

데이터분석입문

Lecture 06. 서울 기온 데이터 시각화하기

동양미래대학교
인공지능소프트웨어학과
강 환수

- ❖ 01. 크리스마스의 기온 변화를 그래프로 그리기
- ❖ 02. 기온 데이터를 다양하게 시각화하기

01. 크리스마스의 기온 변화를 그래프로 그리기

02. 기온 데이터를 다양하게 시각화하기

❖ ① 데이터에 질문하기 (1/2)

- 매년 크리스마스의 최고 기온을 그래프로 그린다면 어떤 모양일까요?
- 데이터를 읽어와서 최고 기온 데이터를 출력합니다.

```
import csv

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)

header = next(data)

for row in data:
    print(row[4])

f.close()
```

❖ ① 데이터에 질문하기 (2/2)

- 최고 기온 데이터를 리스트에 저장합니다.

```
import csv

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)

header = next(data)

result = [] # 최고 기온 데이터를 저장할 리스트 생성

for row in data:
    if row[4] != '': # 최고 기온 데이터가 존재한다면
        result.append(float(row[4])) # 리스트 result에 최고 기온 값을 실수형 (float)으로 추가

f.close()
print(len(result))
```

❖ ② 데이터 시각화하기 (1/2)

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)

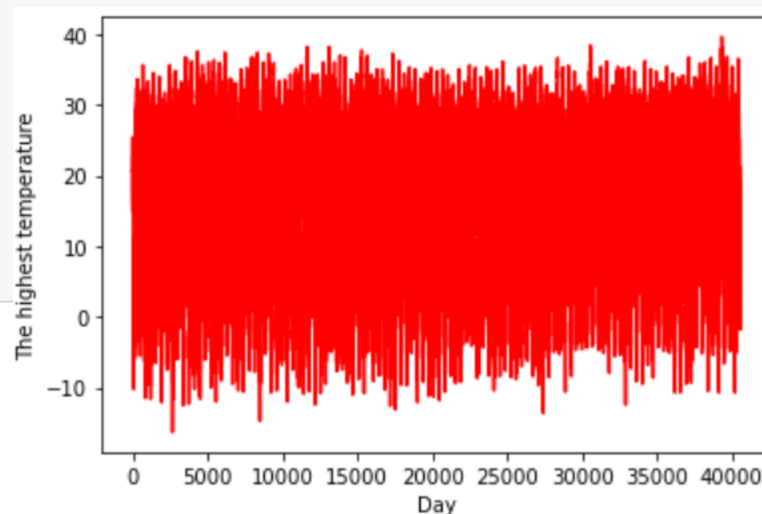
header = next(data)

result = [] # 최고 기온 데이터를 저장할 리스트 생성

for row in data:
    if row[4] != '': # 최고 기온 데이터가 존재한다면
        result.append(float(row[4])) # 리스트 result에 최고 기온 값을 실수형 (float)으로 추가

f.close()
# print(len(result))

plt.plot(result, 'r')
plt.xlabel('Day')
plt.ylabel('The highest temperature')
plt.show()
```



그래프가 뭉쳐 있어서 가시성이 떨어지네요.

❖ ② 데이터 시각화하기 (2/2)

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)

header = next(data)

result = [] # 최고 기온 데이터를 저장할 리스트 생성

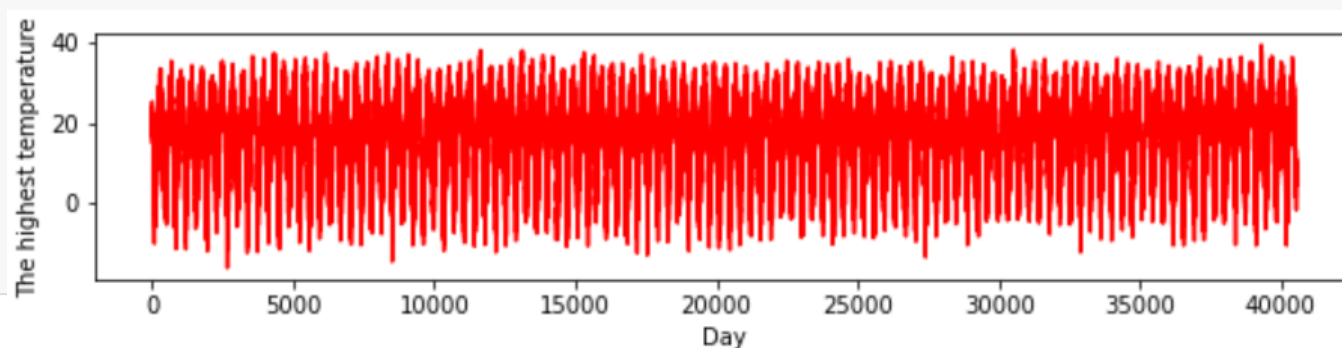
for row in data:
    if row[4] != '': # 최고 기온 데이터가 존재한다면
        result.append(float(row[4])) # 리스트 result에 최고 기온 값을 실수형 (float)으로 추가

f.close()
# print(len(result))

plt.figure(figsize=(10, 2))
plt.plot(result, 'r')
plt.xlabel('Day')
plt.ylabel('The highest temperature')
plt.show()
```

figsize=(가로 길이, 세로 길이)

길이의 단위 인치 (Inch)
1 inch = 2.54 cm



그래프를 크게 해봐도 의미를 파악하는 것이 어렵네요.

❖ ③ 날짜 데이터 추출하기 (1/8)

- 매년 크리스마스 (12월 25일)의 최고 기온 데이터를 추출하기 위해서 우선 '2020-12-25' 같은 형태의 날짜 데이터를 '-'를 기준으로 년, 월, 일로 분리해야 합니다.
- 파이썬이 제공하는 `split()` 함수를 이용하도록 하겠습니다.

```
s = 'Happy New Year!'  
print(s.split())
```

```
['Happy', 'New', 'Year!']
```

`split()` 함수는 사용자가 설정하는 특정 문자가 없다면 기본적으로 공백 문자를 기준으로 문자열을 분리합니다.

❖ ③ 날짜 데이터 추출하기 (2/8)

- `split()` 함수를 이용하여 '2020-12-25' 같은 형태의 날짜 데이터를 '-'를 기준으로 년, 월, 일로 분리해 보겠습니다.

```
date = '2020-12-25'  
print(date.split('-'))
```

```
['2020', '12', '25']
```

- 리스트의 인덱싱 (Indexing) 기능을 활용하여 날짜의 년, 월, 일 정보를 각각 추출해 보겠습니다.

```
date = '2020-12-25'  
  
print(date.split('-')[0])  
print(date.split('-')[1])  
print(date.split('-')[2])
```

```
2020  
12  
25
```

❖ ③ 날짜 데이터 추출하기 (3/8)

- split() 함수를 이용하여 매년 12월의 최고 기온 데이터만 추출하여 그래프로 그려 보겠습니다.

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)

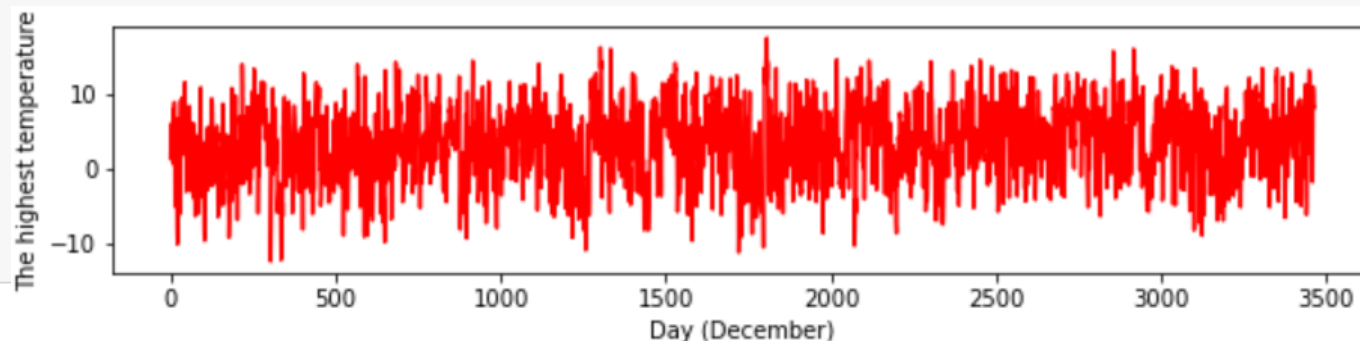
header = next(data)

result = [] # 최고 기온 데이터를 저장할 리스트 생성

for row in data:
    if row[4] != '': # 최고 기온 데이터가 존재한다면
        if row[0].split('-')[1] == '12': # 12월에 해당하는 값이라면
            result.append(float(row[4])) # 리스트 result에 최고 기온 값을 실수형 (float)으로 추가

f.close()

plt.figure(figsize=(10, 2))
plt.plot(result, 'r')
plt.xlabel('Day (December)')
plt.ylabel('The highest temperature')
plt.show()
```



❖ ③ 날짜 데이터 추출하기 (4/8)

- split() 함수를 이용하여 매년 12월 25일의 최고 기온 데이터만 추출하여 그래프로 그려 보겠습니다.

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)

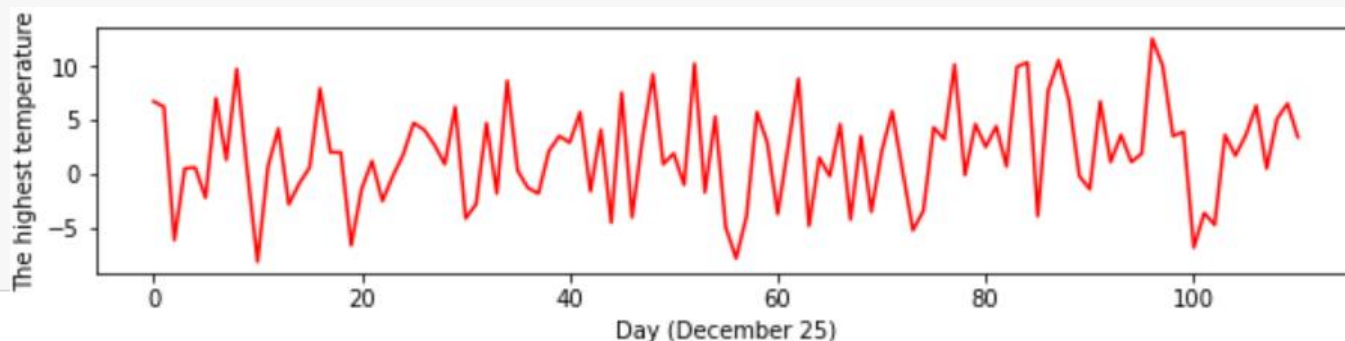
header = next(data)

result = [] # 최고 기온 데이터를 저장할 리스트 생성

for row in data:
    if row[4] != '': # 최고 기온 데이터가 존재한다면
        if (row[0].split('-')[1] == '12') and (row[0].split('-')[2] == '25'):
            result.append(float(row[4])) # 리스트 result에 최고 기온 값을 실수형 (float)으로 추가

f.close()

plt.figure(figsize=(10, 2))
plt.plot(result, 'r')
plt.xlabel('Day (December 25)')
plt.ylabel('The highest temperature')
plt.show()
```



❖ ③ 날짜 데이터 추출하기 (5/8)

- 매년 12월 25일의 평균, 최저, 최고 기온 데이터로 그래프를 그려 보겠습니다.

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)

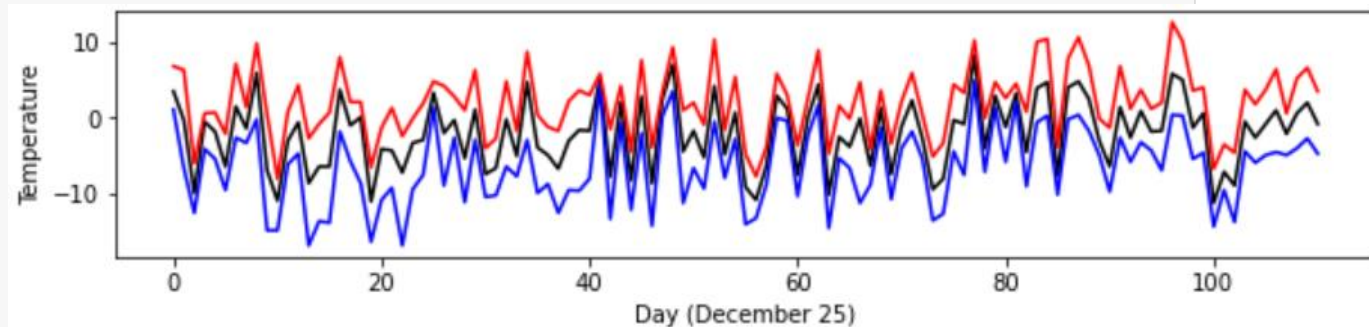
header = next(data)

mean = [] # 평균 기온 데이터를 저장할 리스트 생성
low = [] # 최저 기온 데이터를 저장할 리스트 생성
high = [] # 최고 기온 데이터를 저장할 리스트 생성

for row in data:
    if (row[2] != '') and (row[3] != '') and (row[4] != ''):
        if (row[0].split('-')[1] == '12') and (row[0].split('-')[2] == '25'):
            mean.append(float(row[2]))
            low.append(float(row[3]))
            high.append(float(row[4]))

f.close()

plt.figure(figsize=(10, 2))
plt.plot(mean, 'k')
plt.plot(low, 'b')
plt.plot(high, 'r')
plt.xlabel('Day (December 25)')
plt.ylabel('Temperature')
plt.show()
```



❖ ③ 날짜 데이터 추출하기 (6/8)

- 그래프를 다듬어 보겠습니다.

◆ 한글 폰트 사용하기 (Windows 기준)

- Malgun Gothic은 “맑은 고딕”입니다.
- 만약 macOS 운영체제를 사용하고 있다면 “AppleGothic”이라고 쓰세요.

```
plt.rc('font', family='Malgun Gothic')  
plt.title('크리스마스의 기온 변화 그래프')
```

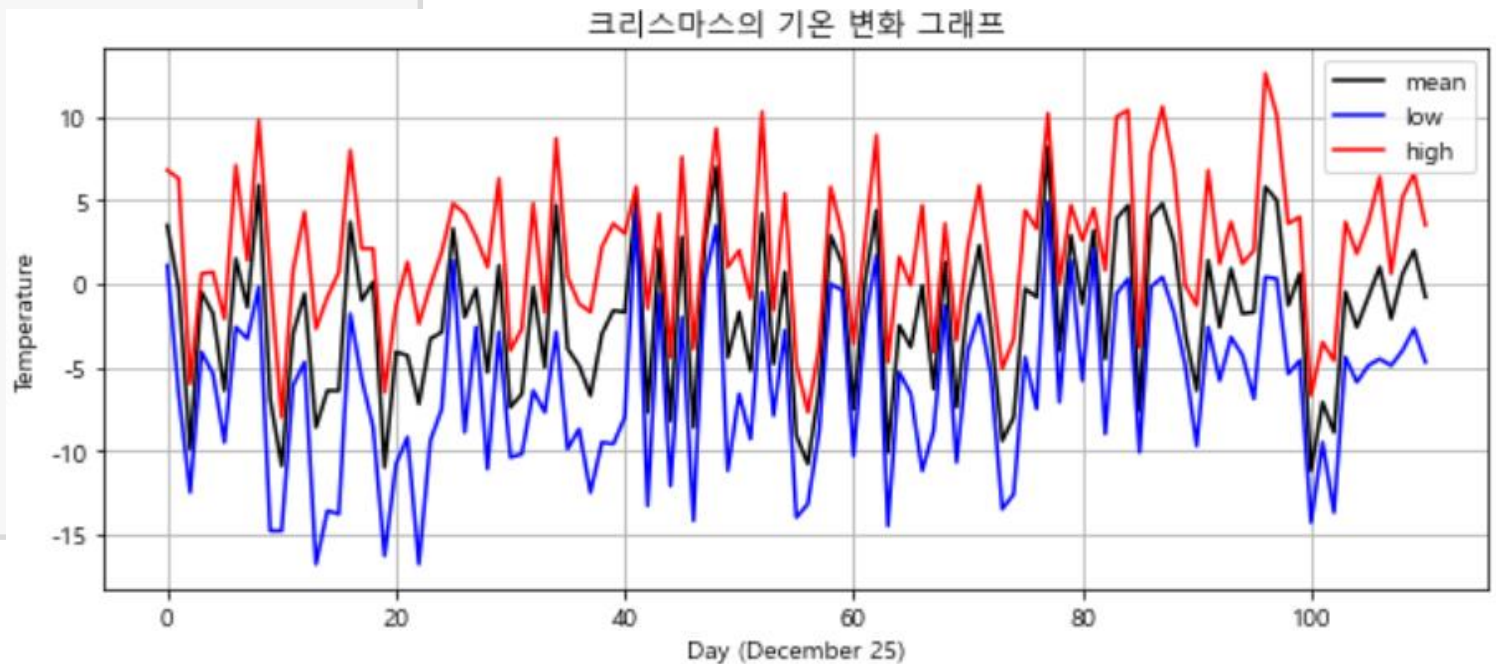
◆ 한글 폰트 사용시 마이너스 부호 표현하기

```
plt.rcParams['axes.unicode_minus'] = False
```

❖ ③ 날짜 데이터 추출하기 (7/8)

- 그래프를 다듬어 보겠습니다.

```
plt.figure(figsize=(10, 4))  
plt.rc('font', family='Malgun Gothic')  
plt.rcParams['axes.unicode_minus'] = False  
plt.title('크리스마스의 기온 변화 그래프')  
plt.plot(mean, 'k', label='mean')  
plt.plot(low, 'b', label='low')  
plt.plot(high, 'r', label='high')  
plt.xlabel('Day (December 25)')  
plt.ylabel('Temperature')  
plt.legend()  
plt.grid()  
plt.show()
```



❖ ③ 날짜 데이터 추출하기 (8/8)

● 평균 기온 변화 그래프 그리기

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding = 'cp949')
data = csv.reader(f)

header = next(data)

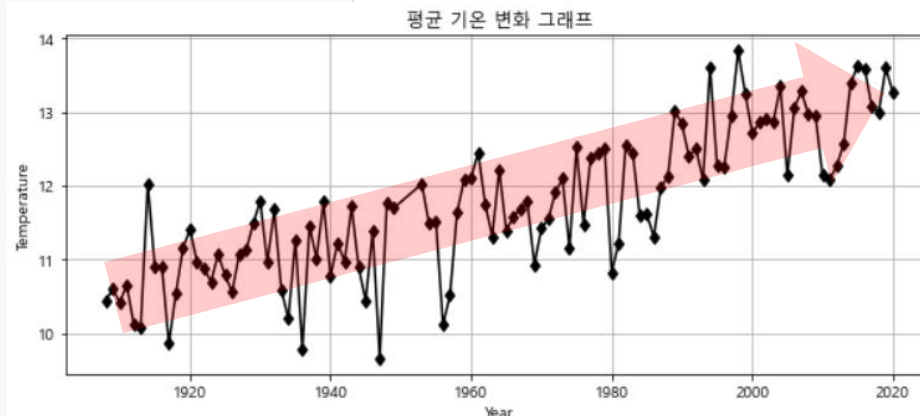
temp_day = []    # 1년 단위로 365일의 일별 평균 기온 데이터를 저장할 리스트 생성
temp_mean = []  # 연도 별로, temp_day에 저장된 평균 기온 데이터의 평균 값을 저장할 리스트 생성
year = []        # 연도 데이터를 저장할 리스트 생성 -> 그래프의 x축 데이터로 활용

for row in data:
    # 1907년도에는 10월부터 기온 데이터가 기록되어 있어서, 평균 기온이 매우 낮음 (분석에서 제외함)
    if int(row[0].split('-')[0]) > 1907:
        if row[2] != '': # 평균 기온 데이터가 결측치가 아니라면 아래를 수행
            # 1월 1일부터 12월 31일까지의 일별 평균 기온 데이터를 저장
            temp_day.append(float(row[2]))
            if (row[0].split('-')[1] == '12') and (row[0].split('-')[2] == '31'):
                year.append(int(row[0].split('-')[0])) # 연도 데이터를 저장
                # 12월 31일이 되면 지금까지 temp_day에 저장했던 평균 기온 데이터의 평균 값을 계산
                temp_mean.append(sum(temp_day) / len(temp_day))
                temp_day = [] # 다음 년도의 평균 기온 데이터를 1월 1일부터 새롭게 저장해야 하므로
                # temp_day 리스트를 [] 비어있는 리스트로 만들어 줌 (이를 "초기화"라고 부름)

f.close()

plt.figure(figsize=(10, 4))
plt.rc('font', family='Malgun Gothic')
plt.title('평균 기온 변화 그래프')
plt.plot(year, temp_mean, color='k', marker='d')
plt.xlabel('Year')
plt.ylabel('Average Temperature')
plt.grid()
plt.show()
```

서울의 평균 기온이
상승하고 있음을 알 수 있습니다.

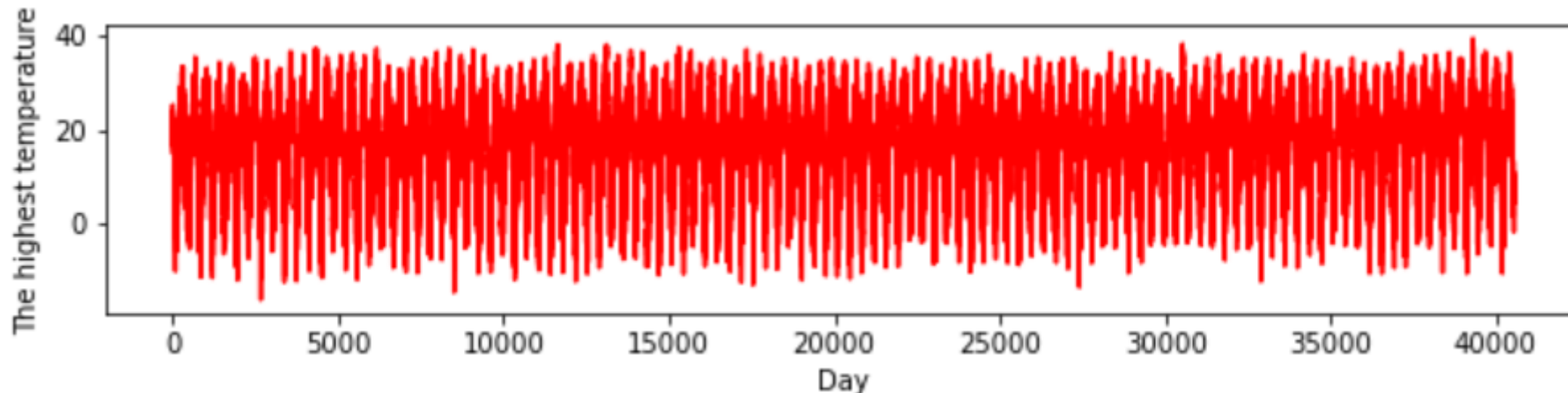


02. 기온 데이터를 다양하게 시각화하기

01. 크리스마스의 기온 변화를 그래프로 그리기

❖ ① 데이터에 질문하기

- 아래 그래프는 매년 크리스마스의 최고 기온 데이터를 추출하여 그린 결과입니다.



이 그래프만 보서는 특별한 정보를 얻을 수가 없습니다.
꺾은선 그래프가 아닌 다른 형태로 시각화 하면 어떨까요?

❖ ③ 기온 데이터를 히스토그램으로 표현하기 (1/3)

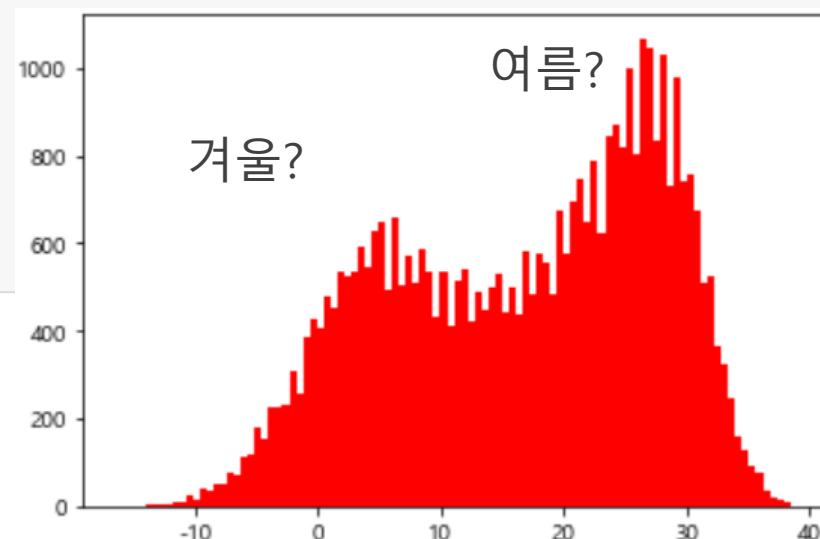
- 최고 기온 데이터를 히스토그램으로 시각화하기

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)
result = []

for row in data:
    if row[4] != '':
        result.append(float(row[4]))

plt.hist(result, bins=100, color='r')
plt.show()
```



❖ ③ 기온 데이터를 히스토그램으로 표현하기 (2/3)

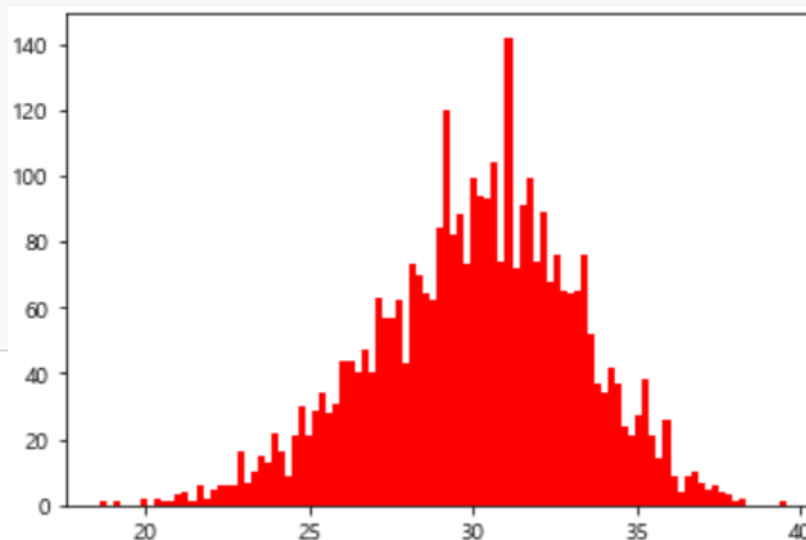
- 8월의 최고 기온 데이터를 히스토그램으로 시각화하기

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)
aug = []

for row in data:
    month = row[0].split('-')[1]
    if row[4] != '':
        if month == '08':
            aug.append(float(row[4]))

plt.hist(aug, bins=100, color='r')
plt.show()
```



❖ ③ 기온 데이터를 히스토그램으로 표현하기 (3/3)

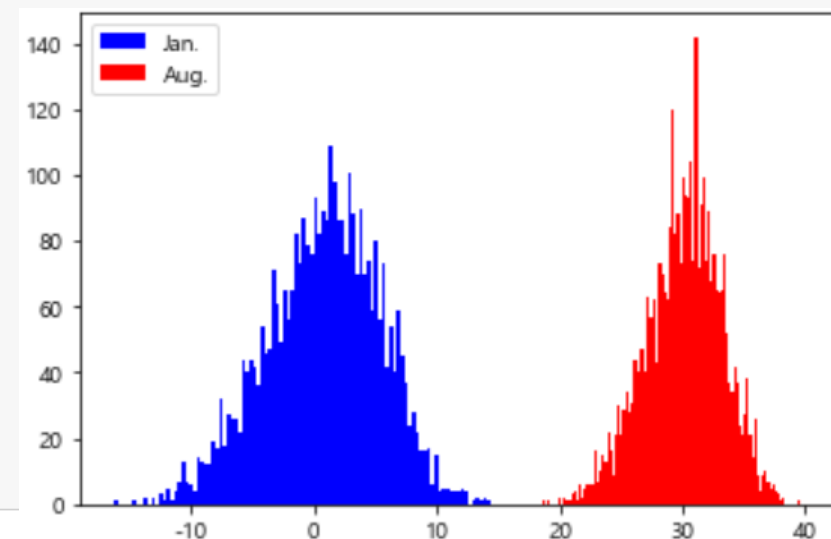
- 1월과 8월의 최고 기온 데이터를 히스토그램으로 시각화하기

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)
jan = []
aug = []

for row in data:
    month = row[0].split('-')[1]
    if row[4] != '':
        if month == '01':
            jan.append(float(row[4]))
        elif month == '08':
            aug.append(float(row[4]))

plt.hist(jan, bins=100, color='b', label='Jan.')
plt.hist(aug, bins=100, color='r', label='Aug.')
plt.legend()
plt.show()
```



❖ ④ 기온 데이터를 상자 그림으로 표현하기 (1/6)

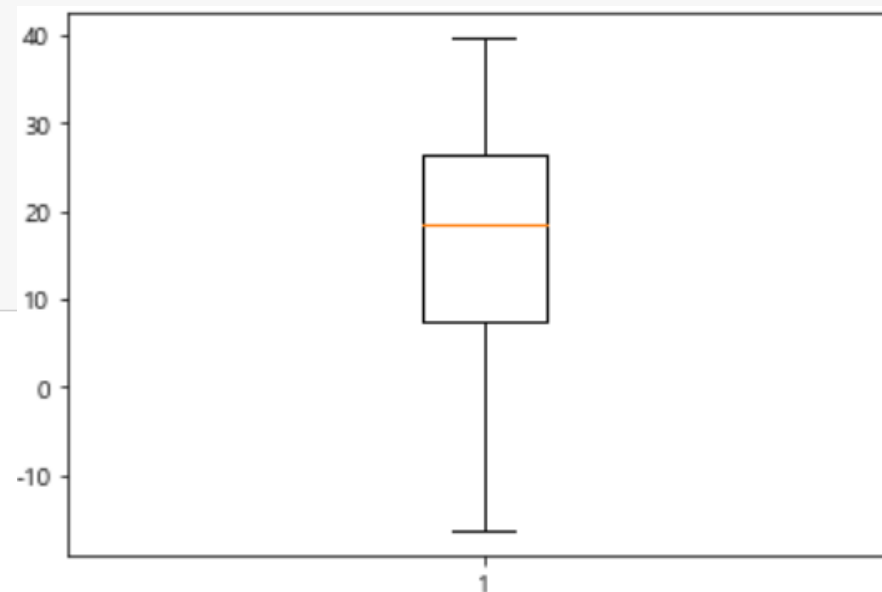
- 최고 기온 데이터를 상자 그림으로 시각화하기

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)
result = []

for row in data:
    if row[4] != '':
        result.append(float(row[4]))

plt.boxplot(result)
plt.show()
```



❖ ④ 기온 데이터를 상자 그림으로 표현하기 (2/6)

- 1월과 8월의 최고 기온 데이터를 상자 그림으로 시각화하기

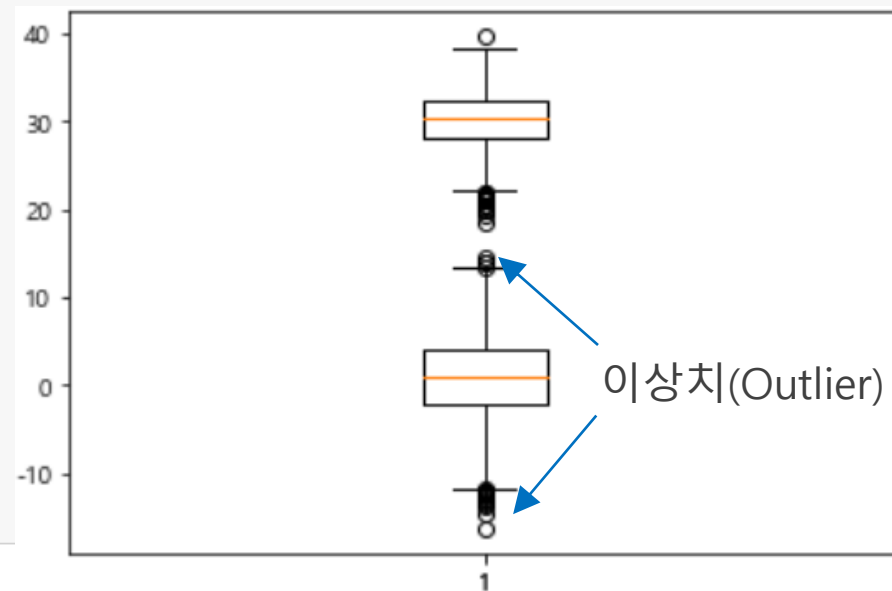
```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)
jan = []
aug = []

for row in data:
    month = row[0].split('-')[1]
    if row[4] != '':
        if month == '01':
            jan.append(float(row[4]))
        elif month == '08':
            aug.append(float(row[4]))

plt.boxplot(jan)
plt.boxplot(aug)
plt.show()
```

1월과 8월 데이터를
분리하여 표현할 순 없을까요?



❖ ④ 기온 데이터를 상자 그림으로 표현하기 (3/6)

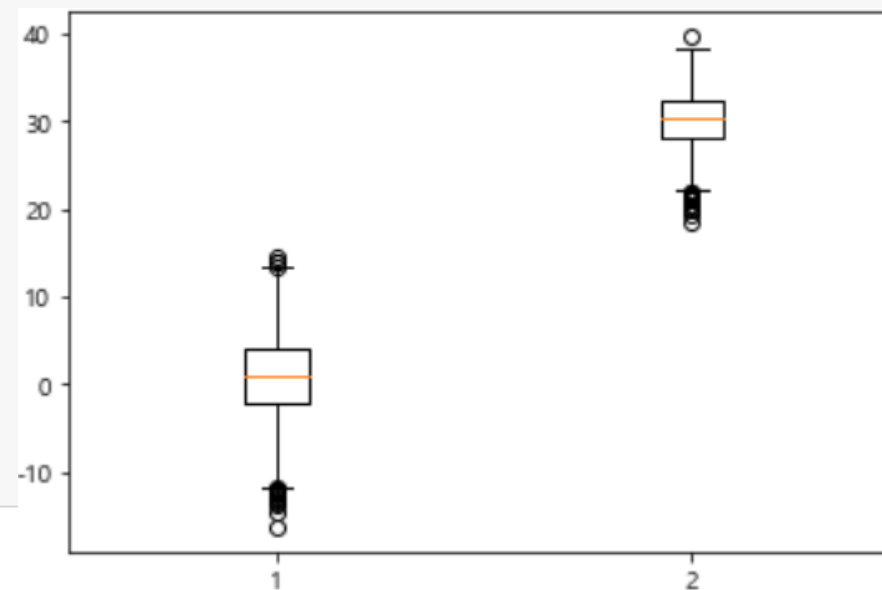
- 1월과 8월의 최고 기온 데이터를 상자 그림으로 시각화하기

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)
jan = []
aug = []

for row in data:
    month = row[0].split('-')[1]
    if row[4] != '':
        if month == '01':
            jan.append(float(row[4]))
        elif month == '08':
            aug.append(float(row[4]))

plt.boxplot([jan, aug])
plt.show()
```



❖ ④ 기온 데이터를 상자 그림으로 표현하기 (4/6)

- 1월부터 12월까지의 최고 기온 데이터를 상자 그림으로 시각화하기

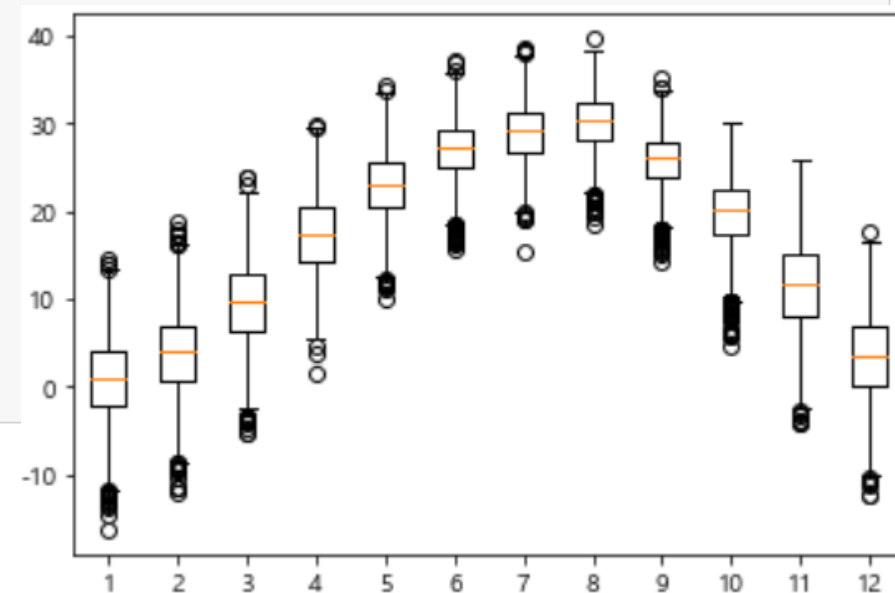
```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)

# 월별 데이터를 저장할 리스트 month (12개 생성)
month = [[] for _ in range(12)]

for row in data:
    if row[4] != '':
        # 월과 같은 번호의 인덱스에 월별 데이터 저장 (예: 1월 -> month[0])
        month[int(row[0].split('-')[1])-1].append(float(row[4]))

plt.boxplot(month)
plt.show()
```



❖ ④ 기온 데이터를 상자 그림으로 표현하기 (5/6)

- 1월 일별 기온 데이터를 상자 그림으로 시각화하기

```
import csv
import matplotlib.pyplot as plt

f = open('seoul.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)

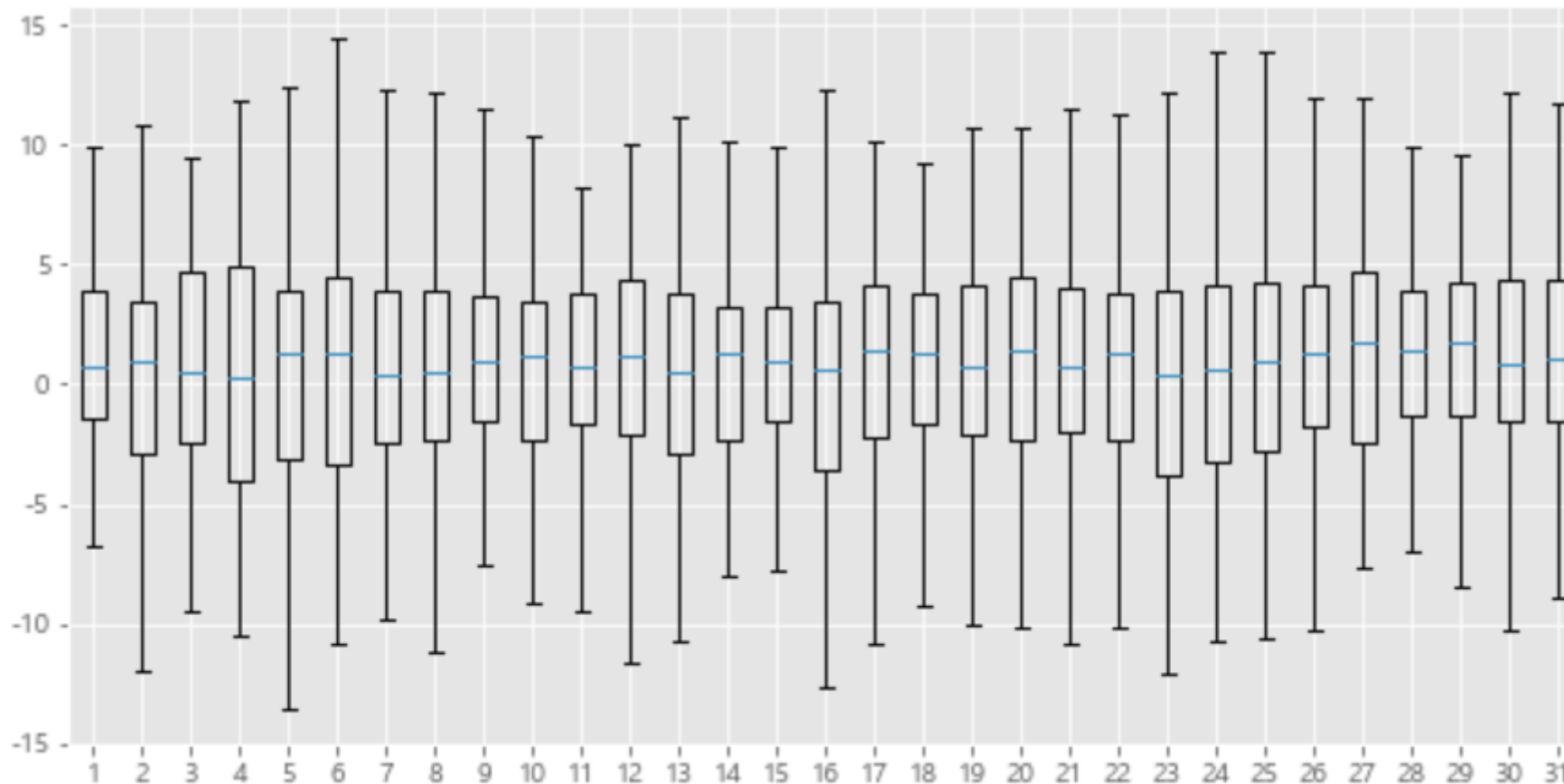
day = []
for i in range(31):
    day.append([])

for row in data:
    if row[4] != '':
        if row[0].split('-')[1] == '01':
            # 일과 같은 번호의 인덱스에 일별 데이터 저장 (예: 1일 -> day[0])
            day[int(row[0].split('-')[2])-1].append(float(row[4]))

plt.style.use('ggplot')          # 그래프 스타일 지정
plt.figure(figsize=(10, 5))     # 그래프 크기 수정
plt.boxplot(day, showfliers=False) # 이상치 (outlier) 표시 생략
plt.show()
```

❖ ④ 기온 데이터를 상자 그림으로 표현하기 (6/6)

- 1월 일별 기온 데이터를 상자 그림으로 시각화하기



- ❖ 01. 크리스마스의 기온 변화를 그래프로 그리기
- ❖ 02. 기온 데이터를 다양하게 시각화하기

THANK YOU!

Q & A

- Name: 강환수
- Office: 동양미래대학교 2호관 706호 (02-2610-1941)
- E-mail: hsknag@dongyang.ac.kr
- Homepage: <https://github.com/ai7dnn/2023-DA>