것 같다, 직접 EDA 해봅시다!

HeartDisease_Dataset.csv

From UCI Machine Learning Repository

303 rows x 14 columns

index	변수명	한글 변수명	변수 설명
1	age	나이	range : 29~77
2	sex	성별	M = male; F = female
3	cp	가슴통증	0 = typical angina; 1 = atypical angina; 2 = non-anginal pain; 3 = asymptomatic
4	trestbps	혈압	In mm Hg / range : 94~200
5	chol	콜레스테롤	in mg/dl / range : 126~564
6	fbs	공복 혈당	1 = true; 0 = false (fasting blood sugar > 120 mg/dl)
7	restecg	심전도	0 = normal; 1 = having ST-T wave abnormality; 2 = showing probable or definite left ventricular hypertrophy by Estes' criteria
8	thalach	최대 심장박동 수	range : 71~202
9	exang	운동 유발 협심증	1 = yes; 0 = no
10	oldpeak	ST 우울증	range : 0~6.2
11	slope	ST segment 기울기	0 = upsloping; 1 = flat; 2 = downsloping
12	ca	혈관 수	0, 1, 2, 3, 4
13	thal	빈혈 여부	3 = normal; 6 = fixed defect; 7 = reversable defect
14	target	심장질환 여부	0 = normal; 1 = heartdisease

데이터 출처 :
https://archive.ics.uci.edu/ml/datasets/heart+disease

© 2022 INSILICOGEN, INC. ALL RIGHTS RESERVED.

Thank you!

Contact Info.

OFFICE 경기도 용인시 기흥구 흥덕1로 13 흥덕IT밸리 타워 A동 2901-2903호
EMAIL info@insilicogen.com
PHONE # 031 278 0061
FAX 031 278 0062