강원지역혁신플랫폼

기계학습

Machine Learning

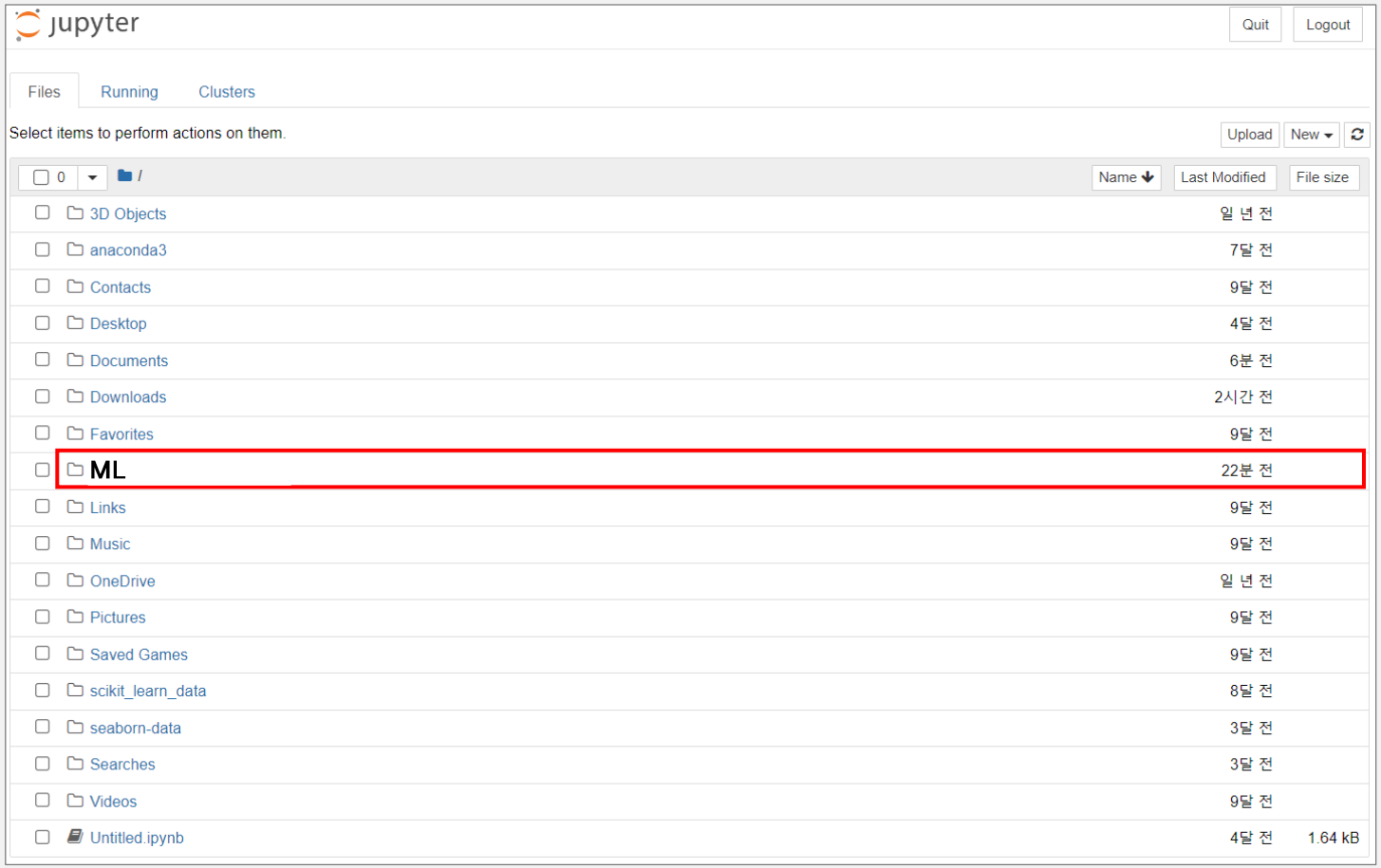
비선형 변환 변수

학습목표

- 비선형 변환 변수의 개념을 이해하고 구현할 수 있습니다.

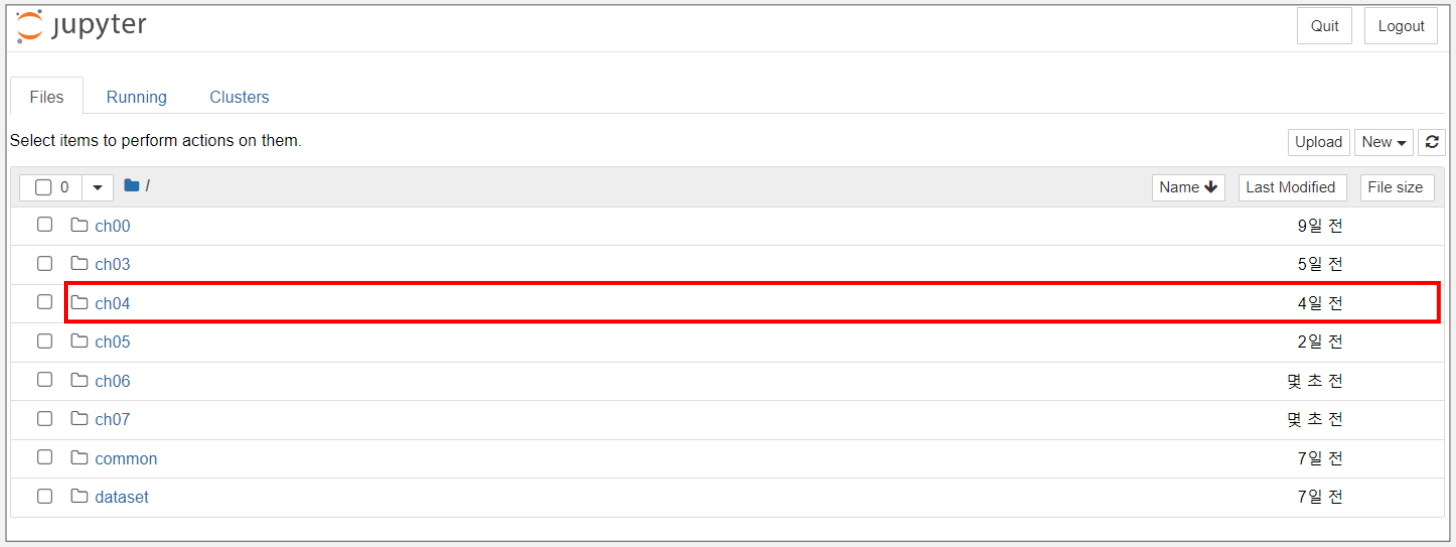
01 | ML 폴더

ML 폴더를 클릭하기



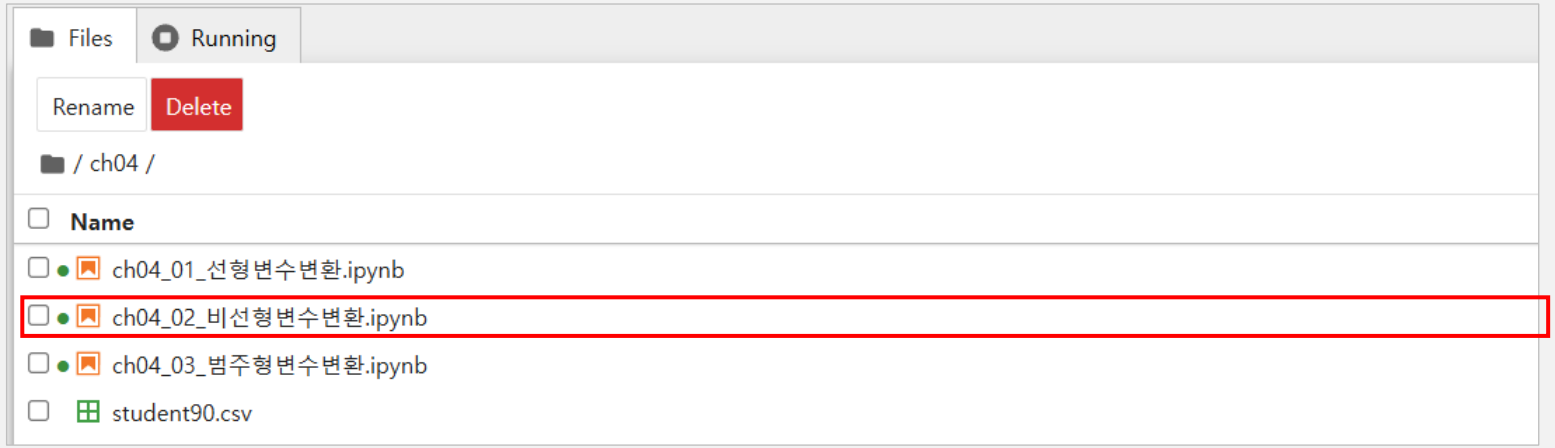
02 | ch04 폴더

ch04 폴더 클릭하기



03 | ch04_02_비선형변수변환.ipynb

✦ ch04_02_비선형변수변환.ipynb 파일 클릭하기



04 | 비선형 변수 변환

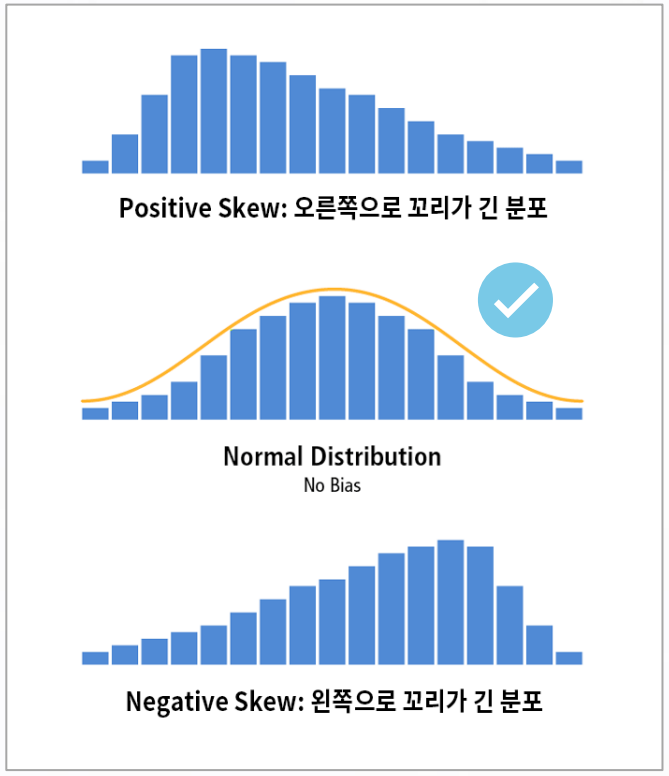
수치형 변수 비선형 변환: 함수 변환

- 로그 변환 (Log Transformation)
- 거듭제곱 변환 (Power Transformation)
- 루트 변환 (Square Root Transformation)
- 역수 변환 (Inverse Transformation)
- 지수 변환 (Exponential Transformation)
- 정규화 (Normalizing – L1, L2, Max)

04 | 비선형 변수 변환

비선형 변수 변환(함수 변환)의 목적

- 함수 변환의 **목적**은 **분포의 좌우 비대칭성**을 **정규분포**에 **가까운 모양으로 변환**하는데 있음
  - 아래 그림처럼 정규분포와 비슷하게 데이터를 변환하는 과정이 중요한 부분임



한쪽으로 치우친 데이터
▶ 정규분포 모양으로 변환

출처: https://www.esri.com/news/arcuser/0111/findmodel.html

04 | 비선형 변수 변환

함수 변환(지수/로그/제곱/루트/역수)을 살펴보면, 아래와 같이 간단하게 다음처럼 숫자 계산이 됨을 알 수 있음

✦ 아래의 표는 로그/루트/역수/제곱 함수의 숫자 계산을 나타낸 것임

함수	X	Y	결과
로그	10	np.log10(10)=1.0	• X가 10에서 100만큼 변하면, Y는 1에서 2만큼 변함 • 즉, X에서 90차이 나던 숫자가 Y에서는 1차이가 남 • 이것은 양의 왜도(Positive Skew)의 오른쪽 긴 부분을 줄여주는 역할
	100	np.log10(100)=2.0	
루트	10	math.sqrt(10)=3.1622	• X가 10에서 100만큼 변하면, Y는 3.xx에서 10만큼 변함 • X에서 90차이 나던 숫자가 Y에서는 6정도 차이가 남 • 이것은 양의 왜도(Positive Skew)의 오른쪽 긴 부분을 줄여주는 역할
	100	math.sqrt(100)=10.0	
역수	10.0	np.reciprocal(10.0)=0.1	• X가 10에서 100만큼 변하면, Y는 0.1에서 0.01만큼 변함 • X가 90차이 나던 숫자가 Y에서는 0.09정도 차이가 남 • 이것은 양의 왜도(Positive Skew)의 오른쪽 긴 부분을 줄여주는 역할
	100.0	np.reciprocal(100.0)=0.01	
제곱	10의 2승	10**2=100	• X가 10에서 100만큼 변하면, Y는 100에서 10000만큼 변함 • X에서 90차이 나던 숫자가 Y에서는 9900정도 차이가 남 • 이것은 음의 왜도(Negative Skew)의 긴 부분을 극단적으로 줄여주는 역할
	10의 4승	10**4=10000	

04 | 비선형 변수 변환

아래의 표는 함수 변환(지수/로그/제곱/루트/역수)이
양의 왜도(Positive Skew)와 음의 왜도(Negative Skew)를 줄여주는 것을 알 수 있음

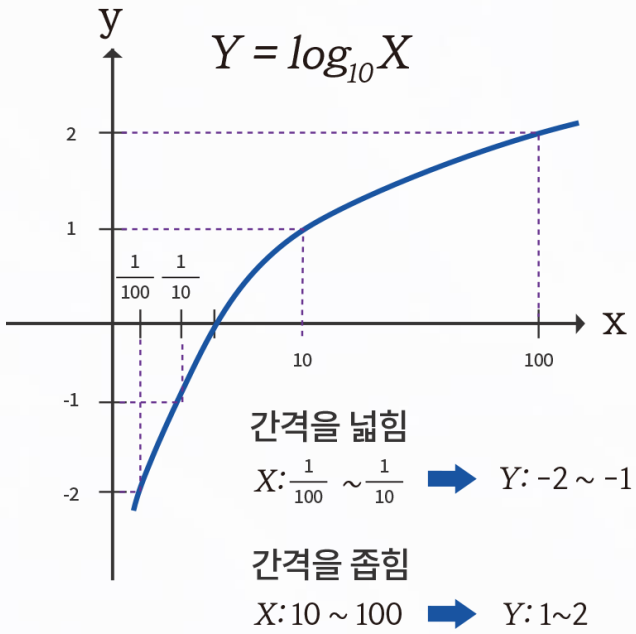
- 양의 왜도를 줄여주는 변환: 로그/루트/역수
  - 큰 숫자들을 줄여줌
- 음의 왜도를 줄여주는 변환: 제곱/지수
  - 작은 숫자들의 크기를 늘려줌

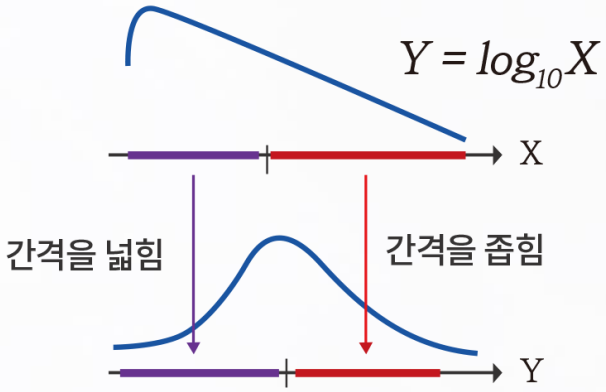
함수	X	Y	결과	
로그	10	np.log10(10)=1.0	• X가 10에서 100만큼 변하면, Y는 1에서 2만큼 변함 • 즉, X에서 90차이 나던 숫자가 Y에서는 1차이가 남 • 이것은 양의 왜도(Positive Skew)의 오른쪽 긴 부분을 줄여주는 역할	양의 왜도를 줄여주는 변환
	100	np.log10(100)=2.0		
루트	10	math.sqrt(10)=3.1622	• X가 10에서 100만큼 변하면, Y는 3.xx에서 10만큼 변함 • X에서 90차이 나던 숫자가 Y에서는 6정도 차이가 남 • 이것은 양의 왜도(Positive Skew)의 오른쪽 긴 부분을 줄여주는 역할	
	100	math.sqrt(100)=10.0		
역수	10.0	np.reciprocal(10.0)=0.1	• X가 10에서 100만큼 변하면, Y는 0.1에서 0.01만큼 변함 • X가 90차이 나던 숫자가 Y에서는 0.09정도 차이가 남 • 이것은 양의 왜도(Positive Skew)의 오른쪽 긴 부분을 줄여주는 역할	
	100.0	np.reciprocal(100.0)=0.01		
제곱	10의 2승	10**2=100	• X가 10에서 100만큼 변하면, Y는 100에서 10000만큼 변함 • X에서 90차이 나던 숫자가 Y에서는 9900정도 차이가 남 • 이것은 음의 왜도(Negative Skew)의 긴 부분을 극단적으로 줄여주는 역할	음의 왜도를 줄여주는 변환
	10의 4승	10****4=10000		

05 | 비선형 변수 변환: 로그 변환

로그 변환(Log Transformation)

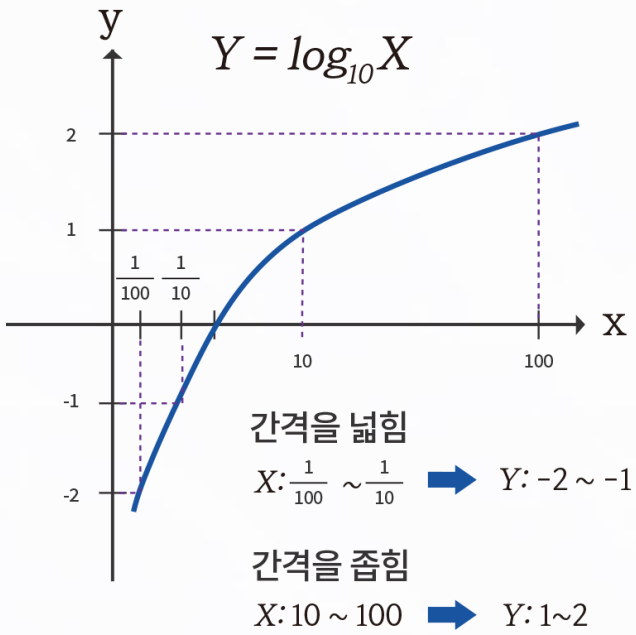
- 로그함수는 큰 숫자의 크기를 작게 좁혀주는 역할을 함
  - 로그함수는 지수함수의 역함수임
    - 아래 그림은 X에서 Y의 로그 함수 변환의 첨도(Skew)가 어떻게 변하는지 이해할 수 있음

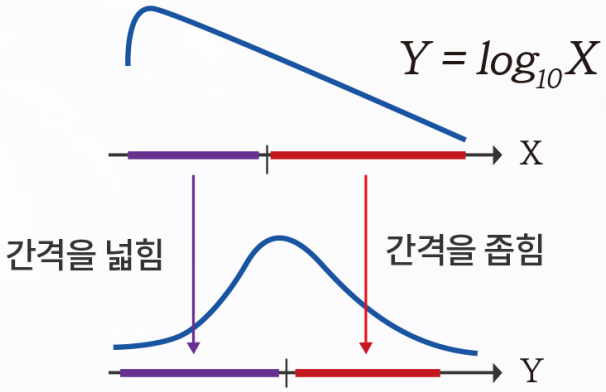




05 | 비선형 변수 변환: 로그 변환

- 아래 그림처럼 X가 10에서 90만큼 늘어날 때 로그를 붙이면 1에서 2로 변하게 됨
  - 이것을 데이터 분포 측면에서 보면, 양의 첨도(Positive Skew) 부분을 늘려주는 역할을 함
    - 로그에서 1보다 작은 숫자들은 다시 크기를 늘려주는 역할을 함
    - 따라서, 전체적으로 양의 첨도가 좌우 대칭처럼 변환할 수 있는 함수 변환이 될 수 있음





05 | 비선형 변수 변환: 로그 변환

- 다음은 대학생 90명의 키(cm)와 몸무게(kg) 데이터셋으로 로그 변환을 수행함
  - 아래의 표는 데이터셋의 데이터 구조임
    - 대학생 데이터셋의 관측치는 90개, 속성은 4개로 구성됨

no	sex	weight_kg	height_cm
1	M	98	198
2	F	77	170
3	M	70	170
4	M	90	198
5	F	71	170

05 | 비선형 변수 변환: 로그 변환

다음은 대학생 90명의 데이터 세트를 읽어오는 코드이다.

- 실행결과 데이터의 형상이 (90, 4)인 것을 알 수 있음
  - 즉, 관측치의 데이터는 90개, 속성은 4개인 것을 알 수 있음

```
# 데이터 읽어오기
sample_data = pd.read_csv(os.getcwd()+'/student90.csv')
print(sample_data.shape)   # (90, 4)
sample_data[:10]           # 10행 출력
```

```
(90, 4)
```

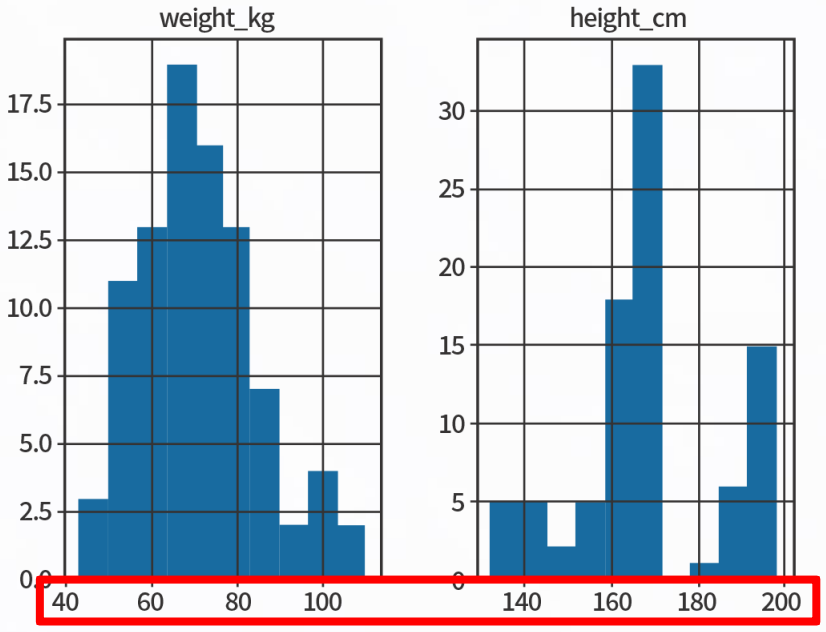
	no	sex	weight_kg	height_cm
0	1	m	98	198
1	2	m	77	170
2	3	m	70	170
3	4	m	90	198
4	5	m	71	170
5	6	m	70	165
6	7	m	73	193
7	8	m	59	142
8	9	m	68	137
9	10	m	86	155

05 | 비선형 변수 변환: 로그 변환

다음은 데이터 세트의 키와 몸무게 속성으로 히스토그램을 그리는 코드이다.

- 아래의 그림과 같이 두 개의 속성이 단위가 다른 것을 알 수 있음
  - 키의 단위 cm이고, 몸무게의 단위는 kg임

```
pd.DataFrame(sample_data, columns=['weight_kg','height_cm']).hist()
plt.subplots_adjust(hspace=1)plt.show()
```



05 | 비선형 변수 변환: 로그 변환

- 다음은 데이터 세트의 키와 몸무게 속성에 **로그 변환**을 수행함
  - 이 때 로그는 (1)**상용로고+1**, (2)**자연로그**, (3)**자연로그+1**, (4)**절대값을 씌우고 자연로그 변환 후 원래의 부호를 붙인** 경우를 **히스토그램**으로 **비교**함
    - 이 중에 일반적으로 파이썬으로 **로그 변환**에 쓰이는 방법은 **자연로그+1**을 사용함
      - 자연로그+1의 파이썬 함수는 np.log1p()을 사용함

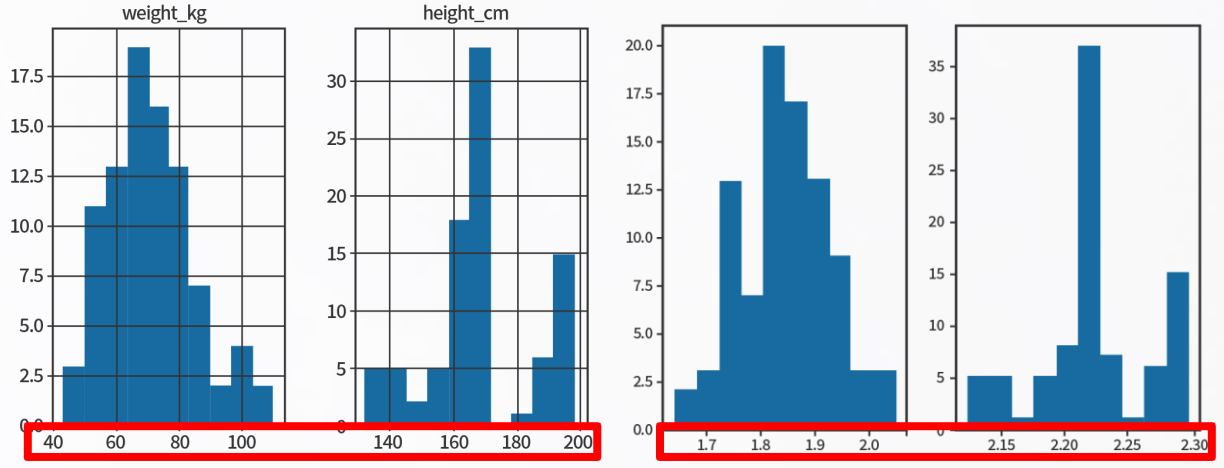
05 | 비선형 변수 변환: 로그 변환

다음은 상용로그+1의 로그 함수로 로그 변환을 수행하는 코드이다.

✦ 아래 그림과 같이 변경 후 히스토그램이 좀 더 정규분포에 가까운 것을 볼 수 있음

```
# (1)상용로그+1 : 히스토그램
weight_log = np.log10(sample_data.filter(['weight_kg'])+1)
# 몸무게가 0인경우 음의 무한대가 되는 것을 방지하기 위해 +1
height_log = np.log10(sample_data.filter(['height_cm'])+1)
# 키가 0인경우 음의 무한대가 되는 것을 방지하기 위해 +1

fig, ax = plt.subplots(1,2)
ax[0].hist(pd.DataFrame(weight_log))
ax[1].hist(pd.DataFrame(height_log))
plt.subplots_adjust(hspace=1)
plt.show()
```



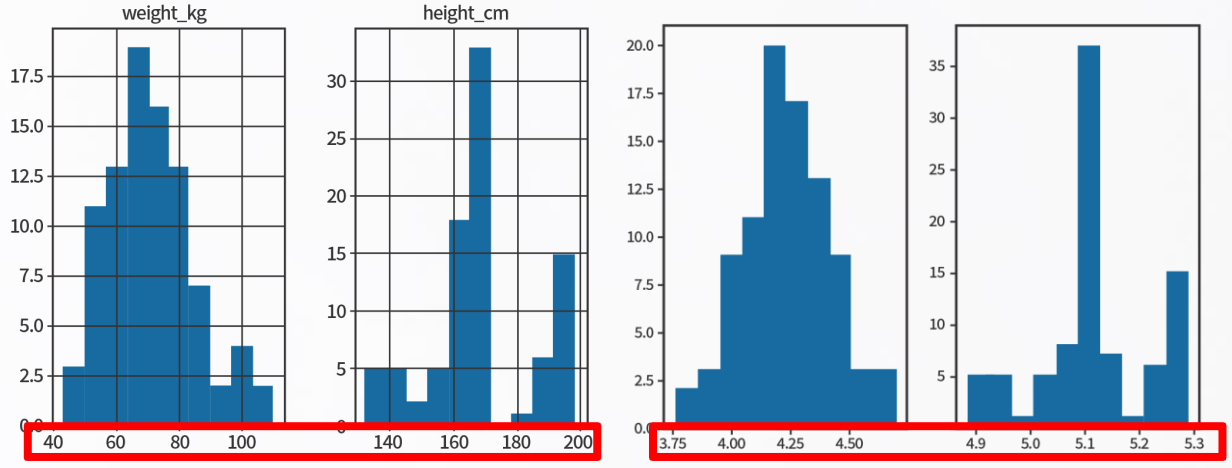
05 | 비선형 변수 변환: 로그 변환

다음은 자연로그의 로그 함수로 로그 변환을 수행하는 코드이다.

✦ 아래 그림과 같이 변경 후 히스토그램이 좀 더 정규분포에 가까운 것을 볼 수 있음

```
# (2)자연로그 : 히스토그램
weight_log = np.log(sample_data.filter(['weight_kg'])) # 자연로그
 height_log = np.log(sample_data.filter(['height_cm'])) # 자연로그

fig, ax = plt.subplots(1,2)
ax[0].hist(pd.DataFrame(weight_log))
ax[1].hist(pd.DataFrame(height_log))
plt.subplots_adjust(hspace=1)
plt.show()
```



변경 전 변경 후

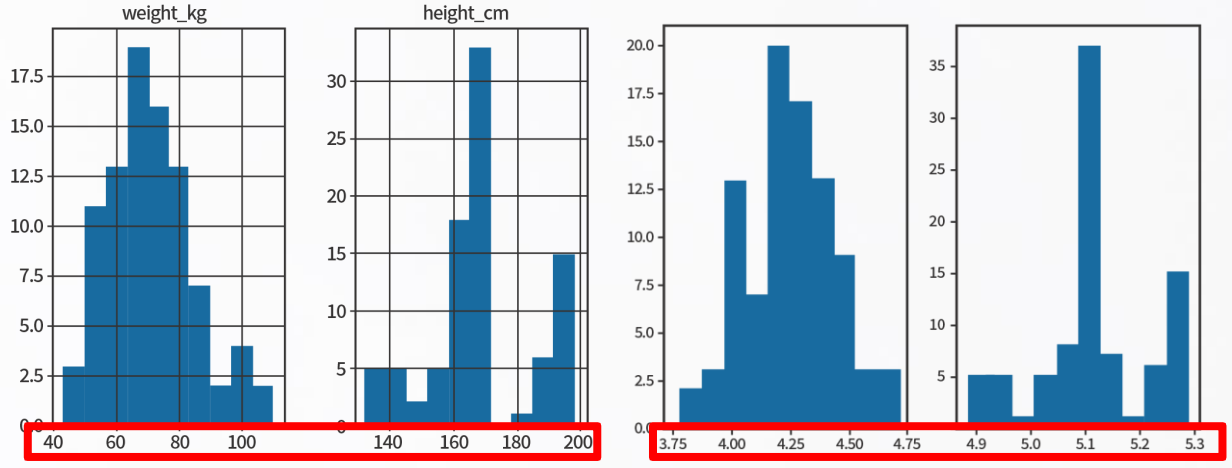
05 | 비선형 변수 변환: 로그 변환

다음은 자연로그+1의 로그 함수로 로그 변환을 수행하는 코드이다.

✦ 아래 그림과 같이 변경 후 히스토그램이 좀 더 정규분포에 가까운 것을 볼 수 있음

```
# (3)자연로그+1 : 히스토그램
weight_log = np.log1p(sample_data.filter(['weight_kg']))  # 자연로그+1
height_log = np.log1p(sample_data.filter(['height_cm']))  # 자연로그+1

fig, ax = plt.subplots(1,2)
ax[0].hist(pd.DataFrame(weight_log))
ax[1].hist(pd.DataFrame(height_log))
plt.subplots_adjust(hspace=1)
plt.show()
```



변경 전 변경 후

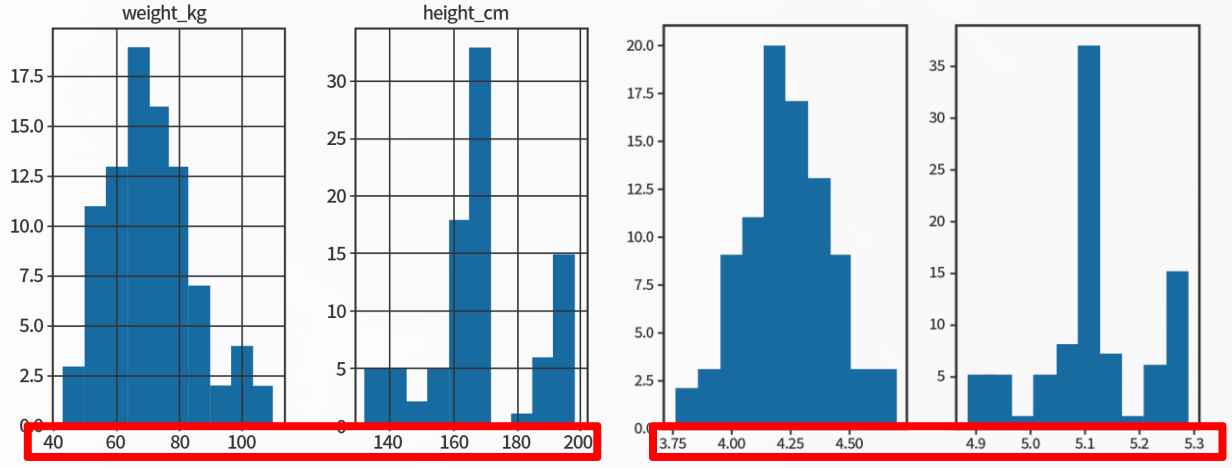
05 | 비선형 변수 변환: 로그 변환

다음은 절대값 자연로그에 원래 부호를 붙임의 로그 함수로 로그 변환을 수행하는 코드이다.

✦ 아래 그림과 같이 변경 후 히스토그램이 좀 더 정규분포에 가까운 것을 볼 수 있음

```
# (4)절대값 자연로그에 원래 부호를 붙임(마이너스인 경우도 사용 가능) : 히스토그램
weight_log = np.sign(sample_data.filter(['weight_kg'])) * np.log(np.abs(sample_data.filter(['weight_kg'])))
height_log = np.sign(sample_data.filter(['height_cm'])) * np.log(np.abs(sample_data.filter(['height_cm'])))

fig, ax = plt.subplots(1,2)
ax[0].hist(pd.DataFrame(weight_log))
ax[1].hist(pd.DataFrame(height_log))
plt.subplots_adjust(hspace=1)
plt.show()
```



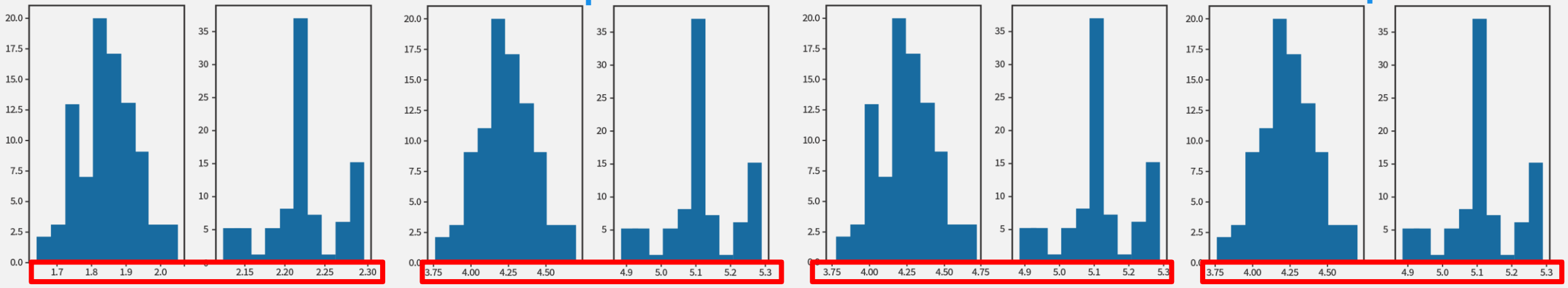
변경 전 변경 후

05 | 비선형 변수 변환: 로그 변환

다음은 로그 변환 후 히스토그램을 비교한 결과임

- 이 때 로그는 (1)상용로고+1, (2)자연로그, (3)자연로그+1, (4)절대값을 씌우고 자연로그 변환 후 원래의 부호를 붙인 경우임

(2)와 (4)의 분산이 비슷함



(1)과 (3)의 분산이 비슷함