PART II-03
머신러닝 기본

통계기초와 데이터 시각화(seaborn)

시간계획

- 오늘도 파이팅 입니다.^^

시간	학습내용
09:00~10:40	통계이해, 대표 통계량 이해 및 실습
11:00~12:40	산포 통계량 이해 및 실습
12:40~14:00	즐거운 점심 시간
14:00~15:00	분포 통계량 이해 및 실습
15:20~16:20	Seaborn 실습
16:40~17:50	Seaborn 실습, 머신러닝 환경 설정

RAPA 한국전파진흥협회

머신러닝 데이터 탐색을 위한 통계 기초 이해

학습 목표

- 통계학 용어에 익숙해 진다.
- 탐색적 데이터 분석을 이해한다.
- 통계학의 용어와 통계 모델링에 대해 살펴본다.
- 기술 통계, 산포 통계, 분포 통계의 핵심 내용에 대해 안다

일화적 증거

- 요즘 애들 대부분 핸드폰을 하루에 5시간 이상한다.
- 옆집에도 하루에 5시간 이상하고, 동생내도 아이들이 하루에 5시간 이상 하고, 우리 아들도 핸드폰 하기 위해 공부하는 모양으로 핸드폰 시간이 6시간이 넘는다.
- 그러니 요즘애들은 핸드폰을 하루에 5시간 이상할 것 같다.

- 일화적 증거의 신빙성이 부족한 이유
  - 적은 관측치
  - 선택편의
  - 확증편의(편향)
  - 부정확함

통계적 접근법

- 자료수집(Data collection) : 대표성을 띠는 자료 수집
- 기술통계(Descriptive statistics) : 데이터 통계량 요약,시각화
- 탐색적 데이터 분석(Exploratory data analysis)
  - 데이터의 패턴, 차이점, 특이점 찾기
- 추정(Estimation) : 사용한 데이터를 토대로 모집단 특징 추정
- 가설검정(Hypothesis testing) : 데이터셋에서 두 그룹 간의 차이가 명백한 것인지 혹은 우연한 사건인지 평가

탐색적 데이터 분석(EDA; Exploratory Data Analysis)

그래프를 통한 **시각화와 통계 분석** 등을
바탕으로 수집한 데이터를 **다양한 각도에서**
관찰하고 이해하는 방법

탐색적 데이터 분석을 위한 통계 기초 숙지 필요

탐색적 데이터 분석 필요 이유

- 데이터가 표현하는 현상을 더 잘 이해
- 데이터에 대한 잠재적인 문제를 발견
- 분석에 필요한 데이터를 결정
- 문제 정의 단계에서 미쳐 발생하지 못했을 다양한 패턴을 발견
- 기존의 가설을 수정하거나 새로운 가설을 세울 수 있음

통계학이란?

- 수치 데이터의 **수집, 분석, 해석, 표현** 등을 다루는 수학의 한 분야
- **기술통계학**
  - 연속형 데이터 -> 평균, 표준편차와 같은 자료 요약 — 키, 나이, 가격 등
  - 범주형 데이터 -> 빈도, 백분율과 같은 자료 요약 — 성별, 성씨 등
- **추론 통계학**
  - **표본**이라 불리는 **일부 자료를 수집**해서 **전체 모집합에 대한 결론을 유추**하는 것
  - 추론은 **가설검정, 수치의 특징 계산, 데이터 간의 상관관계** 등을 통해 이뤄짐
- **통계 모델링**
  - 데이터에 통계학을 적용해 **변수의 유의성을 분석함**으로써 **데이터의 숨겨진 특징**을 찾아내는 것

통계모델

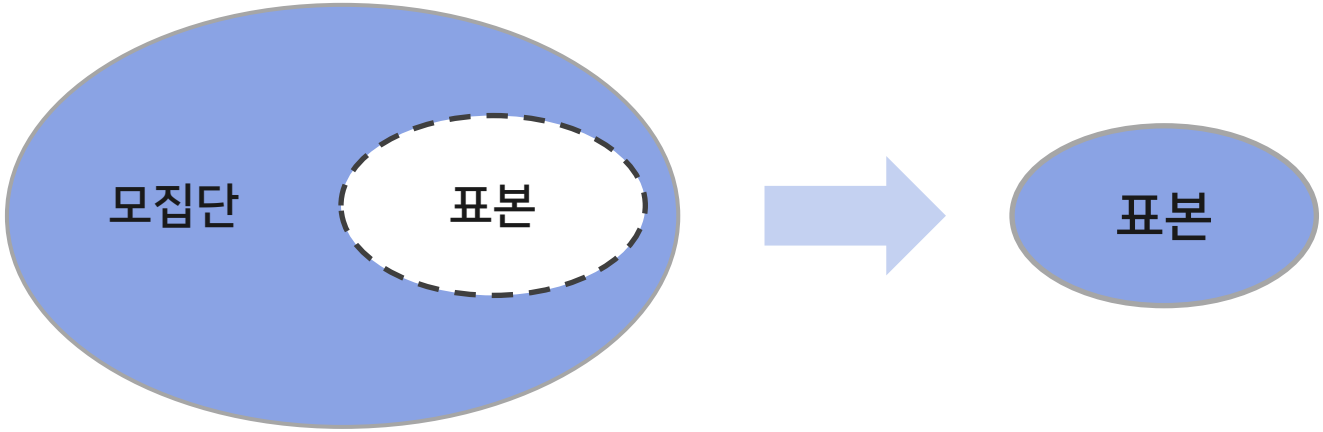
- **통계 모델**
  - 수학적 모델 : 변수들로 이뤄진 수학식을 계산해 **실제 값을 추정**하는 방법
  - 통계 모델을 이루는 여러 가정은 확률 분포를 따름
    : **이산 확률 분포**와 **연속 확률 분포**
  - **모든 변수가 만족해야 하는 기본 가정으로 시작**하며, 이 조건이 만족할 때만 **모델의 성능이 통계학적으로 의미**를 가짐

- 머신러닝 분야의 발전 방향
  - 수학 -> 통계학 -> 컴퓨터 과학 -> 머신러닝

통계모델

모집단 모든 관측값 또는 분석 대상의 전체 데이터를 의미

표본 모집단의 부분 집합으로, 분석 대상 중인 전체 데이터의 일부분



통계모델

- 모수 : **모집단의 특징**을 나타내는 수치값
- 통계량
  - **표본의 특징을 나타내는 수치값**으로, **모수 추정**을 위해
  - 사용 통계량을 계산하는 **기초통계분석(기술통계분석)**을 바탕으로 **확률 분포**를 간단하게 확인 가능
  - **평균, 중앙값, 최빈값, 분산** 등과 같은 데이터를 대표하는 값

통계량이 실제 모집단을 **대표하는 값**이 될 때,
통계적 유의성을 확보할 수 있음

RAPA 한국전파진흥협회

기초 통계 분석

기초 통계 분석

- 통계량의 분류

구분	내용	예시
대표 통계량	데이터의 **중심 및 집중경향**을 나타내는 통계량	평균, 중위값, 최빈값 등
산포 통계량	데이터의 **퍼진 정도**를 나타내는 통계량	분산, 표준편차, 범위
분포 통계량	데이터의 **위치 정보 및 모양**을 나타내는 통계량	왜도, 첨도, 사분위수, 최대값, 최소값

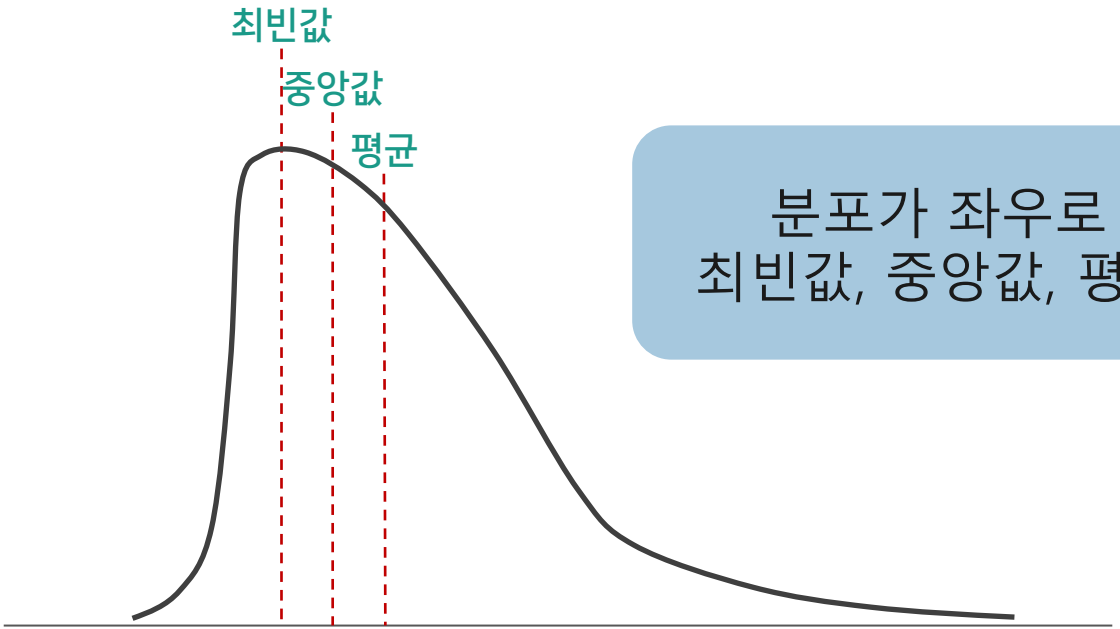
대표 통계량

- 통계량의 분류

구분	내용	예시
대표 통계량	데이터의 **중심 및 집중경향**을 나타내는 통계량	평균, 중위값, 최빈값 등
산포 통계량	데이터의 **퍼진 정도**를 나타내는 통계량	분산, 표준편차, 범위
분포 통계량	데이터의 **위치 정보 및 모양**을 나타내는 통계량	왜도, 첨도, 사분위수, 최대값, 최소값

대표 통계량

데이터의 대푯값(Representative value)
: 중심 경향(central tendency) 값이라고도 함.



분포가 좌우로 치우친 경우
최빈값, 중앙값, 평균이 모두 다름

참고 : https://brunch.co.kr/@data/13

대표 통계량 - 평균

- 산술평균 : 가정 널리 사용되는 평균으로 연속형 변수 사용
  - 이진 변수 대한 산술 평균은 1의 비율과 같음
  - 다른 관측치에 비해 매우 크거나 작은 값에 크게 영향을 받음
- 조화평균 : 비율 및 변화율 등에 대한 평균을 계산할 때 사용
  - 데이터의 역수의 산술평균의 역수
- 절사평균 : 데이터에서 α~1- α 의 범위에 속하는 데이터에 대해서만 평균을 낸 것
  - 매우 크거나 작은 값에 의한 영향을 줄이기 위해 고안됨

대표 통계량 - 평균

- 파이썬을 이용한 평균 계산

구분	구현 함수
산술 평균	• numpy.mean(x) • numpy.array(x).mean() • Series(x).mean()
조화 평균	• len(x) / numpy.sum(1/x) • scipy.stats.hmean(x)
절사 평균	• scipy.stats.trim_mean(x, proportiontocut) - proportiontocut: 절단한 비율

[실습]대표 통계량

데이터의 대푯값

```
1  import numpy as np
2  from scipy import stats
3
4  np.random.seed(0)
5
6  data = np.random.randint(0, 100, 10000)
7
8  mean = np.mean(data); print("평균값: ", mean.round(2))
9  median = np.median(data); print("중앙값: ", median)
10 mode = stats.mode(data); print("최빈값:  {} ({})".format(mode[0][0], mode[1][0]))
```

```
평균값:  49.17
중앙값:  49.0
최빈값:  3 (125)
```

numpy 라이브러리는 최빈값 관련 함수를 제공하지 않으므로, scipy 패키지의 status 모듈에 있는 mode() 함수를 사용

mode[0] : 최빈값
mode[1] : 최빈값의 빈도

[실습]대표 통계량

데이터의 대푯값

```
1  import numpy as np
2  from scipy import stats
3
4  np.random.seed(0)
5
6  data = np.random.randint(0, 100, 10000)
7
8  mean = np.mean(data); print("평균값: ", mean.round(2))
9  median = np.median(data); print("중앙값: ", median)
10 mode = stats.mode(data); print("최빈값:  {} ({})".format(mode[0][0], mode[1][0]))
```

평균값: 49.17
중앙값: 49.0
최빈값: 3 (125)

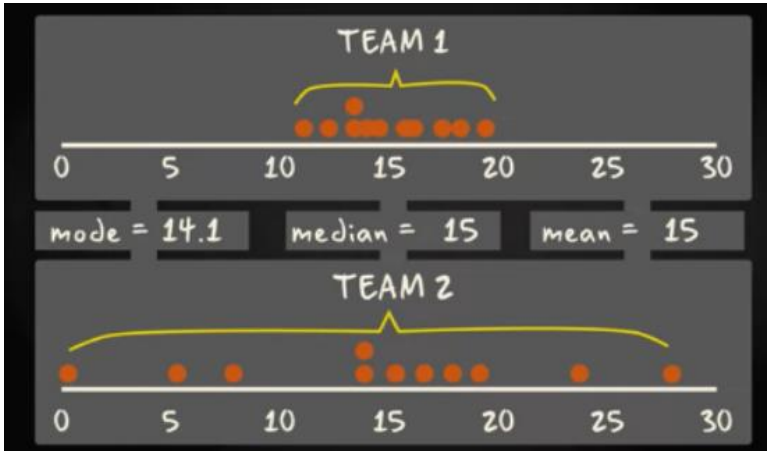
numpy 라이브러리는 최빈값 관련 함수를 제공하지 않으므로, scipy 패키지의 status 모듈에 있는 mode() 함수를 사용

mode[0] : 최빈값
mode[1] : 최빈값의 빈도

[문제 해결]
pandas의 메소드를 활용해 동일한 결과 나오도록 해결해 보세요.

대표 통계량의 문제점

- Team1과 Team2는 분명히 다른 값을 갖고 있지만 평균값과 중위값, 최빈값은 동일하게 나타남
- 대푯값 만으로는 데이터를 설명하는데 한계가 있음



출처 : cousera.org

데이터의 변화량을 나타내는 것이 필요함.

산포 통계량

- 통계량의 분류

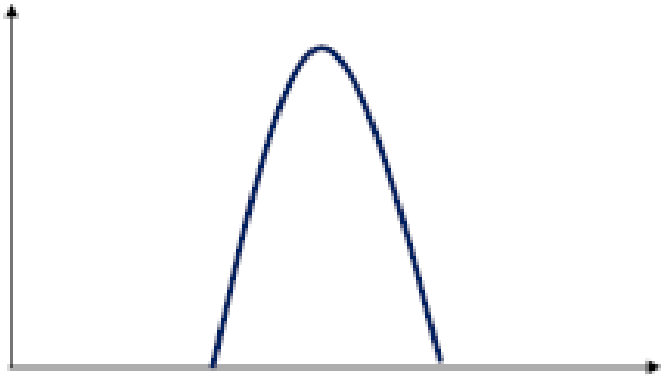
구분	내용	예시
대표 통계량	데이터의 **중심 및 집중경향**을 나타내는 통계량	평균, 중위값, 최빈값 등
산포 통계량	데이터의 **퍼진 정도**를 나타내는 통계량	분산, 표준편차, 범위
분포 통계량	데이터의 **위치 정보 및 모양**을 나타내는 통계량	왜도, 첨도, 사분위수, 최대값, 최소값

산포 통계량

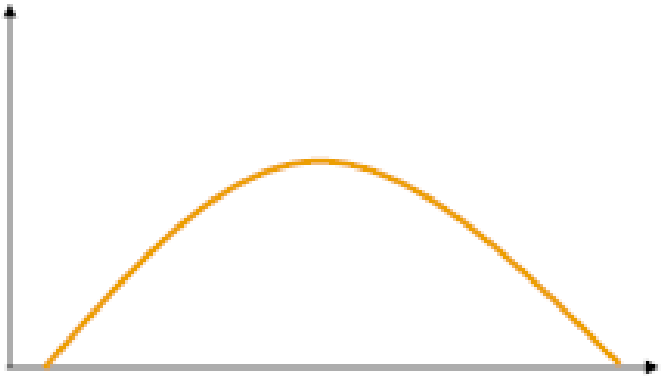
- 분산(variance) : 평균과의 거리를 제곱한 값의 평균
- 표준편차(standard deviation) : 분산의 제곱근
- 범위(range) : 최대값과 최소값의 차이

산포 통계량

- 산포 : 데이터가 **얼마나 퍼져 있는지**를 의미함.



산포가 작은 변수



산포가 큰 변수

- 즉, 산포 통계량이란 데이터의 산포를 나타내는 통계량

산포 통계량 - 분산, 표준편차

- 편차 : 한 샘플이 평균으로부터 떨어진 거리

$$x_i - \mu \ (x_i : i\text{번째 관측치}, \mu : \text{평균})$$

- 분산(variance)
  - 편차의 제곱평균, 편차의 합은 항상 0이 되기 때문에, 제곱을 사용
  - **샘플 데이터는 n-1로 나눔**
  - **자유도가 0(모분산)이면, n으로 나눔**

$$\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}$$

x_i : i번째 관측치
μ : 평균
n : 관측치의 개수

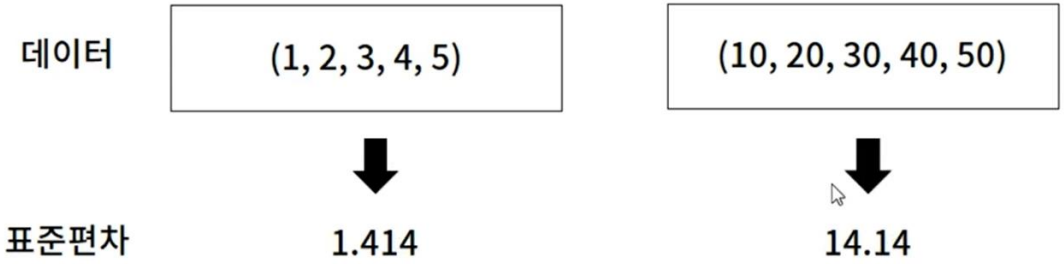
- 표준편차(standard deviation)
  - 분산의 제곱근

$$\sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}}$$

x_i : i번째 관측치
μ : 평균
n : 관측치의 개수

산포 통계량 - 변동계수

- 분산과 표준편차 모두 값의 스케일에 크게 영향을 받아서 상대적인 산포를 보여주는데 부적합함.



- 따라서, **변수를 스케일링한 뒤, 분산 혹은 표준편차**를 구해야 함.
- 만약 모든 데이터가 양수인 경우에는 변동계수(상대 표준편차)를 사용할 수 있음

변동계수 = 표준편차 / 평균

파이썬을 이용한 분산, 표준편차, 변동계수 계산

구분	구현 함수	비고
분산	• numpy.var(x, ddof) • numpy.array(x).var(ddof) • Series(x).var(ddof)	ddof : 자유도
표준편차	• numpy.std(x, ddof) • numpy.array(x, ddof)std() • Series.std(x, ddof)	
변동계수 계산	• numpy.std(x, ddof) / numpy.mean(x) • scipy.stats.variation(x)	

파이썬을 이용한 분산, 표준편차, 변동계수 계산

- 02. 통계학_산포 통계량 실습.ipynb

산포 통계량 - 스케일링(Scaling)

- 둘 이상의 변수의 값을 상대적으로 비교할 때 사용
- 예) 영어 점수 = 85점인 학생, 수학 점수 = 75점인 학생 누가 더 잘 했나?
  - 국어 점수 변수와 수학 점수 변수의 분포가 다르기 때문에 정확히 비교가 힘듬
  - 국어 점수 평균 = 90점, 수학 점수 평균 = 35점
  - 상대적을 봤을 때 수학점수 75점이 더 잘한 것임.

산포 통계량 - 스케일링(Scaling)

- 상대적으로 비교하기 위해 **각 데이터에 있는 값을 상대적인 값을 갖도록 변환**함

$$\frac{x-\mu}{\sigma}$$

Standard Scaling

$$\frac{x-\min(x)}{\max(x)-\min(x)}$$

Min-max Scaling

- 스케일링은 **변수간 비교 및 머신러닝에서 피처간의 데이터 수준을 맞춰주기 위해 많이 사용**됨

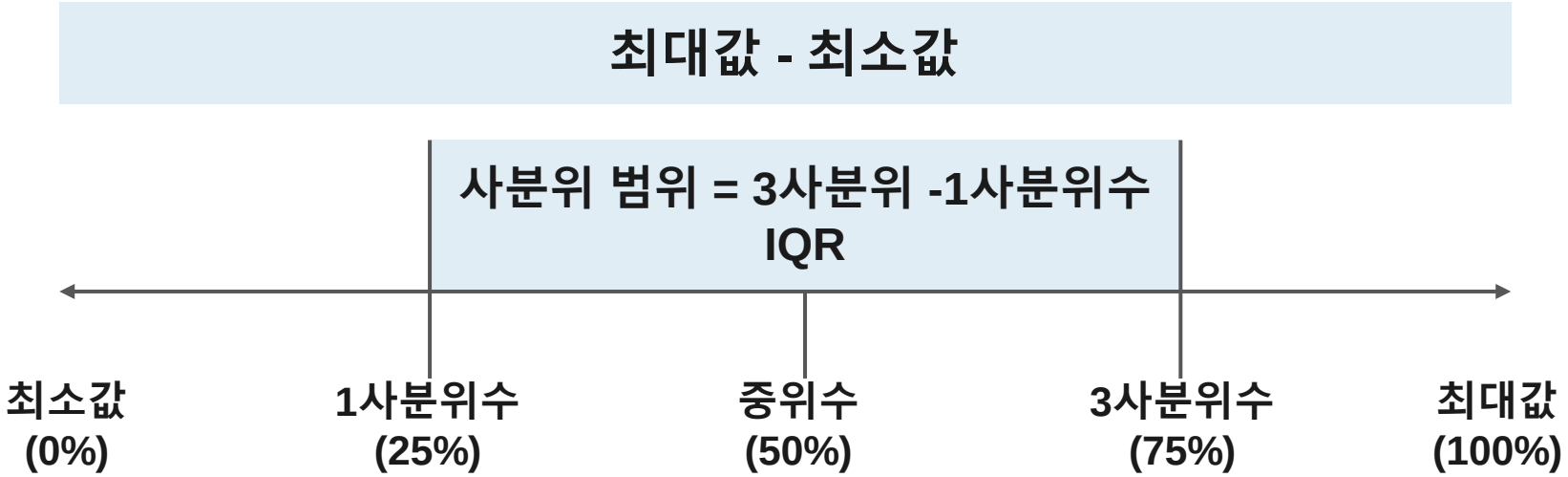
산포 통계량 - 스케일링(Scaling)

구분	구현 함수
Standard Scaling	• (x-x.mean()) / x.std() # x: ndarray • **sklearn.preprocessing.StandardScaler**
Min-max Scaling	• x-x.min() / (x.max() – x.min()) # x: ndarray • **sklearn.preprocessing.MinMaxScaler**

- 실습
  - StandardScaler와 MinMaxScaler의 계산해서 두 피처 간의 결과 확인
  - sklearn 클래스의 메소드 결과 비교해 보기

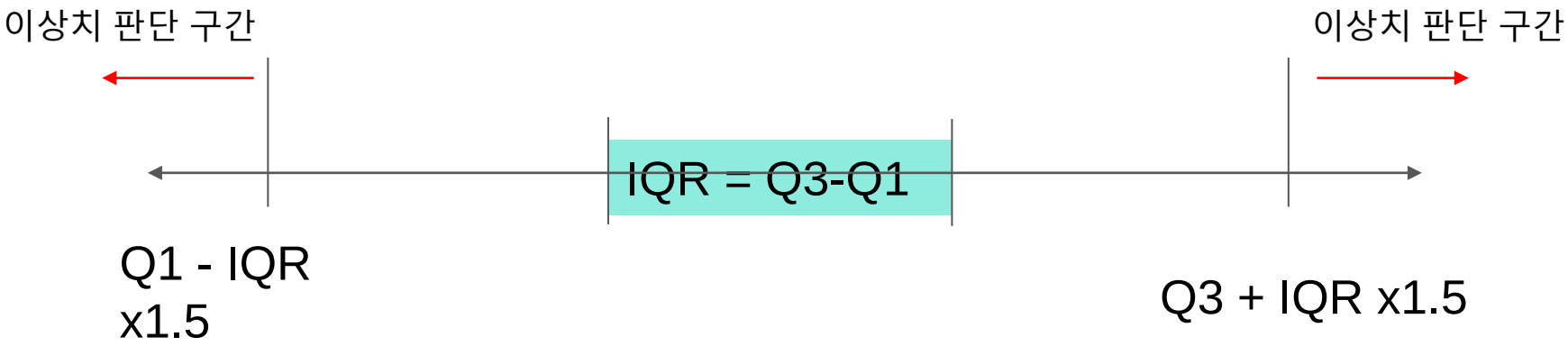
범위와 사분위 범위

- 범위와 사분위 범위는 산포를 나타내는 직관적인 지표
- IQR(Interquartile Range)이라고 하며, 이상치 탐색 시 활용함



IQR Rule

- 변수별로 IQR 규칙을 만족하지 않는 샘플들을 판단하여 삭제하는 방법



파이썬을 이용한 범위 및 사분위 범위 계산

구분	구현 함수
범위	• numpy.ptp(x) • numpy.max(x) – numpy.min(x)
사분위 범위	• numpy.quantile(x, 0.75) – numpy.quantile(x, 0.25) • scipy.stats.iqr(x)

- 산포 통계량 실습
  - 정규분포(np.random.normal) 로 무작위 샘플 만들기(100, 20, size=1000)
  - np.random.normal(0,1,1000) # [정규분포] 평균 0, 표준편차 1, 개수 1000개

기초 통계 분석

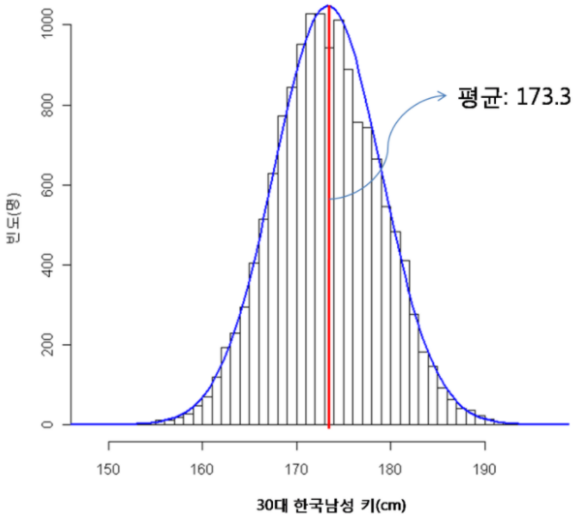
- 통계량의 분류

구분	내용	예시
대표 통계량	데이터의 **중심 및 집중경향**을 나타내는 통계량	평균, 중위값, 최빈값 등
산포 통계량	데이터의 **퍼진 정도**를 나타내는 통계량	분산, 표준편차, 범위
분포 통계량	데이터의 **위치 정보 및 모양**을 나타내는 통계량	사분위수, 최대값, 최소값, 왜도, 첨도

분포 통계량

- 최댓값(maximum), 최솟값(minimum)
- 사분위수(quartiles) : 자료를 4등분한 값
- 왜도(skewness), 첨도(kurtosis) : 데이터 분포의 **대칭 정도**와 데이터의 **이상치(outliers) 존재 여부**를 나타냄
- 정규분포(normal distribution)

정규분포, 표준정규분포 참고 :
https://m.blog.naver.com/istech7/50153739190



분포 통계량 – 최댓값과 최솟값

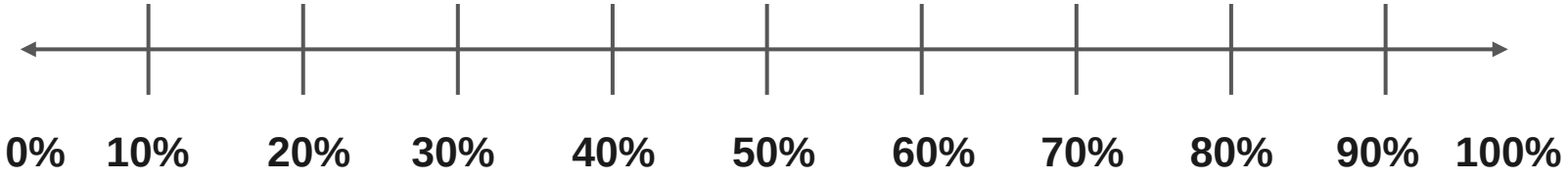
- 범위(range)

  임의의 값을 가지는 데이터 집합에서 최댓값과 최솟값의 차이

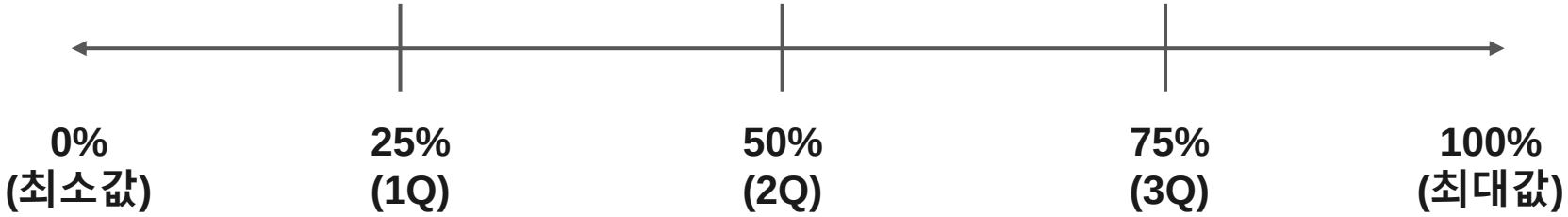
  - Team1 : 8.5
  - Team2 : 27.7

분포 통계량 – 백분위수와 사분위수

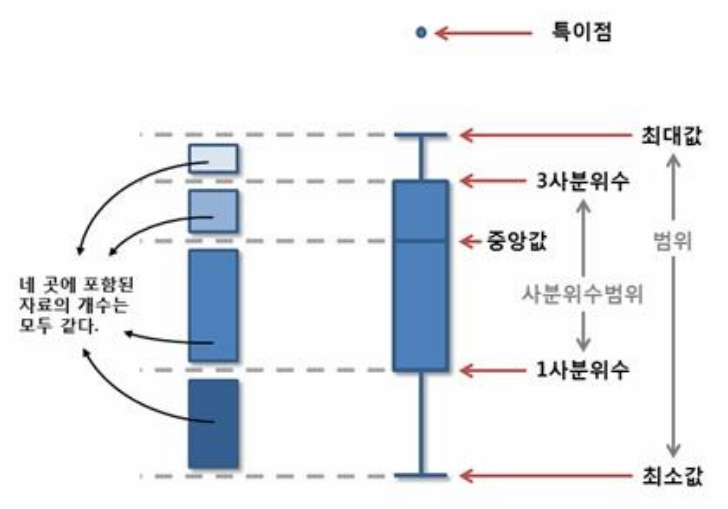
- **백분위수(percentile):** 데이터를 크기 순서대로 **오름차순 정렬**했을 때, **백분율**로 나타낸 특정 위치의 값을 의미 함

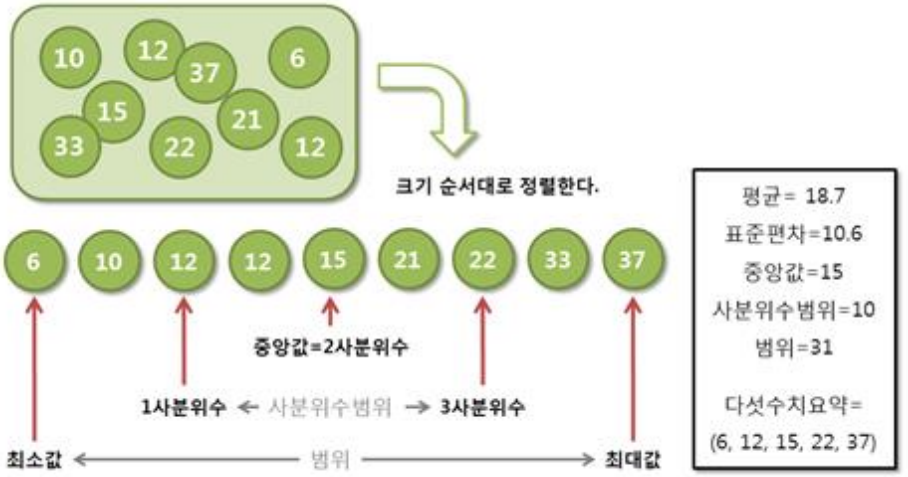


- **사분위수(quantile)**: 데이터를 크기 순서대로 **오름차순 정렬**했을 때, **4등분한 위치**의 값을 의미함



분포 통계량 – 사분위수



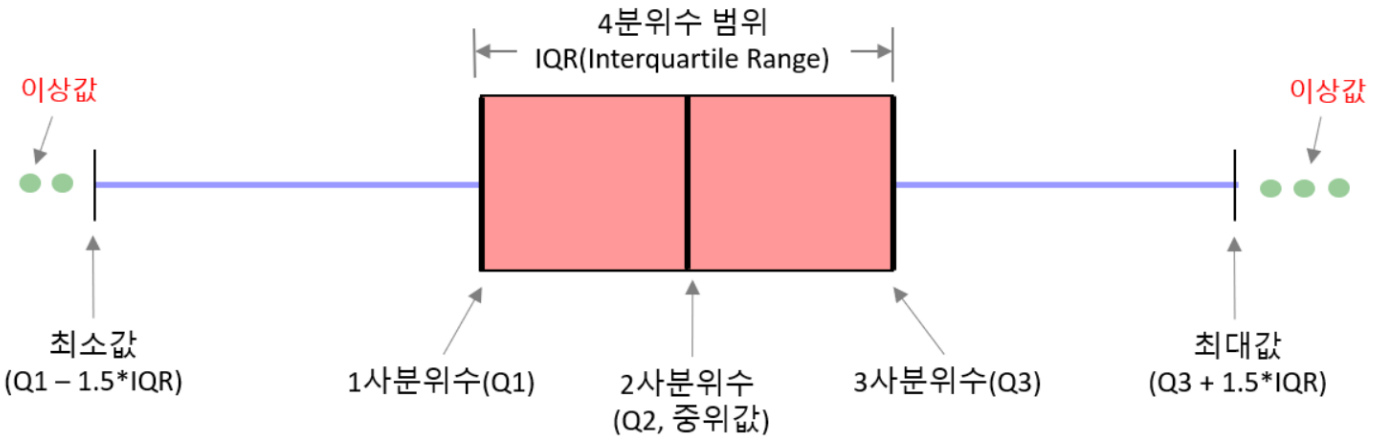


참고: https://t1.daumcdn.net/cfile/tistory/2223A24654298E1415

- 사분위수 : 데이터를 4등분한 값
  - 25%값 : 1사분위수(Q1),
  - 50%값 : 2사분위수(Q2),
  - 75%값 : 3사분위수(Q3)
  - IQR : Interquartile Range, (Q3 – Q1)

분포 통계량 - 사분위수

- Box plot에서 활용
  - 박스 플롯은 관측값의 대략적 분포와 개별적 이상치를 파악할 수 있는 시각화 차트,
  - 한 공간에서 여러 개의 관측값 그룹 시각화

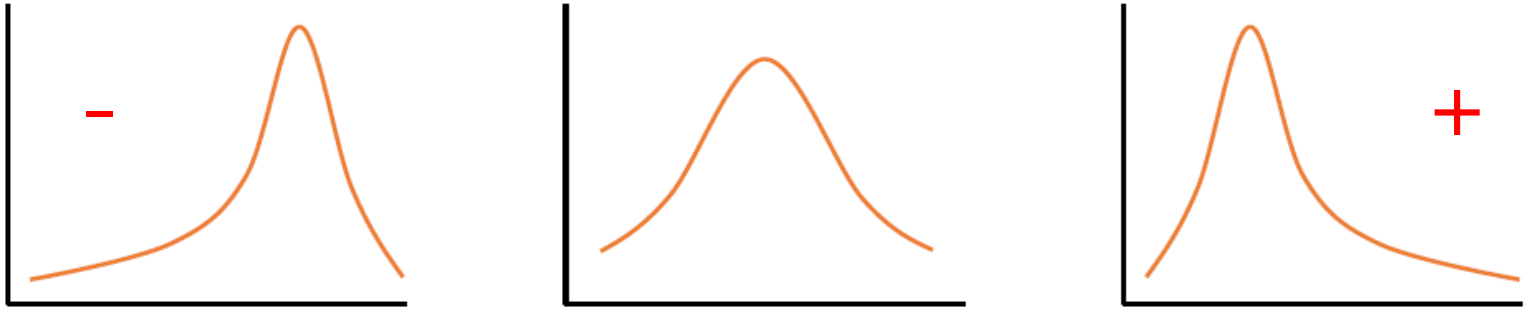


분포 통계량 – 백분위수와 사분위수

구분	구현 함수	비고
백분위수	• numpy.percentile(x, q)	q: 위치(0~100)
사분위수	• numpy.quantile(x, q)	q: 위치(0~1)

분포 통계량 – 왜도

- 왜도(skewness) : 분포의 비대칭도를 나타내는 통계량
- 왜도가 음수면 오른쪽으로 치우친 것을 의미하며,
  양수면 왼쪽으로 치우침을 의미함

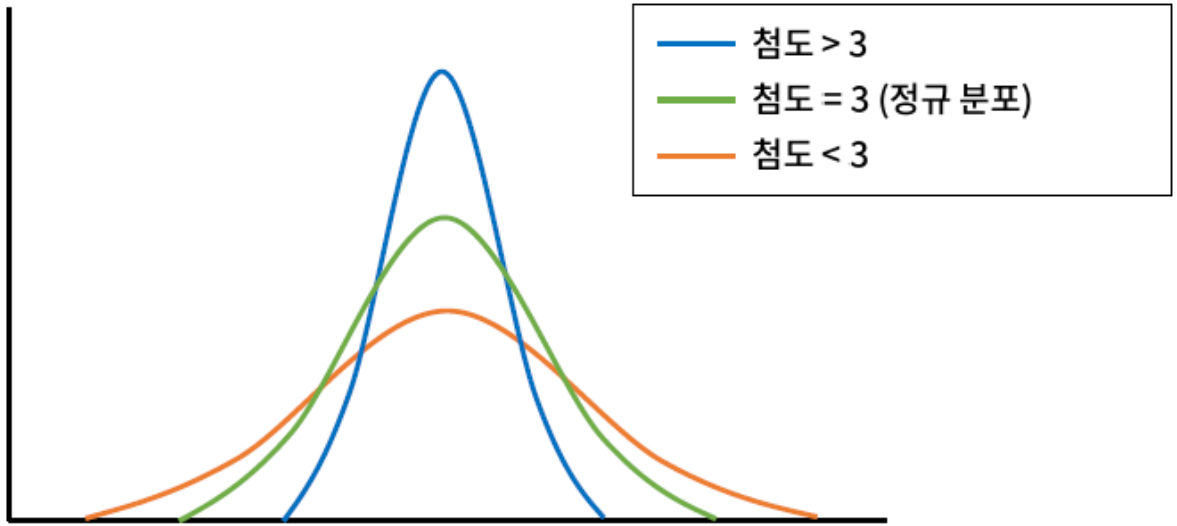


왜도 < 0　　왜도 = 0　　왜도 > 0

- 일반적으로 왜도의 절대값이 1.5이상이면 치우쳤다고 봄

분포 통계량 – 첨도

- 첨도(kurtosis) : 데이터의 분포가 얼마나 뾰족 한지를 의미함 즉, 첨도가 높을 수록 이 변수가 좁은 범위에 많은 값들이 몰려 있다고 할 수 있음.



분포 통계량 – 첨도

구분	구현 함수
왜도	• scipy.stats.skew(x) • Series(x).skew()
첨도	• scipy.stats.kurtosis(x) • Series(x).kurtosis()

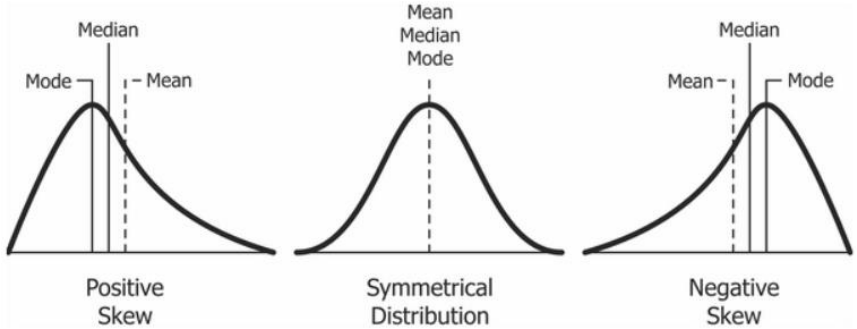
- 실습 : 분포 통계량 – 왜도, 첨도

머신러닝을 위한 통계학 핵심 개념

분포

- 왜도:데이터 분포의 대칭 정도를 나타냄
  - 음의 왜도 : 오른쪽으로 치우침
  - 양의 왜도 : 왼쪽으로 치우침

- 첨도:데이터의 이상치(outliers) 존재여부를 나타냄
  - 음의 첨도 : 납작한 분포
  - 양의 첨도 : 뾰족한 분포



참고: https://brunch.co.kr/@data/13

머신러닝을 위한 통계학 핵심 개념

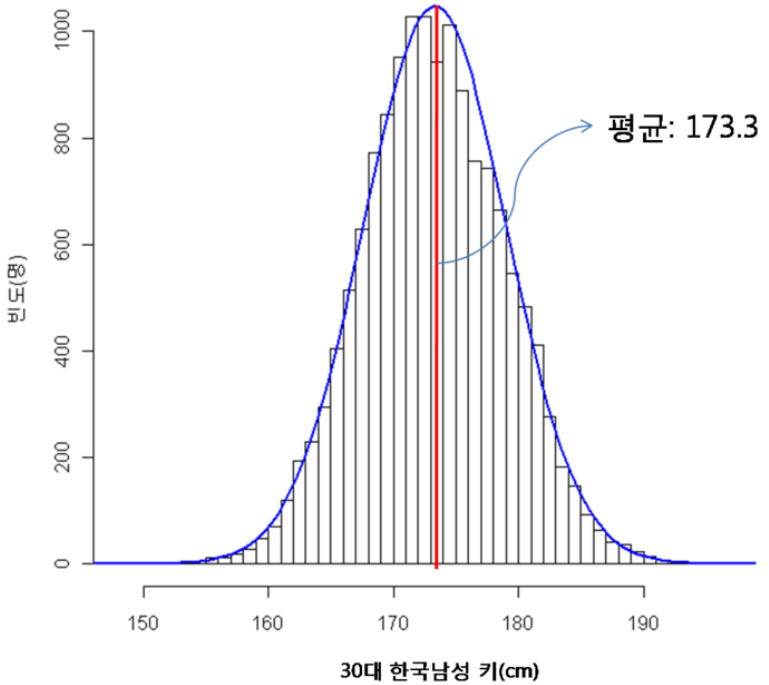
- 분산, 표준편차, 범위, 사분위수, IQR 구하기

```python
import numpy as np
from statistics import variance, stdev

np.random.seed(0)

points = np.random.randint(0, 100, 20)
var = variance(points); print("분산: ", var)
std = stdev(points); print("표준편차: ", np.round(std, 2))
range = np.max(points) - np.min(points); print("범위: ", range)
print("사분위수:")
for val in [0, 25, 50, 75, 100]:
    quantile = np.percentile(points, val)
    print("{}% => {}".format(val, quantile))

q1, q3 = np.percentile(points, [25, 75])
print("IQR: ", q3 - q1)
```

```
분산:  662
표준편차:  25.73
범위:  79
사분위수:
0% => 9.0
25% => 42.75
50% => 64.5
75% => 84.0
100% => 88.0
IQR:  41.25
```

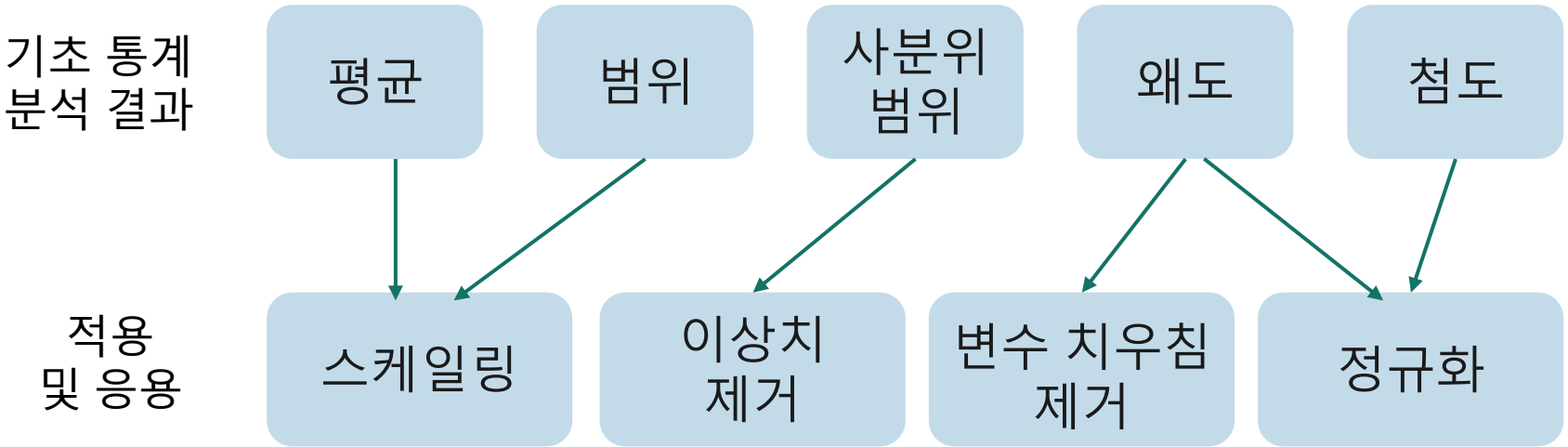
분포 통계량 - 정규분포

- 정규분포(normal distribution)
  - 평균을 중심으로 해서 사건이 중앙에 가장 많이 분포하고 양끝으로 갈수록 희박하게 분포하며, 평균을 축으로 그래프의 양쪽이 정확히 겹쳐짐
  - 표준 정규분포 : 평균 0, 표준편차 1인 정규분포



변수 분포 문제를 확인 후 적용

- 머신러닝에서 각 **변수를 이해**하고, 특별한 분포 **문제가 없는지 확인**하기 위해 기초 통계 분석을 수행함



RAPA 한국전파진흥협회

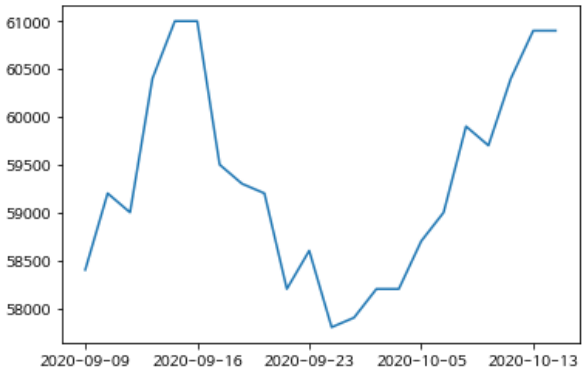
Seaborn 시각화 +

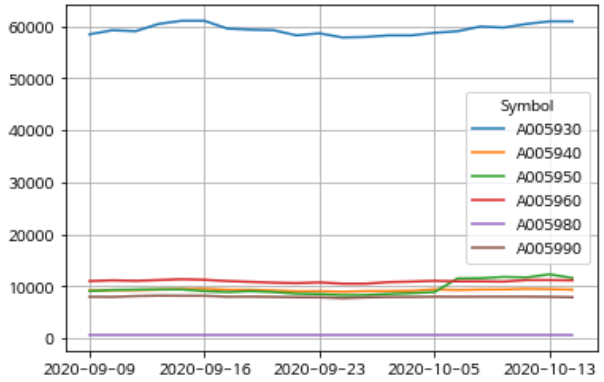
데이터 시각화

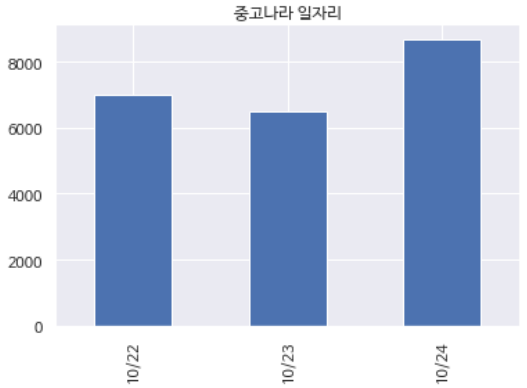
종류	내용
시간시각화	- 분절형과 연속형 - 특정시점 또는 특정시간의 구간 값을 막대 그래프, 누적막대 그래프, 점 그래프 등으로 표현
분포 시각화	- 전체분포와 시간에 따른 분포 - 전체 분포 : 파이차트, 도넛차트, 누적막대, 박스플롯 그래프 - 시간에 따른 분포 : 누적연속 그래프, 누적영역그래프, 선그래프
관계 시각화	- 상관관계, 분포, 비교 - 각 기 다른 변수 사이에서 관계를 시각화 - 스캐터플롯, 행렬, 버블차트 등으로 표현
비교 시각화	- 히트맵, 아웃라이어 찾기(box 플롯)

시간적 시각화(시계열)

- 경제활동 : 국내 총생산, 소비자 물가지수, 수출액
- 물리적활동 : 일일강수량, 기온, 습도
- 인구관련 : 총인구, 농가 수
- 사회생활과 관련 : 교통사고 건수, 범죄발생 수
- 품질 생산관리 등 : 품질 지수, 생산성

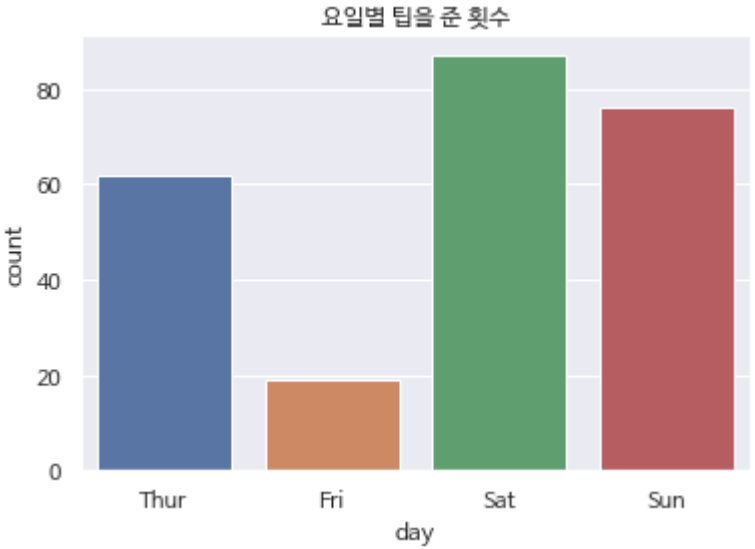


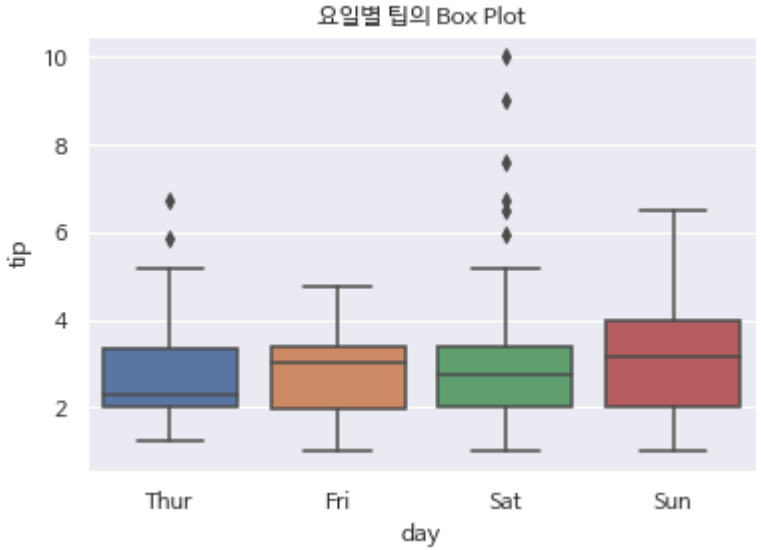




분포 시각화 이해

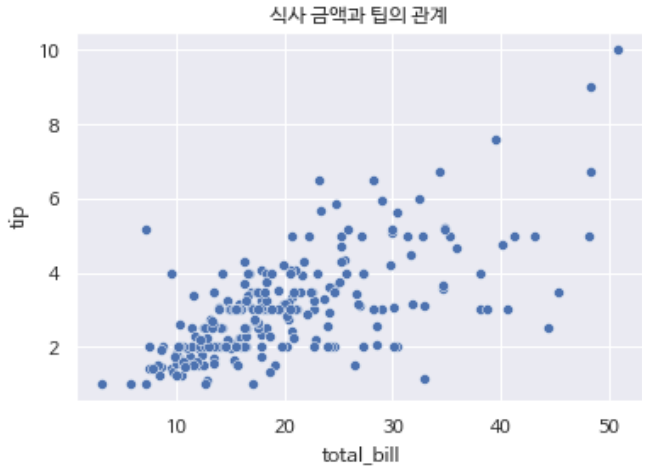
- 분포 데이터 구분 : 샘플 측정 범위에서의 분류
- 분포 데이터 특성 : 최대, 최소, 전체 분포로 나눔
- 전체에 대한 데이터의 양이나 크기를 표현 할 때 사용

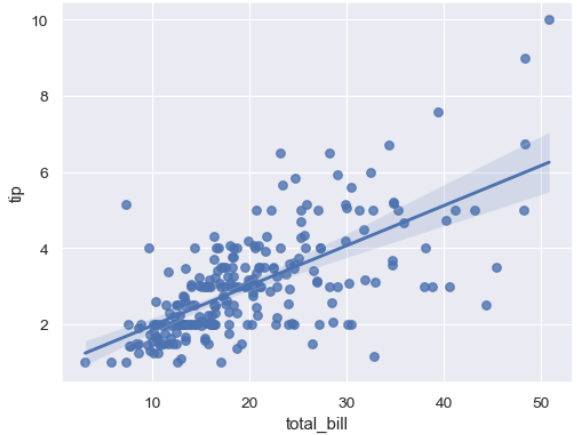


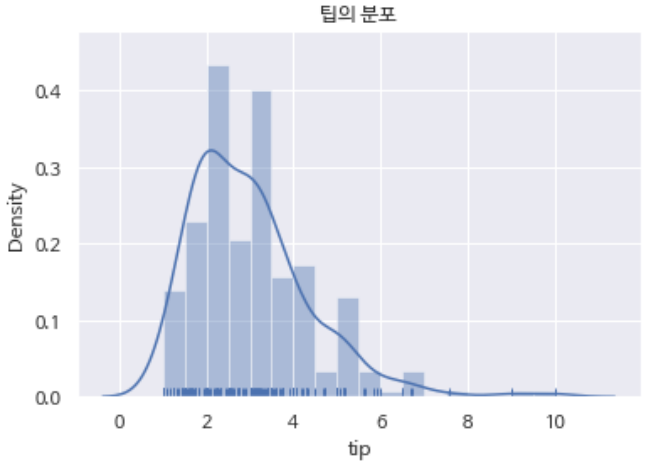


관계 시각화

- 어떤 항목이 다른 항목에 어떤 영향을 주는지 알기 위해 사용
- Scatter 변수 간의 관계를 설명하기 위해
- Histogram : 측정값을 몇 개의 구간으로 나누어 표현한 차트
- Bubble Chart : 스캐터 플롯 + 버블의 크기

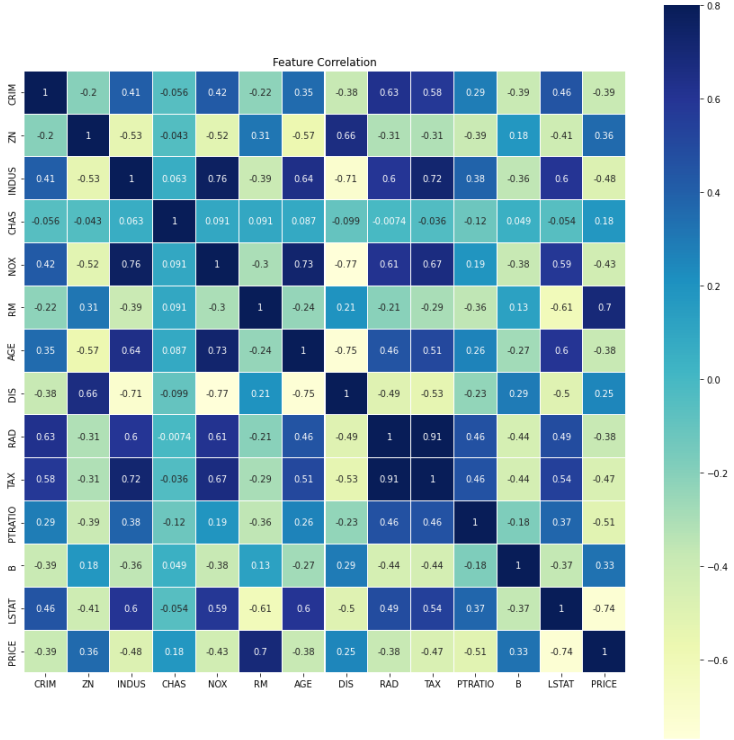






비교 시각화

- 다양한 변수의 특징을 한 번에 비교하여 전체적인 정보 표현이 가능함
- 히트맵 : 색상의 명암으로 값의 크기 표현한 차트



시각화 도구

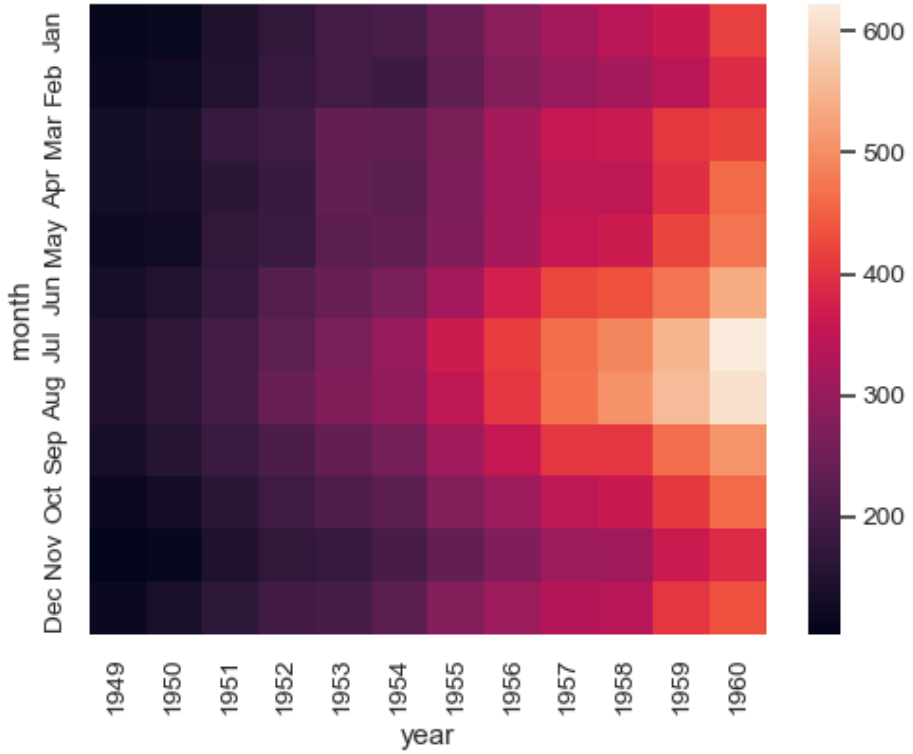
- Matplotlib
  - 그래프를 그리거나, 분포를 보여주는 파이썬 시각화 패키지
  - 연구용으로 많이 쓰인 MATLAB의 코드 스타일 모방
  - 사용하기 좀 복잡함.

- Seaborn
  - matplotlib을 wrapping하여 만든 보다 쉬운 패키지
  - 다양하고 화려한 그래프 제공
  - matplotlib 보다 쉽고 단순한 코드
  - matplotlib의 명령을 사용할 수 있음.

Seaborn 데이터 시각화

- matplotlob과 statsmodel 패키지를 이용하여 만듬
- 데이터의 통계적인 부분을 살펴볼 때, matplotlib에 비해 쉽고 간편함
- 내장 데이터 셋 제공
- 다양한 그래프
  - 데이터의 수를 세는 countplot
  - 5분위 도표 boxplot
  - 두 변수 관계를 점 찍어 그리는 scatterplot
  - 두 변수 관계를 점과 분포로 보기 jointplot
  - Hue 인자로 x축 안에서 botplot 그래프 분리
  - 바이올린 모양의 그래프 viloinplot
  - 데이터의 분포를 그리는 distplot
  - Regression 을 표현하는 regplot

sns.Heapmap()

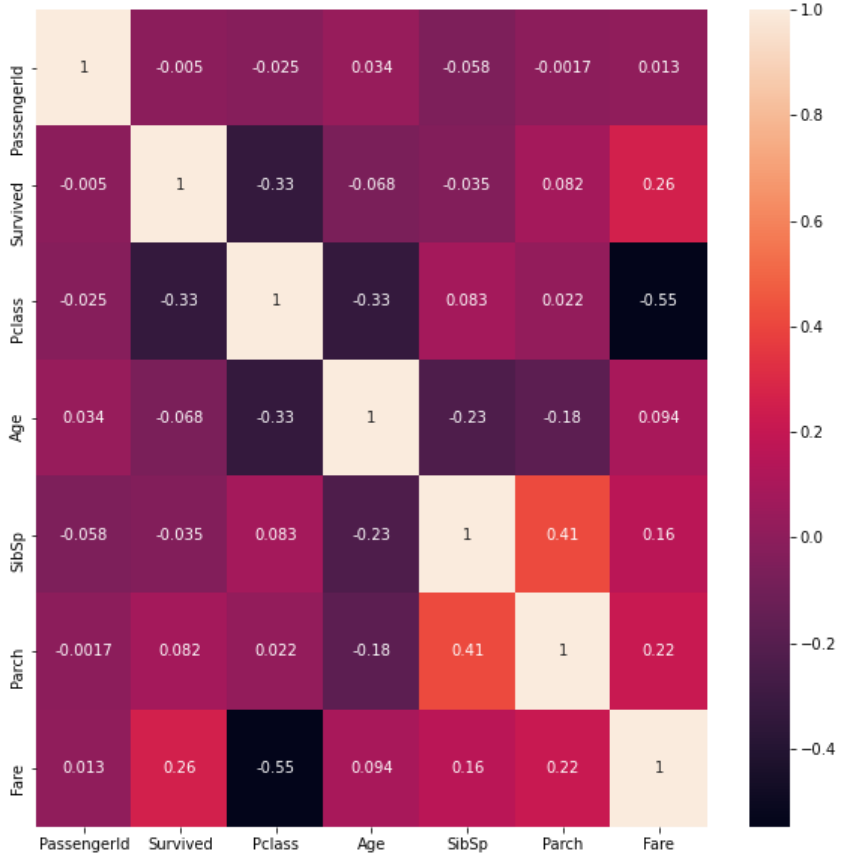


데이터 : 1949~1960년간 승객 수

예) 년도별 월별 승객 수 :

```
flights = sns.load_dataset("flights")
flights = flights.pivot("month", "year", "passengers")
ax = sns.heatmap(flights)
```

Heatmp에 상관분석(Correlation Analysis) 적용



sns.heatmap(df_eda.corr(), annot=True)

- 상관분석(상관관계)는 두 변수 간에 어떤 선형적 또는 비선형적 관계를 갖고 있는지 알아보는 방법
- 1에 가까운 값 : 두 변수들 간의 양의 상관관계가 있음.
- 0에 가까운 값 : 두 변수들 간의 상관관계가 없음.
- -1에 가까운 값 : 두 변수들 간의 음의 상관관계가 있음.

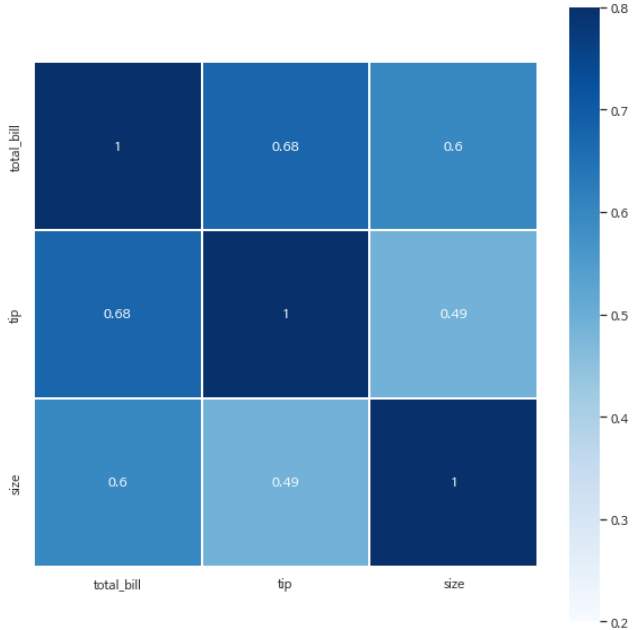
Seaborn heatmap 그리기

- 컬럼 간의 상관관계를 만들어 주는 함수

  corr = pandas.corr()

- searborn의 heatmap()으로 시각화

```
sns.heatmap(corr,   #데이터
    vmin = 0.2,       #최소값 지정(default -1)
    vmax = 0.8,       #최댓값 지정(default 1)
    cbar = True,      #corr color bar 표시
    linewidths=0.01, #cell사이에 선 넣기
    annot = True,     #cell 에 값 표시
    cmap = 'Blues'     #color map 지정
)
```



공식문서 참조:
https://seaborn.pydata.org/generated/seaborn.heatmap.html?highlight=heatmap#seaborn.heatmap

[실습]Seaborn 데이터 시각화

- seaborn 실습 하기
- 파일 : 데이터_시각화_seaborn_실습.ipynb