Data Analysis and Preprocessing, Transform

2020.10

References:

- **KPC, DSAC(Data Scientist Academy & Certificate) Manual, 2019**
- **many internet sites**

데이터 형태

형식	내용
정형 (structured)	• 데이터의 포맷이 정해져 있는 데이터 • 서식이 정해진 데이터(엑셀의 표 등) • CSV(comma separated value) 파일 • JSON 파일
비정형 (unstructured)	• 미리 정해진 포맷을 가지지 않는 데이터 • 블로그, 트위터 데이터 등 임의의 문장 (Text) 등 • 오디오나 비디오 데이터
반정형 (semi-structured)	• 데이터 내부에는 논리적인 형식을 가지고 있으나 외형상으로는 데이터 포맷이 정형 데이터처럼 완전하게 정의되어 있지는 않은 데이터 • 센서 데이터, 웹 사용 기록 등 • HTML/XML

DATA Types

- **Numerical (수치형)**
  - Discrete data: numbers that are limited to integers
    - (ex) Numbers in dice game, True or False, Classes
  - Continuous data: numbers that are of infinite value
    - (ex) Height, Weight, Housing cost, etc.
- **Categorical (or nominal) (범주형)**
  - Values that can not be measured up against each other
  - (ex) colors, days of a week, months, blood types
- **Ordinal (순서형)**
  - Data are like categorical, but can be measured up against each other
  - (ex) school grades (A>B>C), size of clothes (XL>L>M>S)

Data Types

- **수치형(Numerical):** 숫자의 양이 어떤 의미를 가지는 데이터
  - 무게, 길이, 온도, 압력, 속도, 화폐 단위
  - 덧셈과 뺄셈, 순서, 평균 등의 결과가 의미를 갖는다.
- **범주형(Categorical):** 클래스를 구분하는 데이터
  - 성별, 국가명, 요일, 사람 이름 등은 범주형 데이터이다.
  - 범주형은 대부분 문자로 표현되지만 편의상 숫자로 대체하여 표현하기도 한다. 예를 들어 월요일=1, 화요일=2, 수요일=3 등
- **순서형(Ordinal)**: 순서가 의미를 가지는 데이터
  - 여성의 옷 사이즈를 나타내는 44, 55, 66 같은 숫자, 달력의 1일, 2일, 3일 등이 순서형 데이터이다.
  - 순서형 데이터에서는 덧셈이나 뺄셈이 아무런 의미가 없다.

데이터 탐색과 시각화

탐색적 데이터 분석

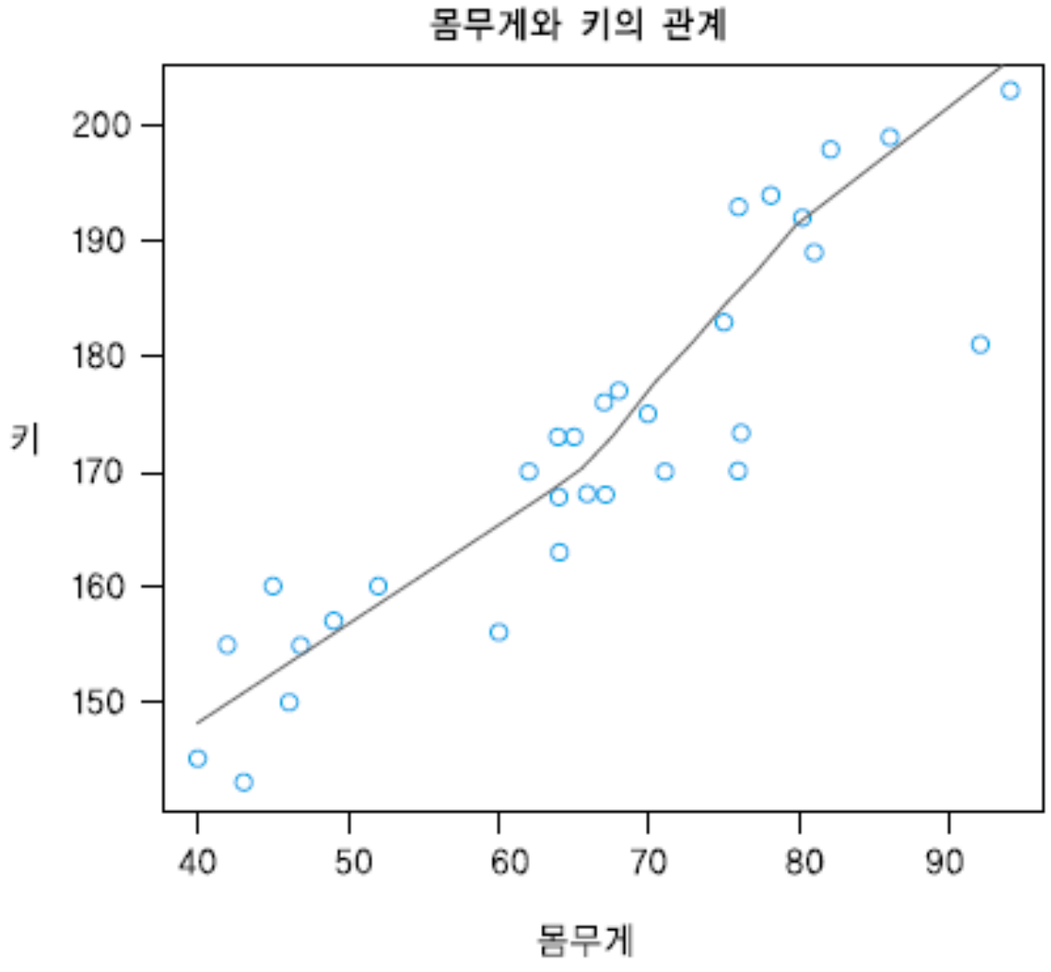
- 수집한 데이터의 전체적인 특성을 분석
  - Exploratory Data Analysis: EDA
- 본격적인 데이터 분석에 앞서 수집한 데이터가 분석에 적절한지 알아보는 과정
- 기본적인 통계적 특성 파악
  - 숫자형 데이터의 평균, 최대값, 최소값, 표준편차, 분산 등
  - 시각화 도구 이용

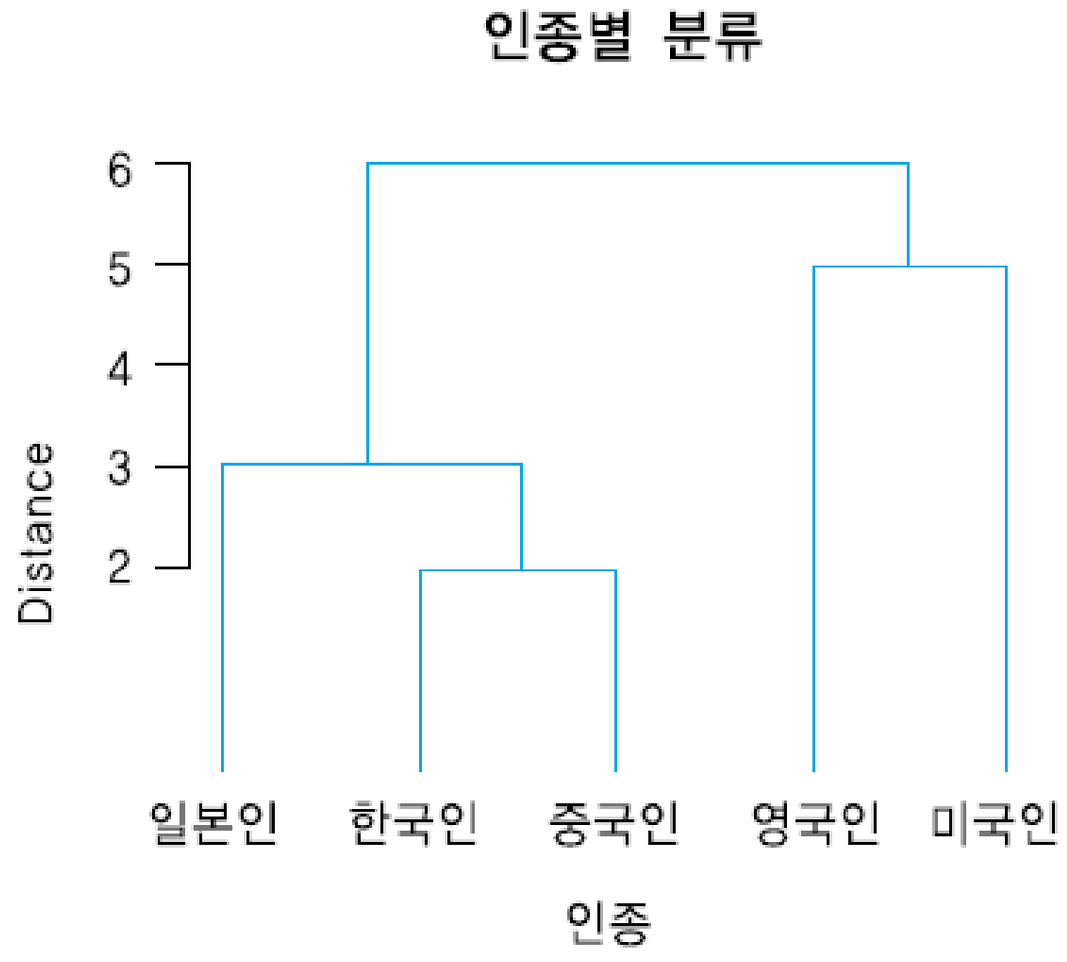
데이터 시각화

- 데이터 시각화(visualization)란 그래프, 도표, 도형 등을 이용하여 데이터의 특징을 파악하게 하는 것
- 숨어 있던 새로운 의미를 찾아낼 수 있음
- 데이터 탐색 뿐만 아니라 분석 결과를 고객에게 설명할 때에도 필수
- 위치, 길이, 각도, 방향, 형태, 면적, 부피, 명암, 색상 정보를 활용

기본적 시각 모형 - 위치

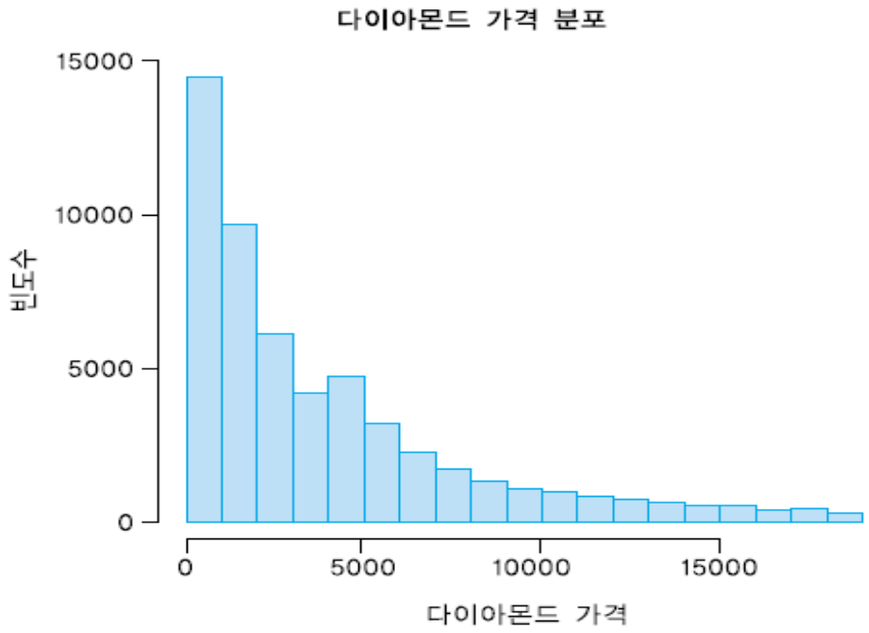
- 산포도(scattering plot), 덴드로그램(dendrogram)

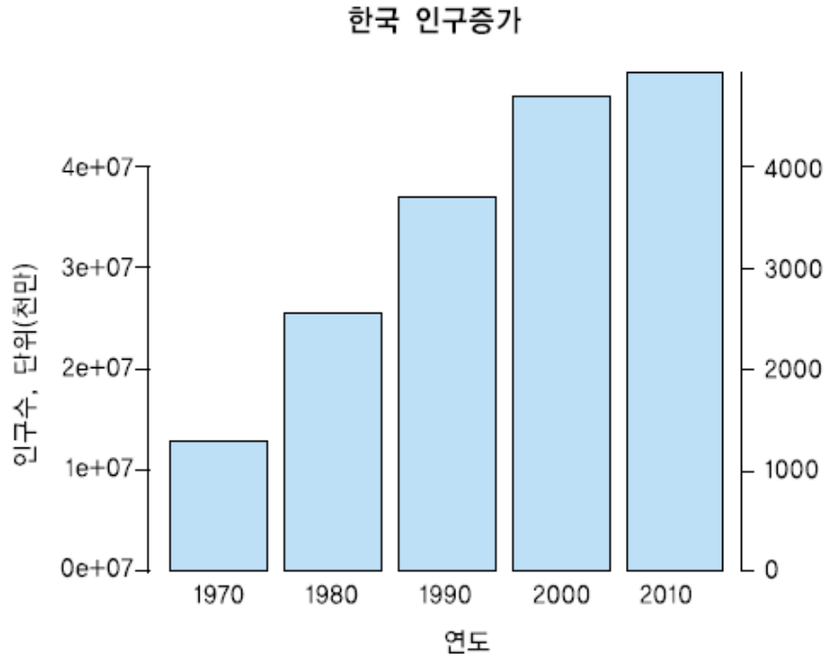


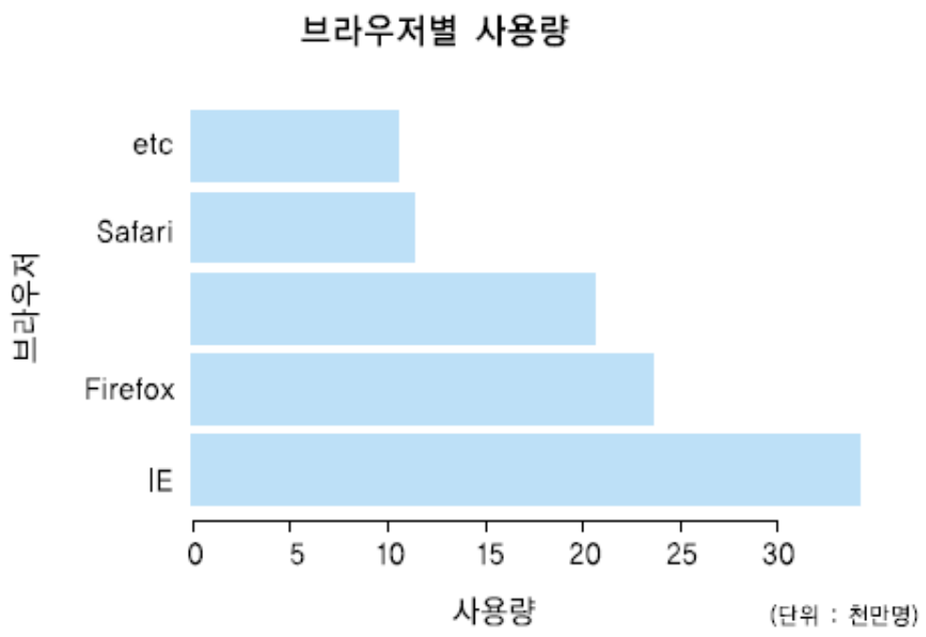


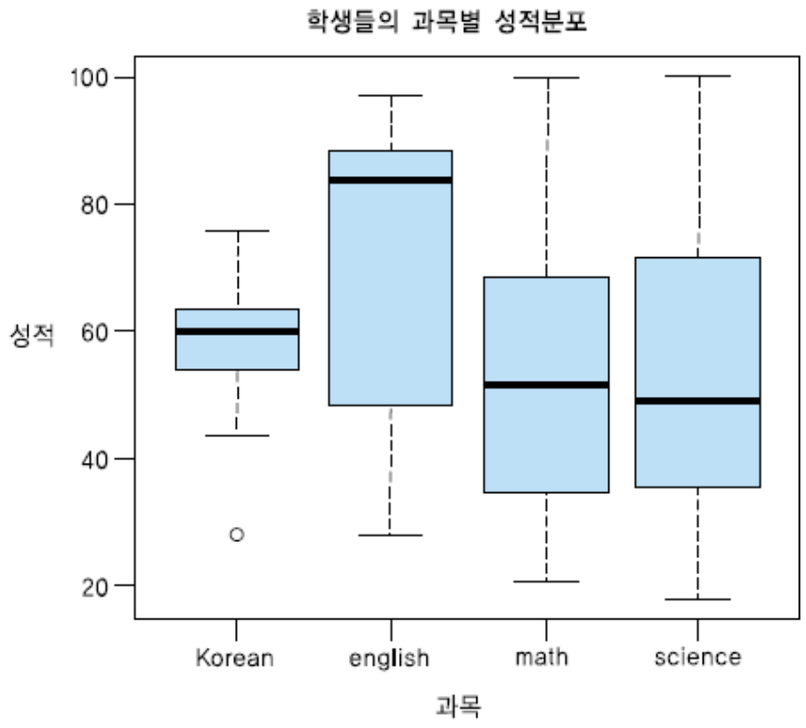
기본적 시각 모형 - 길이

- 히스토그램, 바플롯(막대그래프), 박스플롯

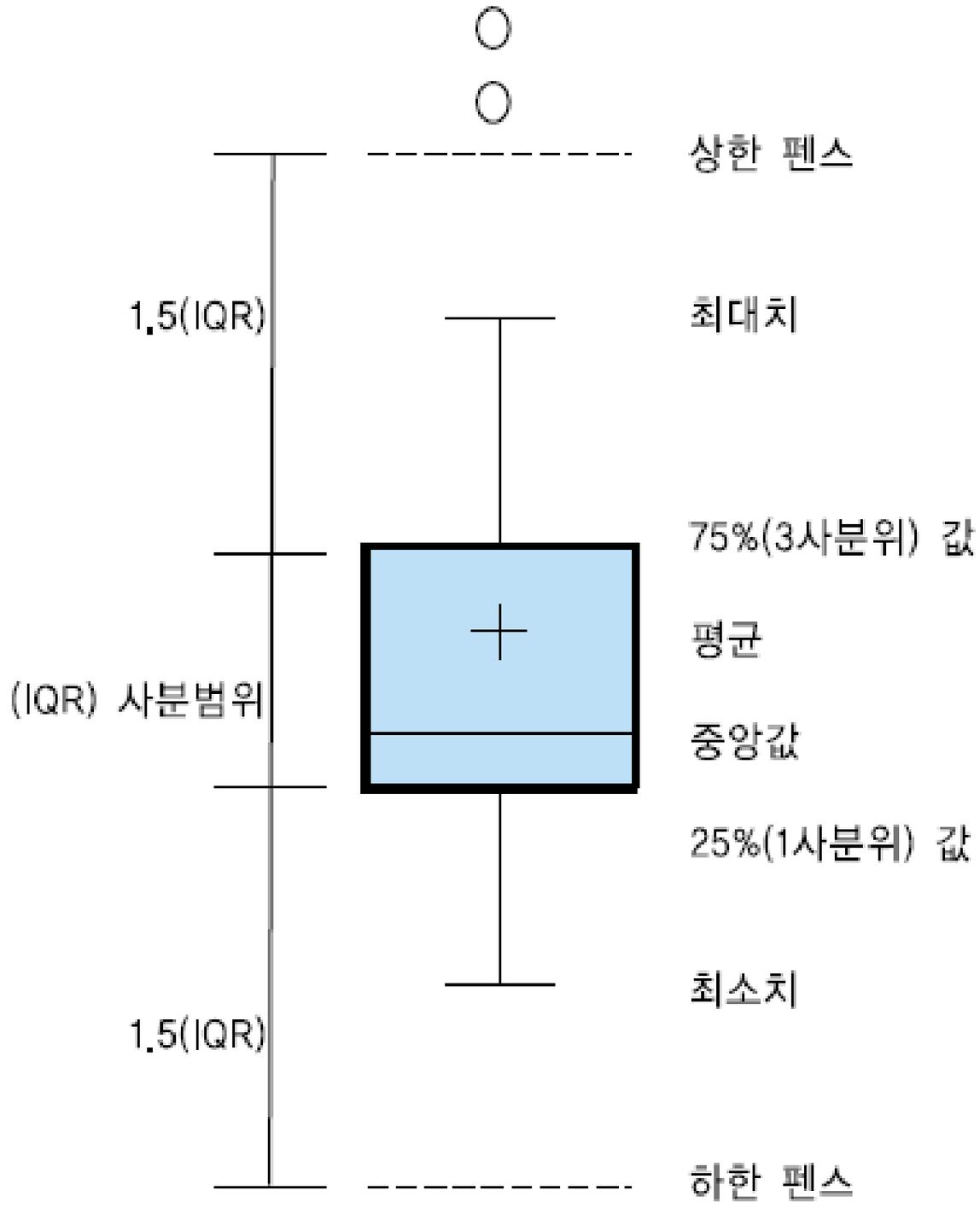






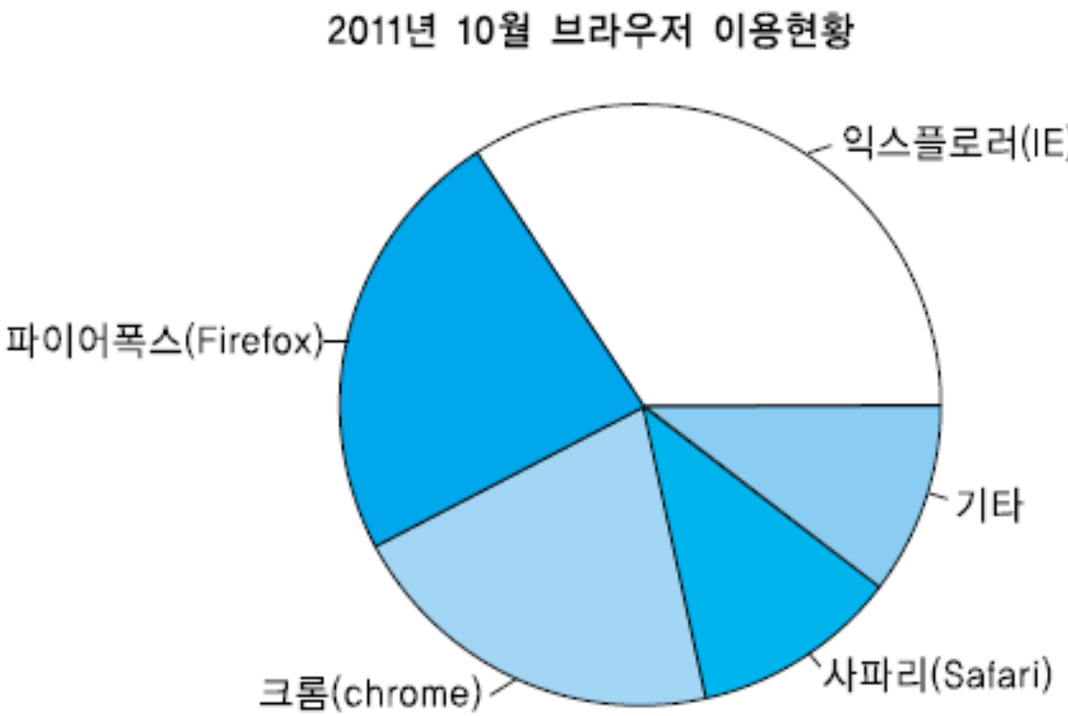


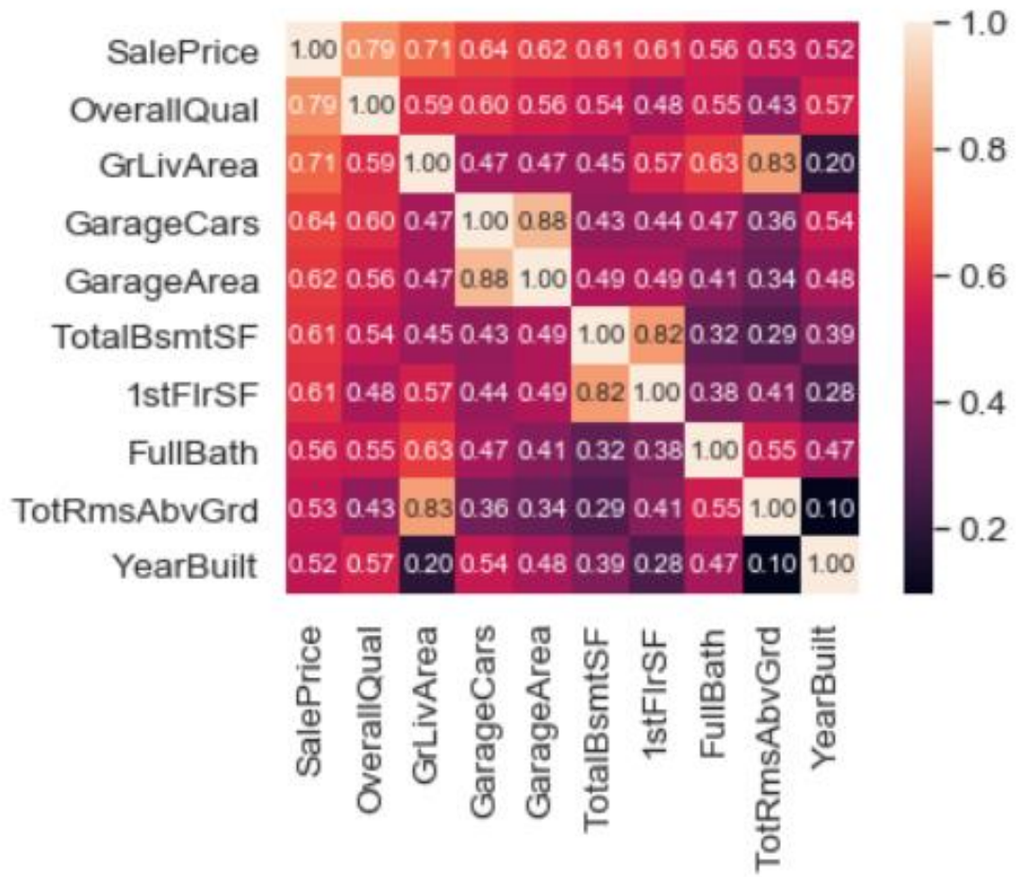
박스 플롯 boxplot()



기본적 시각 모형 – 각도, 면적/부피

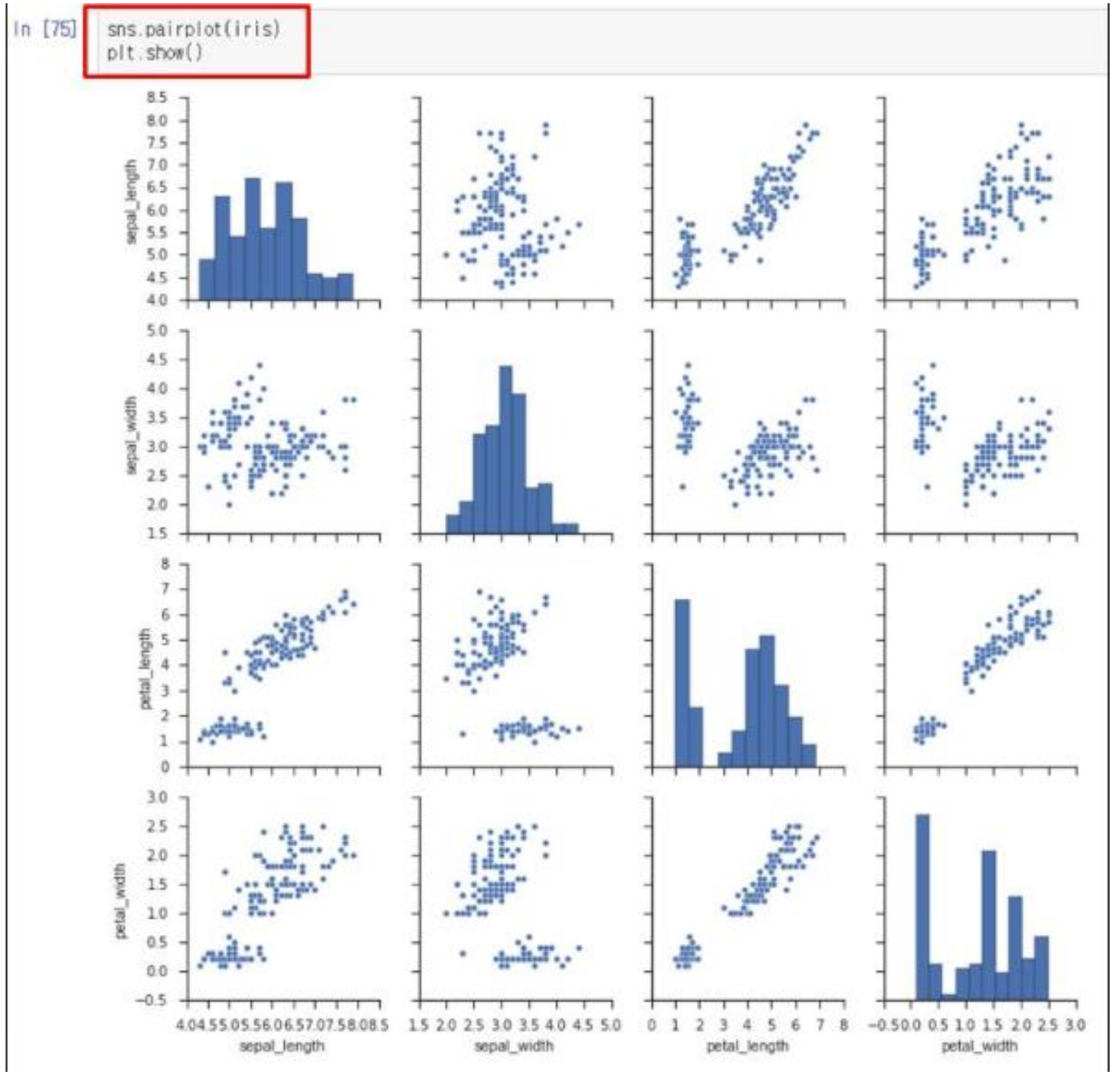
- Pie Chart(원그래프), Heatmap (히트맵)





기본적 시각 모형 – 각도, 면적/부피

- Pairplot (페어플롯)



Bi-Variate Analysis

- **Numerical & Numerical**
  - Scatter Plot
  - Linear Correlation
- **Categorical & Categorical**
  - Stacked column chart
  - Combination chart
  - Chi-square Test
- **Numerical & Categorical**
  - Line chart with Error bars
  - Combination chart
  - Z-test and t-test

Bi-Variate Analysis

- **Numerical & Numerical**
  - Scatter Plot
  - Linear Correlation

$$r = \frac{Covar(x,y)}{\sqrt{Var(x)Var(y)}}$$

$$Covar(x,y) = \frac{\sum(x-\bar{x})(y-\bar{y})}{n}$$

$$Var(x) = \frac{\sum(x-\bar{x})^2}{n}$$

$$Var(y) = \frac{\sum(y-\bar{y})^2}{n}$$

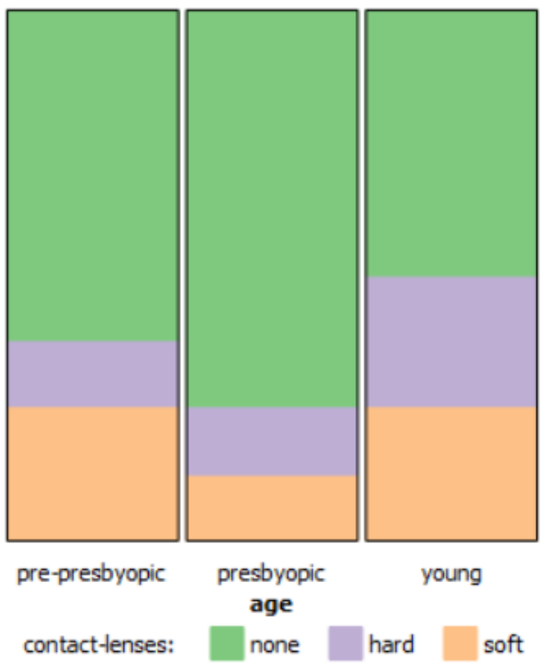
$r : Linear\ Correlation$

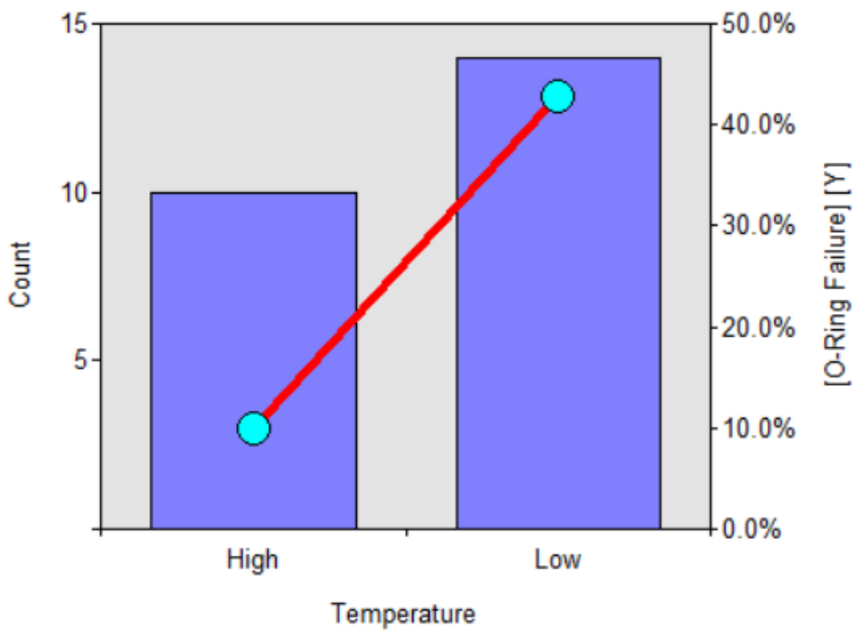
$Covar : Covariance$

$Var : Variance$

Bi-Variate Analysis

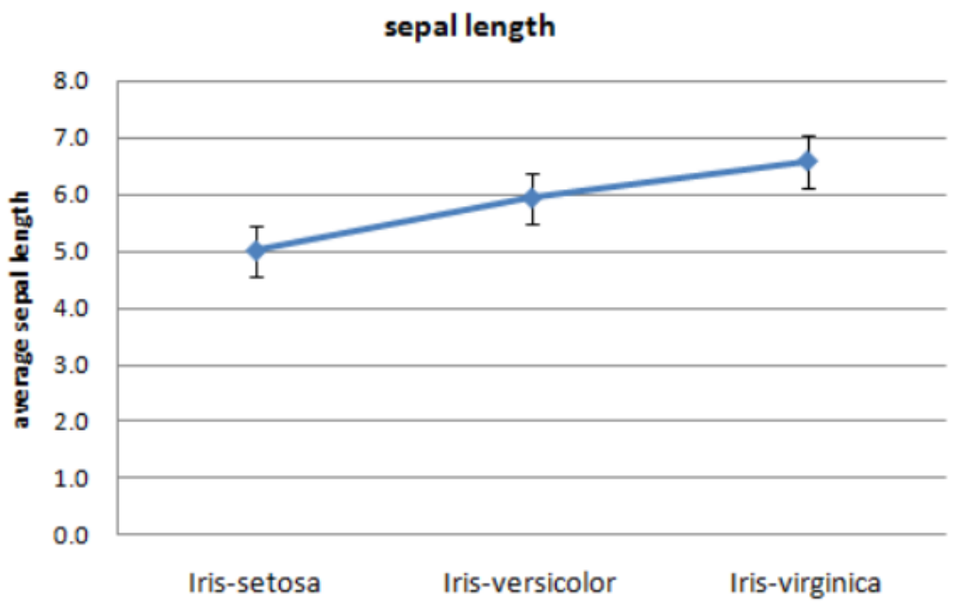
- **Categorical & Categorical**
  - Stacked column chart
  - Combination chart
  - Chi-square Test (교차분석)

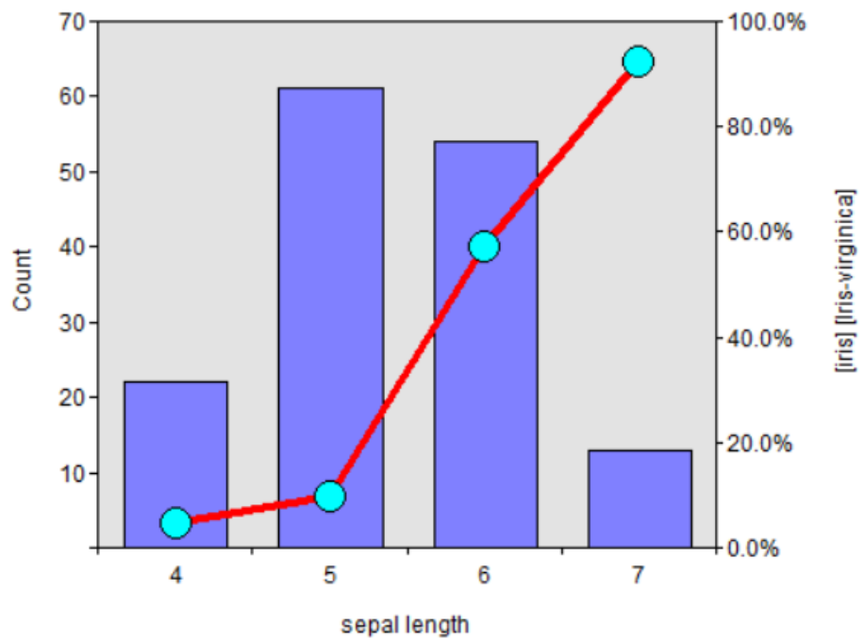




Bi-Variate Analysis

- **Categorical & Numerical**
  - Line chart with Error bars
  - Combination chart
  - Z-test and t-test





데이터 전처리

데이터 전처리(preprocessing)

- 수집한 데이터를 분석하기 좋게 변환하는 모든 작업으로 데이터 정제(Data Cleaning)라고도 함
- 분석 목적에 맞는지 데이터의 품질을 확인하고 필요하면 품질을 높이는 작업
- 데이터 품질
  - 신뢰성
  - 정확성
  - 적시성 (최신성) 등

데이터 전처리 종류

구분	처리 방법
결측치 처리 (missing value) 처리	• 결측치가 포함된 항목을 모두 버리는 방법 (버리는 항목의 비중이 크면 무시하기 어려움) • 결측치를 적절한 값으로 대체 (평균값, 인접 값으로 추정, 0, 최소값, 특정 상수 등) • 분석 단계로 결측치 처리를 넘김(NA로 표기) • 별도의 범주형 변수를 정의하여 추적 가능하게 관리 • dataframe.dropna() dataframe.fillna(0) dataframe.fillna(data.mean())
틀린값 처리 (invalid value) 처리	• 틀린 값이 포함된 항목을 모두 버리는 방법 • 틀린 값을 다른 적절한 값으로 대체 • 분석 단계로 틀린 값의 처리를 넘김 • (예) 키 3.7 미터, 양수가 있어야 할 곳에 음수, 등

데이터 전처리 종류

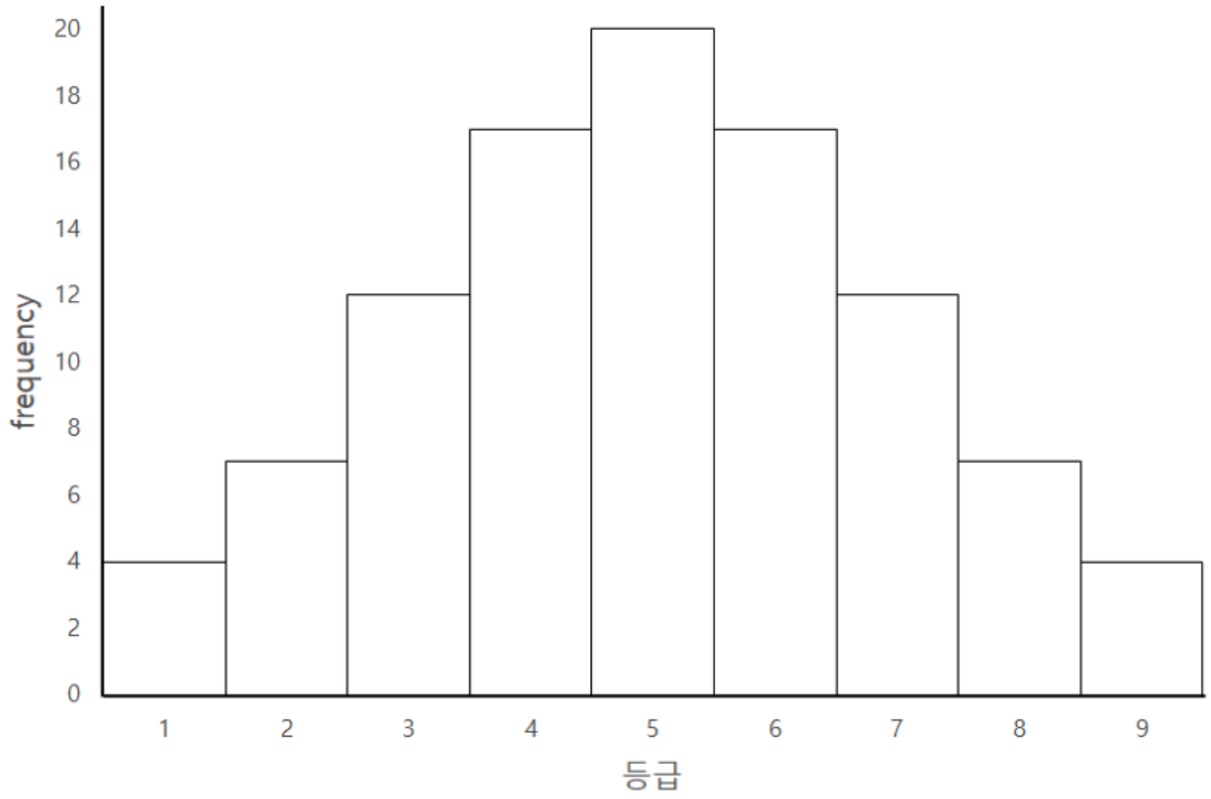
구분	처리 방법
이상치 처리 (outliers)	• 값이 일반적인 범위를 벗어나 특별한 값을 갖는 경우 • 데이터 분석 과정의 활동이므로 분석 단계로 넘김 • 도난 카드의 사용, 불법 보험료 청구 등의 탐지 • (예) 키 2.0 미터 – 극히 드물지만 가능
데이터변환	• 범주형 데이터 변환 • 로그변환 • 역수변환 • 스케일링(min-Max Scaling, Standard Scaling, Robust Scaling)

데이터 변환

- 데이터를 주어진 그대로 사용하지 않고 다른 형태로 바꾸어 사용하는 것이 필요한 경우가 많다.
  - 같은 성적을 나타내는데 A, B, C 등 학점으로 표현하거나 100점 만점으로 환산하기도 한다 (97, 94, 91 등).

범주형으로 변환

- 수치 데이터의 개별 값 구분이 오히려 혼란스러울 때
- 나이 => 10대, 20대, 30대, 40대
- 연간 소득 => 고소득층, 중간층, 저소득층
- 내신 등급 분포 (등급 차이에 대한 느낌이 같도록 정한다)



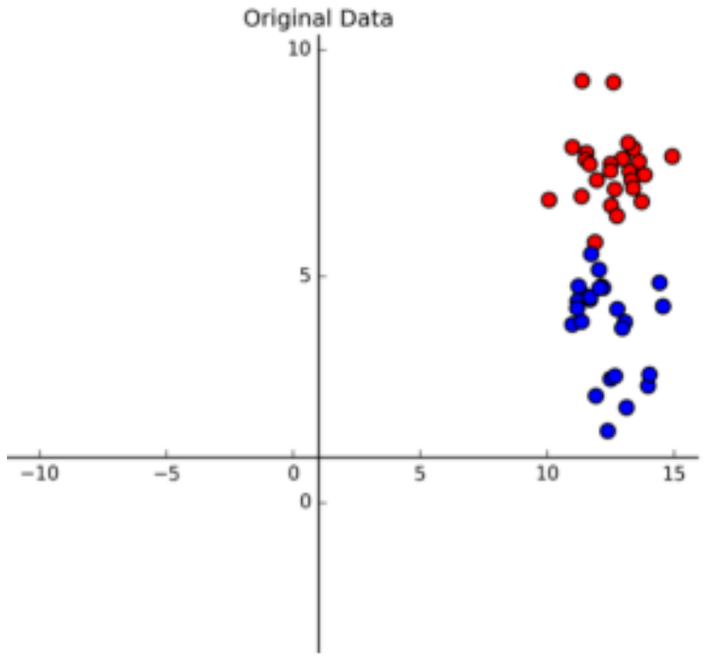
범주형 변수 코딩 *

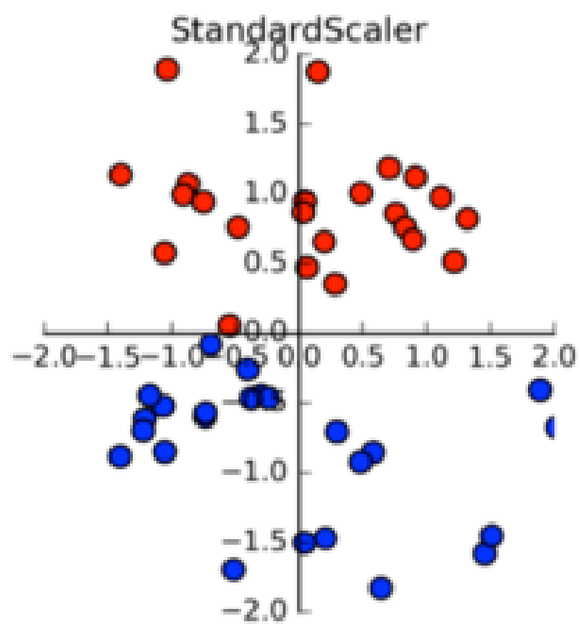
- 예를 들어 요일을 1, 2, 3, 4, 5, 6, 7 등으로 표시한 경우 이 변수를 컴퓨터가 연산(덧셈이나 곱셈)을 할 수 있는 숫자로 인식해서는 안 된다.
- 이 숫자를 범주형(카테고리형)으로 분명하게 처리되어야 한다. 컴퓨터가 범주형(카테고리형) 변수를 분명히 인식하게 하는 방법이 필요하다
- **One Hot Encoding**
  - 하나로 하나의 특성(컬럼)만 1이 될 수 있고 다른 특성은 모두 0으로 코딩하는 방법
  - 판다스가 제공하는 get_dummies()를 사용하면 카테고리형 변수들을 One-Hot encoding 으로 만들어준다
- **Ordinal Encoding**: 순서 있는 범주형
- **Label Encoding**: target 변수의 encoding

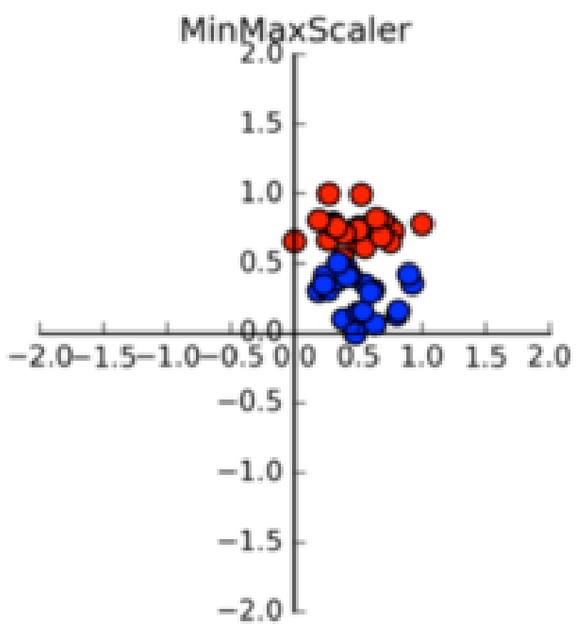
스케일링(Scaling)

- 원래 데이터가 갖는 값의 범위를 다르게 조정하는 작업
- 스케일링을 하는 이유는 여러 특성 변수의 중요도를 갖게 맞추기 위해서이다.
- 최소-최대 스케일링 (min-max scaling)
  - 예를 들어 모든 시험은 100점 만점으로 환산해야 동일한 비중으로 취급되며, 어떤 과목은 50점 만점, 어떤 과목은 80점 만점이면 동일한 조건으로 특성이 반영되지 않는다.
  - 주어진 값의 최소값을 0으로 최대값을 1로 재조정하는 것
  - 파이선에서는 MinMaxScaler() 함수를 사용
  - zi = (xi – min) / (max – min)
- 표준 스케일링 (standard scaling)
  - 데이터 분포를 평균은 0, 표준 편차는 1이 되도록 정규화 하는 방법
  - 파이선에서 StandardScaler() 함수 사용
  - zi= (xi – mean) / sigma
- Robust Scaler
  - IQR(inter-Quartile Range) 에 대하여 스케일링 (Outliers 에 robust)
  - 파이썬에서는 RobustScaler() 함수 사용
  - zi = (xi – median) / IQR

스케일링 비교

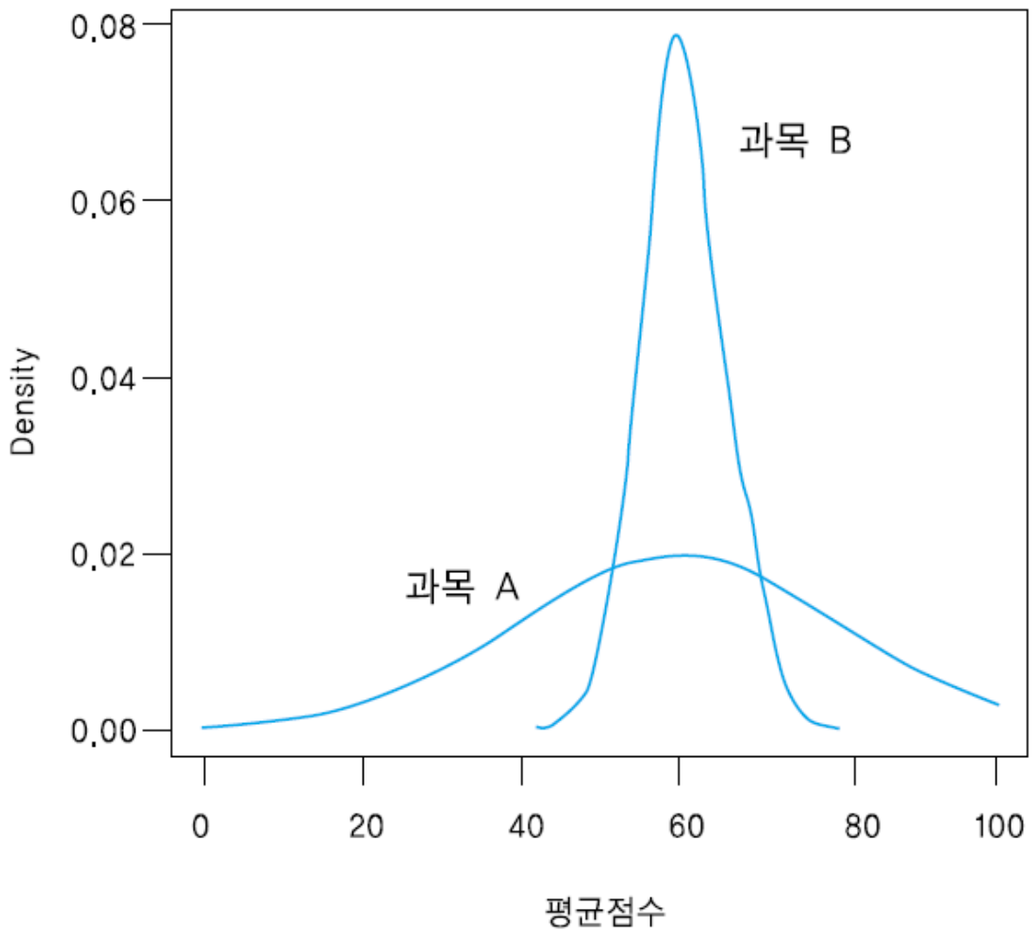

Original Data
10
5
0
−10
−5
0
5
10
15


StandardScaler
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0


MinMaxScaler
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

표준 스케일링 (표준 정규화)

- Who is better?



학생	과목 A	과목 B	평균
갑	90	80	85
을	80	90	85

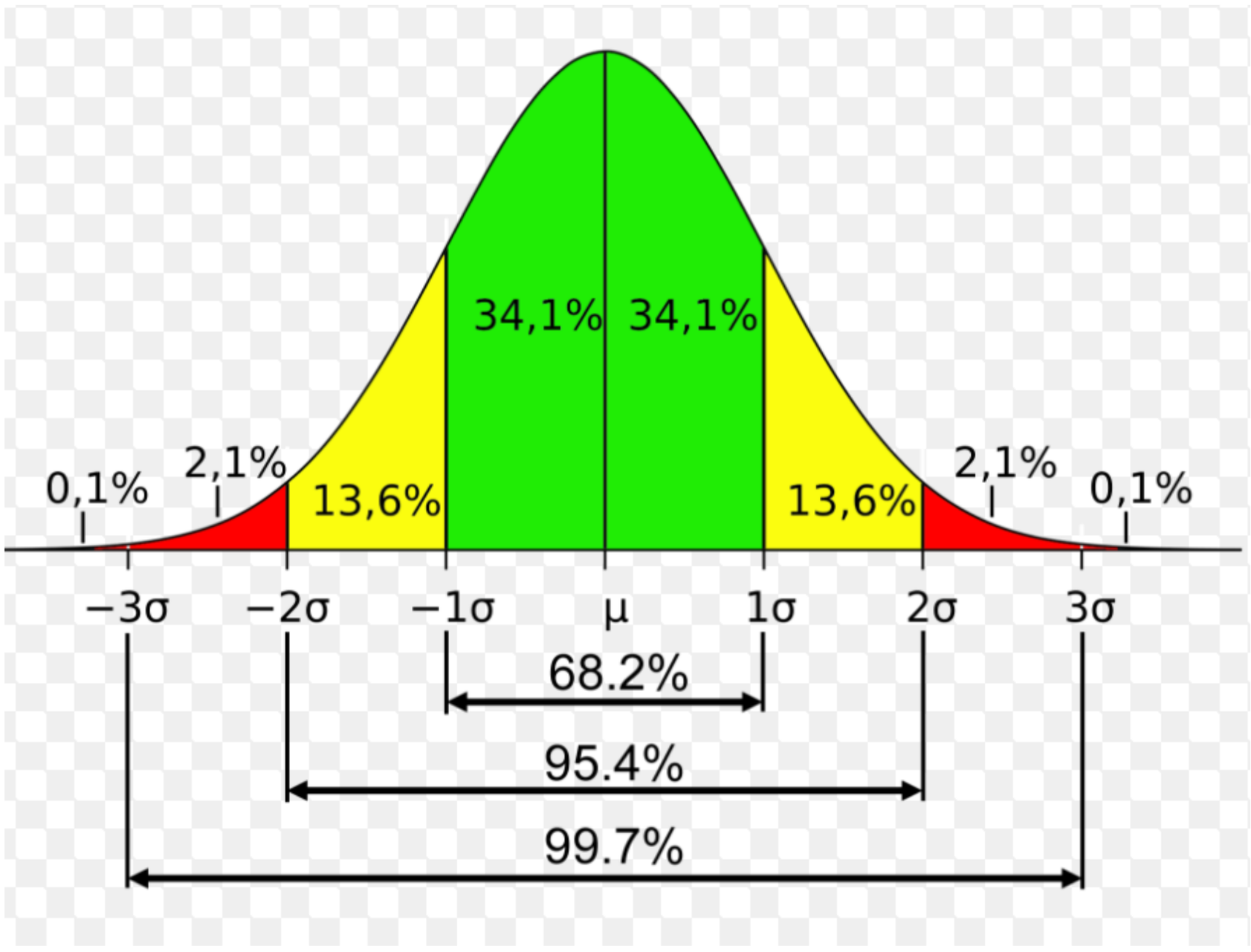
표준 정규화

- 표준정규분포(Standard normalization, mean=0, sigma=1)로 변환
- Z-변환(Z-score Transform) 사용

$$z = \frac{x - u}{\sigma}$$

	학생	과목 A	과목 B	평균
변환 전	갑	90	80	85
	을	80	90	85
변환 후	갑	(90 - 60) / 20 = 1.5	(80 - 60) / 5 = 4	2.75
	을	(80 - 60) / 20 = 1	(90 - 60) / 5 = 6	3.50

정규 분포(Normal Distribution)

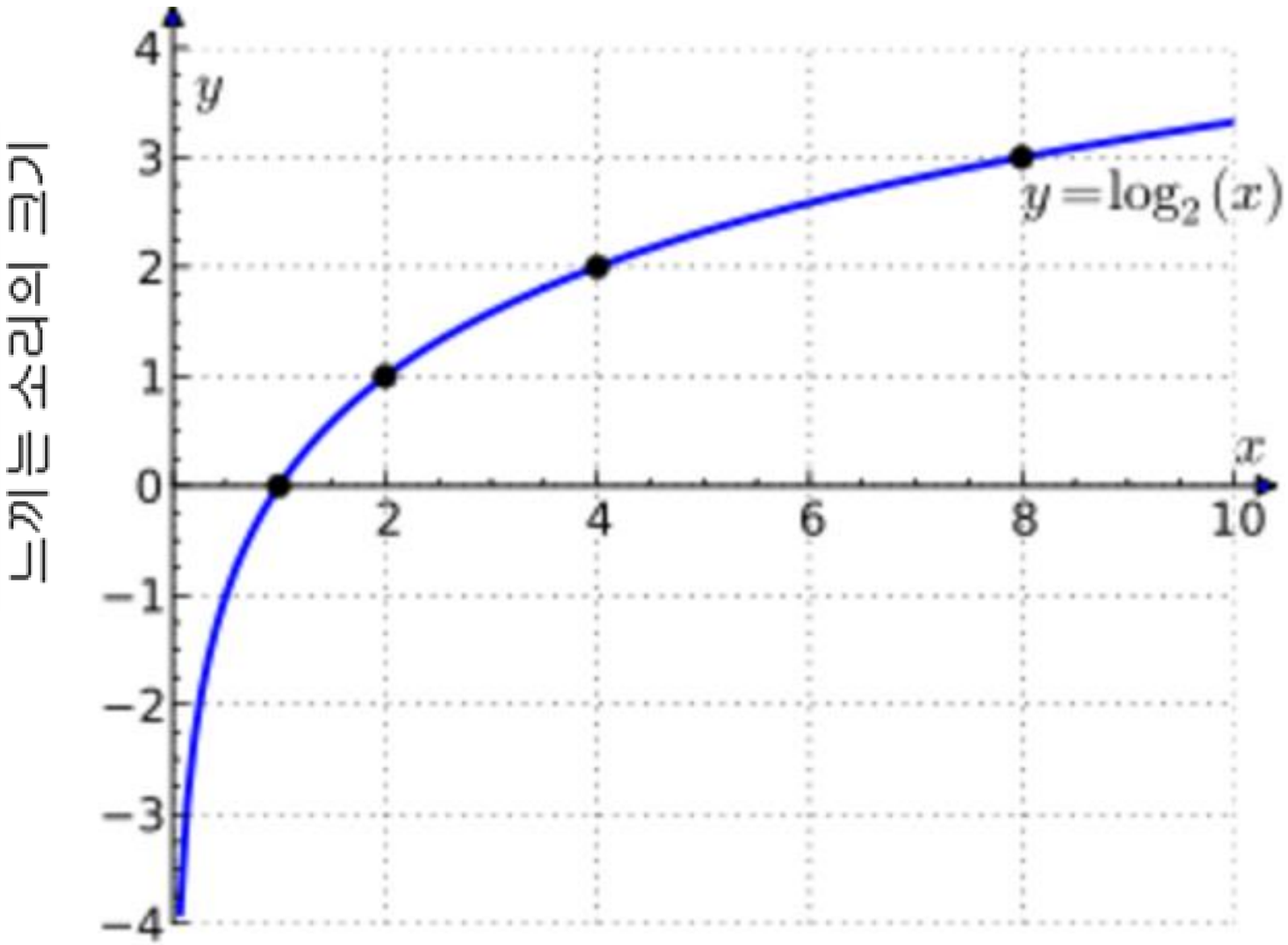


로그 변환

- 로그의 도입
  - **체감형 수치를 선형적으로 표현할 때 사용** - 사람이 자연적으로 느끼는 느낌의 양을 수학적 모델로 설명할 때 사용
  - 돈, 소리, 빛, 압력, 냄새 등 생물학적인 자극을 주는 경우
- 같은 자극을 느끼려면 현재 보유한 양이 많을수록 이에 비례한 더 강한 자극이 필요하다는 것
- 이를 수학적으로 표현하면 로그 함수가 됨
- 현재 보유한 양이 x이고 이의 변화량, 즉 미분값이 1/x이 되려면 로그 함수를 얻음
- 로그를 취한 이후의 값에 대해서 사람들이 변화량을 느끼는 것이 선형적이라는 특성

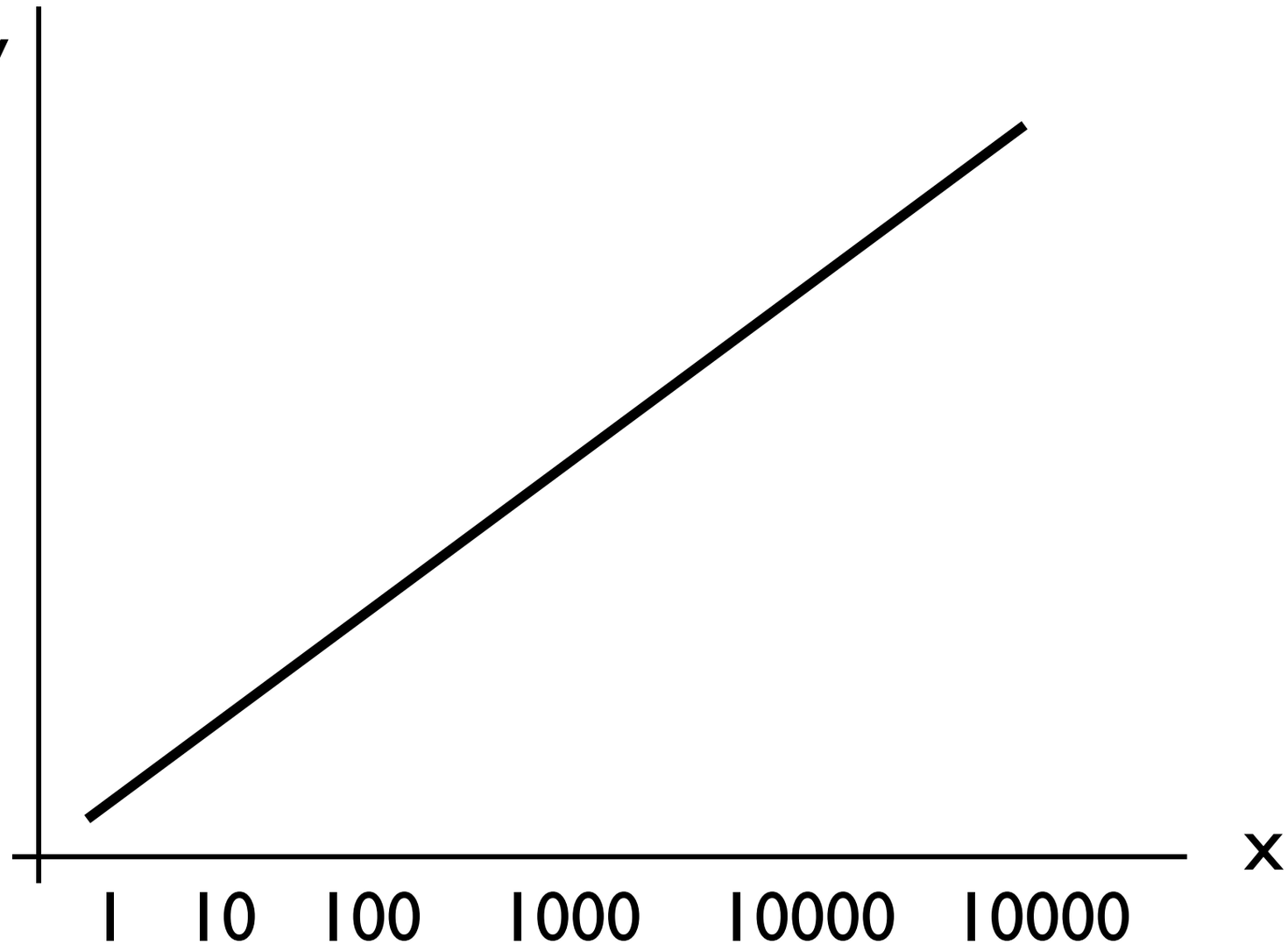
로그 함수

- Log(x)의 기울기(미분): 1/x



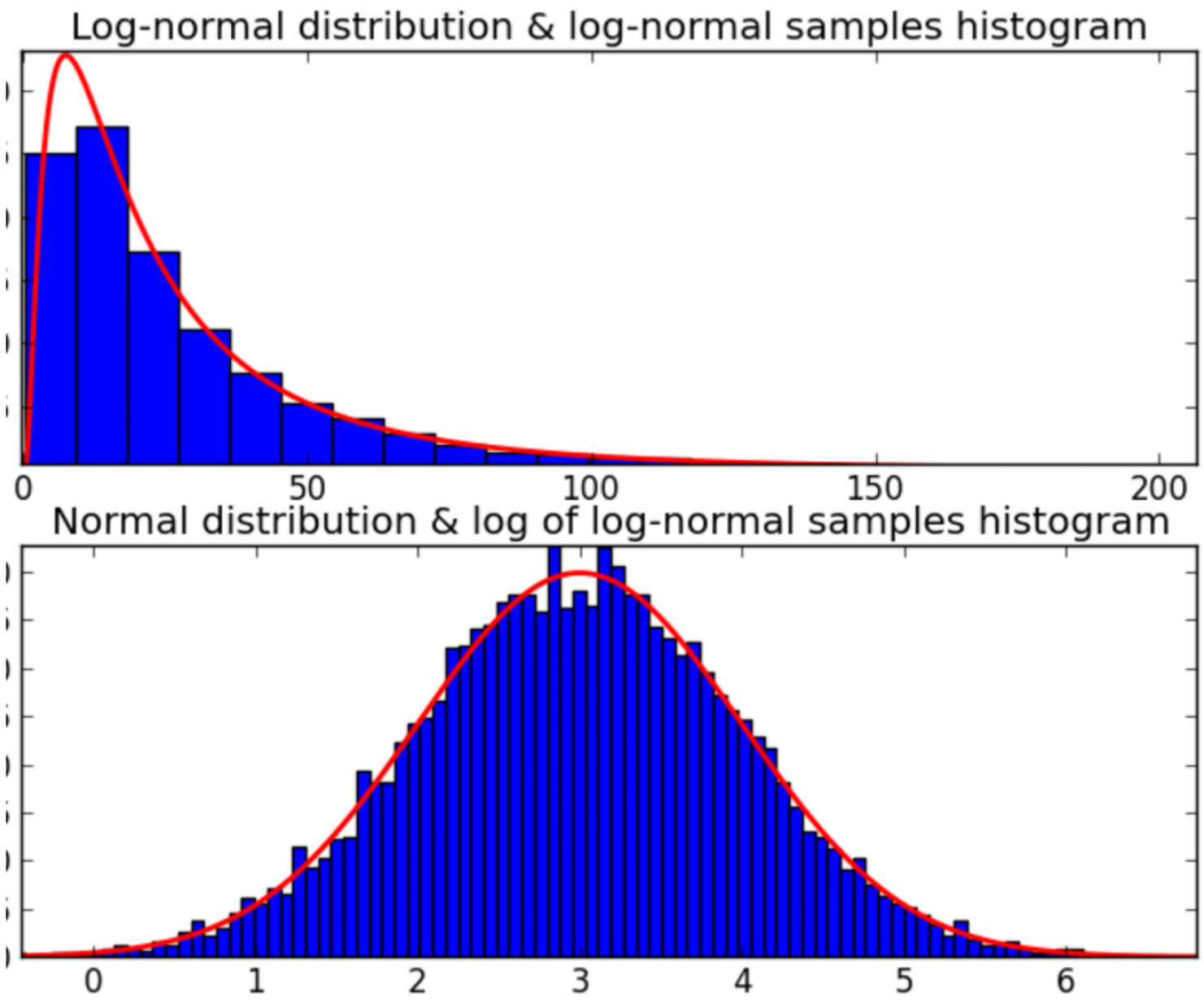
로그 스케일

- 로그 스케일 입력에 대해 선형 특성을 갖는 경우



로그-노멀 분포

- (ex) 도시의 인구, 재산분포



역수 변환

- 역수를 사용하면 선형적인 특성을 가져 분석의 정확도가 높아지는 경우
- 자동차 마일리지(연료 1L로 가는 거리 Km)와 연비(100km 주행하는데 필요한 연료 L)는 모두 자동차의 성능을 나타내지만 서로 역수의 관계
- 측정 목적:
  - 같은 비용을 얼마나 멀리 갈 수 있는가?
  - 같은 거리를 여행하는데 비용이 얼마가 드는가?

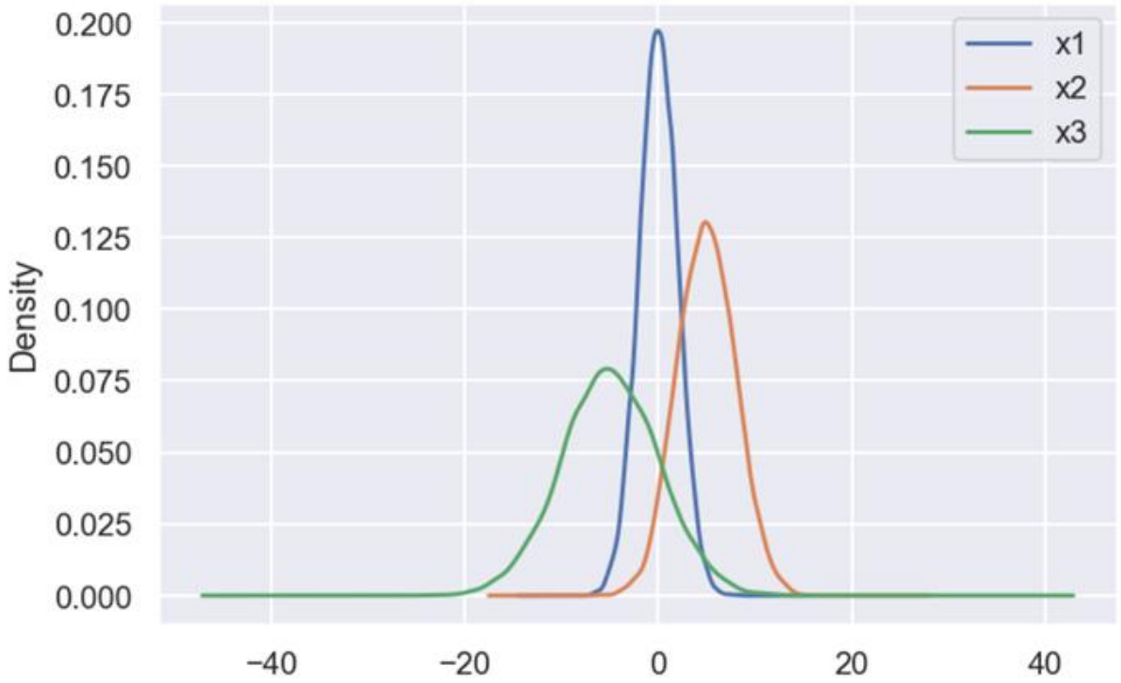
데이터 변환

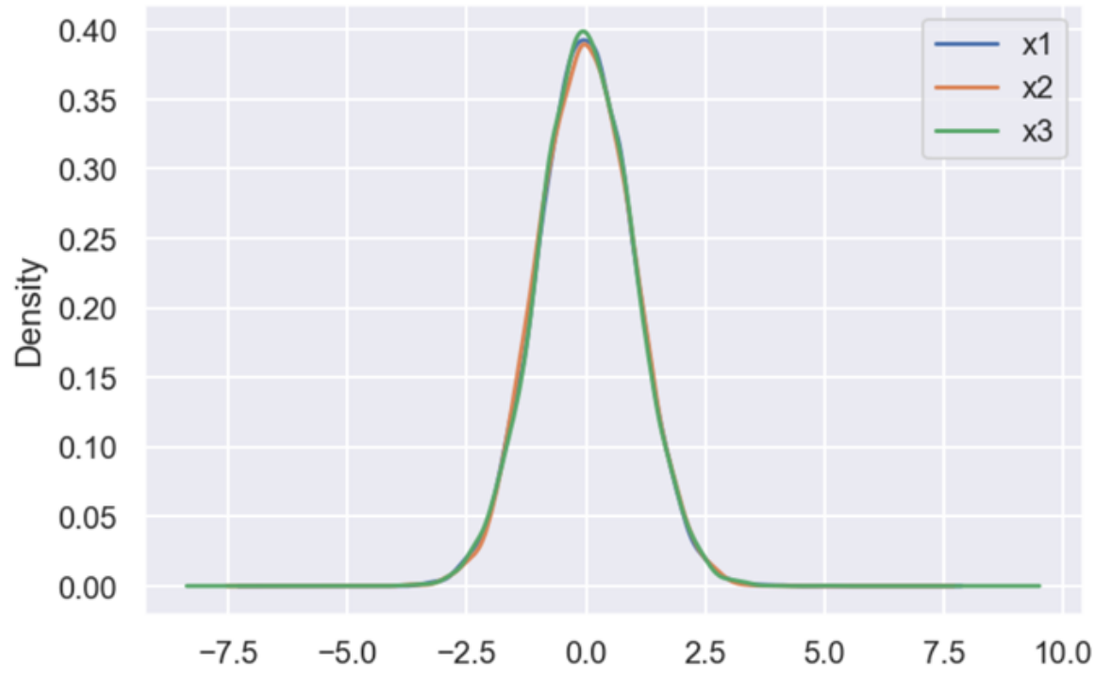
- 선택을 적절히 해야 한다

구분	내용
범주형으로 변환	• 수치 데이터가 아닌 것을 명시
Min-Max Scaler	• 수치 데이터의 범위가 다를 때
Standard Scaler (z-score 정규화)	• 일반 정규화에 표준 편차를 고려한 변환
Robust scaler	• 이상치에 강함
로그 변환	• 로그를 취하면 선형 특성을 가질 때 (또는 로그 정규 분포를 가질 때)
역수 변환	• 역수를 사용하면 선형적인 특성을 가질 때

정규화(Scaling)

- 표준정규화(Standard Scaler)





정규화(Scaling)

- (예제) Standard, Min-Max, Robust Scaler (data with outliers)

```
df = pd.DataFrame({'x':np.concatenate([np.random.normal(20, 1, 1000),
                                       np.random.normal(1, 1, 50)])
                   })
```