
Introduction To Data Analysis
데이터분석입문
Lecture 26. 연령별 인구구조 유사지역 분석
인공지능소프트웨어학과 강환수 교수

학습개요

- 데이터분석을 위한 문제 정의
- 연령별 인구구조가 유사한 지역을 찾기 위한 알고리즘
- 입력 받은 지역과 연령별 인구구조가 가장 유사한 지역을 찾기

학습목표

- 데이터분석을 위한 문제를 정의할 수 있다.
- 연령별 인구구조가 유사한 지역을 찾기 위한 알고리즘을 이해할 수 있다.
- 패키지 numpy의 배열 array를 사용해 문자열을 정수로 변환할 수 있다.
- 입력 받은 지역과 연령별 인구구조가 가장 유사한 지역을 찾기 위한 코드를 구현할 수 있다.

LESSON 01

명확한 문제 정의

데이터 살펴보며 질문하기 (1/2)

데이터를 살펴보는 방법

- 엑셀 같은 스프레드시트 프로그램 등을 활용해 데이터 자세히 살펴보기
- 데이터가 담고 있는 내용(또는 담고 있지 않은 내용)
- 데이터가 기록된 기간은 언제부터 언제까지인지
- 어떤 형태로 시각화해보면 어떤 정보들을 알 수 있을지 생각해보기
- 데이터를 보며 궁금한 내용들 자유롭게 질문하기

데이터 살펴보며 질문하기 (2/2)

age.csv의 인구 데이터를 보며 떠오른 질문들 예시

- 전국에서 영유아들이 가장 많이 사는 지역은 어디일까?
- 보통 학군이 좋다고 알려진 지역에는 청소년들이 많이 살까?
- 광역시 데이터를 10년 단위로 살펴보면 청년 비율이 줄고 있다는 사실을 알 수 있을까?
- 서울에서 지난 5년간 인구가 가장 많이 증가한 구는 어디일까?
- 우리 동네의 인구 구조와 가장 비슷한 동네는 어디일까?

영유아들? 청소년들? 청년비율? 인구 구조와 가장 비슷한 동네?

- 문제를 명확히 정의할 필요가 있어 보입니다.

질문을 명확한 문제로 정의하기

예를 들어 아래와 같이 문제를 좀 더 명확하게 정의할 수 있습니다.

- 전국에서 영유아들이 가장 많이 사는 지역은 어디일까?
  - ✚ 전국에 있는 읍면동 중 만 0세 이상 6세 이하의 인구 비율이 높은 상위 10곳은?
- 우리 동네의 인구 구조와 가장 비슷한 동네는 어디일까?
  - ✚ 전국에서 우리 동네의 연령별 인구 구조와 가장 형태가 비슷한 지역은 어디일까?

위 문제를 해결하기 위해서,
알고리즘을 어떻게 설계해야 할까요?

알고리즘 설계하기

[문제] 전국에서 우리 동네의 연령별 인구 구조와 가장 형태가 비슷한 지역은 어디일까?

- Step 1) 데이터를 읽어온다.
- Step 2) 궁금한 지역의 이름을 입력 받는다.
- Step 3) 궁금한 지역의 인구 구조를 저장한다.
- Step 4) 궁금한 지역의 인구 구조와 가장 비슷한 인구 구조를 가진 지역을 찾는다.
- Step 5) 가장 비슷한 곳의 인구 구조와 궁금한 지역의 인구 구조를 시각화한다.

numpy를 활용하여 궁금한 지역의 인구 데이터를 출력하는
코드를 작성해 보겠습니다!

LESSON 02

알고리즘 코드 표현

데이터 읽어 오기

```
import csv

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

header = next(data)  # 헤더 제거
print(header)

for row in data :
    print(row)
    break

f.close()
```

Step 1) 데이터 읽어 오기(실행결과)

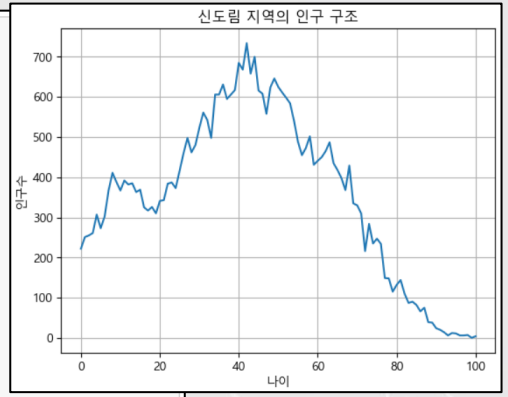
```
['행정구역', '2023년09월_계_총인구수', '2023년09월_계_연령구간인구수', '2023년09월_계_0세', '2023년09월_계_1세', '2023년09월_계_2세', '2023년09월_계_3세', '2023년09월_계_4세', '2023년09월_계_5세', '2023년09월_계_6세', '2023년09월_계_7세', '2023년09월_계_8세', '2023년09월_계_9세', '2023년09월_계_10세', '2023년09월_계_11세', '2023년09월_계_12세', '2023년09월_계_13세', '2023년09월_계_14세', '2023년09월_계_15세', '2023년09월_계_16세', '2023년09월_계_17세', '2023년09월_계_18세', '2023년09월_계_19세', '2023년09월_계_20세', '2023년09월_계_21세', '2023년09월_계_22세', '2023년09월_계_23세', '2023년09월_계_24세', '2023년09월_계_25세', '2023년09월_계_26세', '2023년09월_계_27세', '2023년09월_계_28세', '2023년09월_계_29세', '2023년09월_계_30세', '2023년09월_계_31세', '2023년09월_계_32세', '2023년09월_계_33세', '2023년09월_계_34세', '2023년09월_계_35세', '2023년09월_계_36세', '2023년09월_계_37세', '2023년09월_계_38세', '2023년09월_계_39세', '2023년09월_계_40세', '2023년09월_계_41세', '2023년09월_계_42세', '2023년09월_계_43세', '2023년09월_계_44세', '2023년09월_계_45세', '2023년09월_계_46세', '2023년09월_계_47세', '2023년09월_계_48세', '2023년09월_계_49세', '2023년09월_계_50세', '2023년09월_계_51세', '2023년09월_계_52세', '2023년09월_계_53세', '2023년09월_계_54세', '2023년09월_계_55세', '2023년09월_계_56세', '2023년09월_계_57세', '2023년09월_계_58세', '2023년09월_계_59세', '2023년09월_계_60세', '2023년09월_계_61세', '2023년09월_계_62세', '2023년09월_계_63세', '2023년09월_계_64세', '2023년09월_계_65세', '2023년09월_계_66세', '2023년09월_계_67세', '2023년09월_계_68세', '2023년09월_계_69세', '2023년09월_계_70세', '2023년09월_계_71세', '2023년09월_계_72세', '2023년09월_계_73세', '2023년09월_계_74세', '2023년09월_계_75세', '2023년09월_계_76세', '2023년09월_계_77세', '2023년09월_계_78세', '2023년09월_계_79세', '2023년09월_계_80세', '2023년09월_계_81세', '2023년09월_계_82세', '2023년09월_계_83세', '2023년09월_계_84세', '2023년09월_계_85세', '2023년09월_계_86세', '2023년09월_계_87세', '2023년09월_계_88세', '2023년09월_계_89세', '2023년09월_계_90세', '2023년09월_계_91세', '2023년09월_계_92세', '2023년09월_계_93세', '2023년09월_계_94세', '2023년09월_계_95세', '2023년09월_계_96세', '2023년09월_계_97세', '2023년09월_계_98세', '2023년09월_계_99세', '2023년09월_계_100세 이상']
['서울특별시  (1100000000)', '9,407,540', '9,407,540', '38,101', '41,599', '43,518', '44,893', '48,097', '51,107', '56,215', '64,176', '66,545', '66,588', '68,791', '73,452', '73,667', '70,811', '70,182', '75,957', '76,605', '71,528', '72,968', '85,145', '88,965', '95,128', '114,431', '130,725', '134,291', '147,512', '157,700', '164,260', '165,915', '169,505', '169,721', '167,914', '154,238', '145,524', '139,088', '132,125', '129,417', '129,449', '128,551', '132,545', '141,799', '149,998', '151,352', '151,696', '141,877', '131,295', '134,730', '132,627', '140,149', '152,989', '154,654', '157,451', '166,643', '157,897', '159,261', '151,094', '139,108', '134,184', '138,870', '130,717', '141,481', '135,134', '151,553', '151,136', '137,487', '129,042', '128,405', '117,703', '123,829', '101,401', '92,397', '96,553', '70,066', '78,008', '78,177', '76,171', '75,999', '56,031', '55,814', '53,247', '54,490', '56,569', '44,137', '37,671', '34,905', '29,591', '25,447', '21,166', '18,331', '14,079', '11,132', '8,984', '6,319', '5,441', '4,430', '3,158', '2,286', '1,306', '916', '703', '1,505']
```

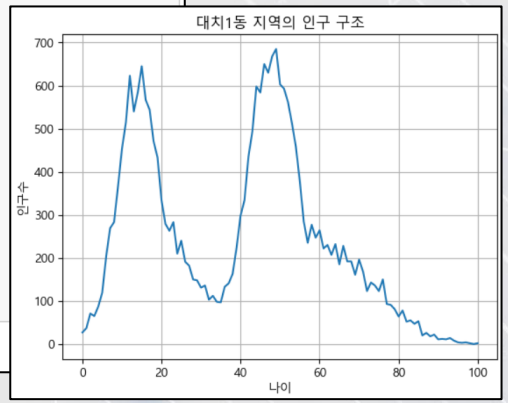
문자열 자료형이고 숫자 사이에 쉼표 (,)가 있습니다.

궁금한 지역의 인구 구조 시각화하기

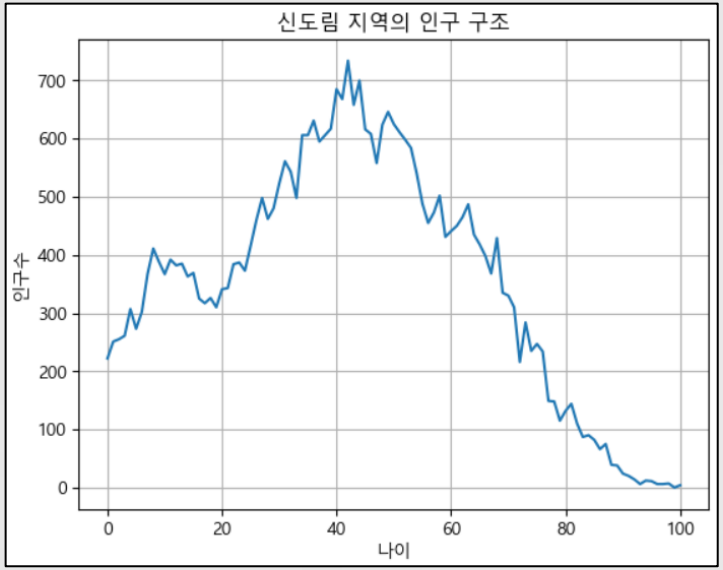
```
import csv
import numpy as np
import matplotlib.pyplot as plt

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

name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    row[2:] = map(lambda s: s.replace(',', ''), row[2:])
    if name in row[0] :
        home = np.array(row[3:], dtype = int)
        break

f.close()

plt.rc('font', family ='Malgun Gothic')
plt.plot(home)
plt.title(f'{name} 지역의 인구 구조')
plt.grid(True)
plt.xlabel('나이')
plt.ylabel('인구수')
plt.show()
```

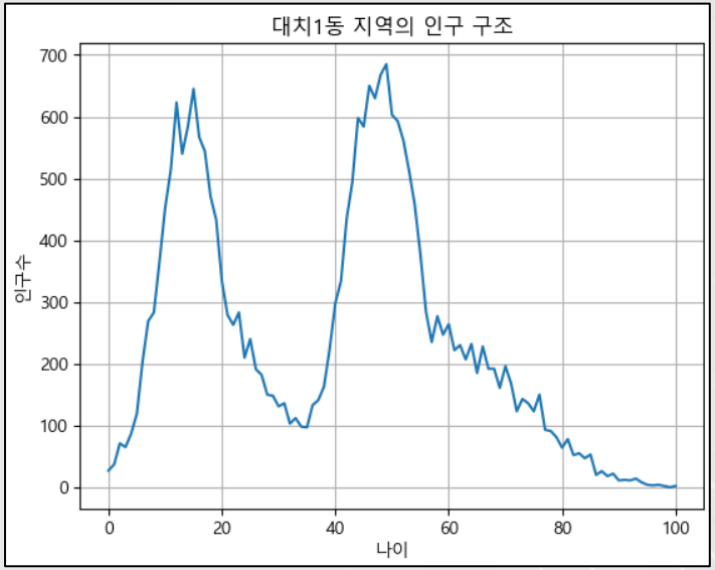
```
인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : 신도림
```





궁금한 지역의 인구 구조 시각화하기





궁금한 지역의 인구 구조와 가장 비슷한 인구 구조를 가진 지역 찾기

- 내가 알고자 하는 궁금한 지역: A
- 비교할 지역(= A지역을 제외한 나머지 지역): B

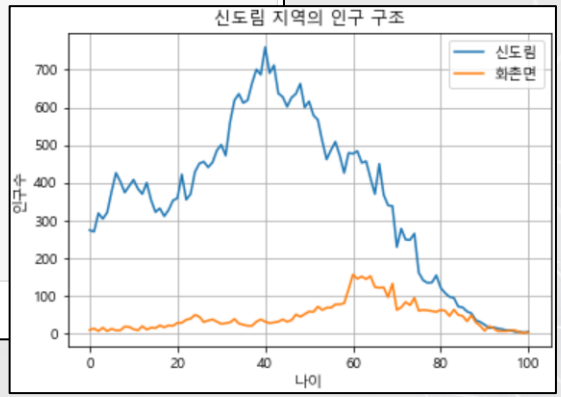
A와 B의 인구 구조가 비슷하다는 것을
어떻게 알 수 있을까요?

궁금한 지역의 인구 구조와 가장 비슷한 인구 구조를 가진 지역 찾기

```
# 관심 지역과 '화촌면'의 연령별 인구를 그리기
import csv
import numpy as np
import matplotlib.pyplot as plt

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

name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    row[2:] = map(lambda s: s.replace(',', ''), row[2:])
    if name in row[0] :
        home = np.array(row[3:], dtype = int)
    if '화촌면' in row[0] :
        homeB = np.array(row[3:], dtype = int)
        print(row[0])

f.close()

plt.rc('font', family ='Malgun Gothic')
plt.plot(home, label=name)
plt.plot(homeB, label='화촌면')
plt.title(name +' 지역의 인구 구조')
plt.grid(True)
plt.xlabel('나이')
plt.ylabel('인구수')
plt.legend()
plt.show()
```

```
인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : 신도림
강원특별자치도 홍천군 화촌면(5172031000)
```

- 관심지역과 '화촌면'의 인구 구조 시각화



인구수 차이가 아니라 인구 비율을 고려해 보겠습니다.

궁금한 지역의 인구 구조와 가장 비슷한 인구 구조를 가진 지역 찾기

```
# 관심 지역과 '화촌면'의 연령별 인구비율 그리기
import csv
import numpy as np
import matplotlib.pyplot as plt

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

name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    # row[2]를 나누기에 사용하므로 먼저 정수로 변환
    row[2:] = map(int, map(lambda s: s.replace(',', ''), row[2:]))
    if name in row[0] :
        home = np.array(row[3:], dtype = int) / row[2]
    if '화촌면' in row[0] :
        homeB = np.array(row[3:], dtype = int) / row[2]

f.close()

# plt.style.use('ggplot')

plt.rc('font', family ='Malgun Gothic')
plt.plot(home, label=name)
plt.plot(homeB, label='화촌면')
plt.title(f'{name} 지역과 화촌면 지역의 인구 구조 비교')
plt.grid(True)
plt.xlabel('나이')
plt.ylabel('인구비율')
plt.legend()
plt.show()
```

```
인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : 신도림
```

- 관심지역과 '화촌면'의 연령별비율 시각화

전체 인구수가 다르더라도
인구 구조를 비교할 수 있을 것 같습니다.

궁금한 지역의 인구 구조와 가장 비슷한 인구 구조를 가진 지역 찾기

```
import csv
import numpy as np
import matplotlib.pyplot as plt

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

print(type(data))

name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    # row[2]를 나누기에 사용하므로 먼저 정수로 변환
    row[2:] = map(int, map(lambda s: s.replace(',', ''), row[2:]))
    if name in row[0] :
        home = np.array(row[3:], dtype = int) / row[2]

# print(home)
# 반복자는 한번 반복한 이후 더 이상 반복을 못함
for row in data :
    print(row)

f.close()
```

아무것도 출력되지 않습니다!

```
<class '_csv.reader'>
인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : 신도림
```

궁금한 지역의 인구 구조와 가장 비슷한 인구 구조를 가진 지역 찾기

```
import csv
import numpy as np
import matplotlib.pyplot as plt

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

print(type(data))
data = list(data)  # 추가
print(type(data))

name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    # row[2]를 나누기에 사용하므로 먼저 정수로 변환
    row[2:] = map(int, map(lambda s: s.replace(',', ''), row[2:]))
    if name in row[0] :
        home = np.array(row[3:], dtype = int) / row[2]

for row in data :
    print(row)

f.close()
```

다시 반복을 하기 위해 리스트로 변환해서 작업

궁금한 지역의 인구 구조와 가장 비슷한 인구 구조를 가진 지역 찾기

리스트의 항목을 수정해 저장

- 원래의 리스트에 반영

```
d = [['1', '2'], ['10', '20', '30'], ['11', '22', '33']]
print(d) # 수정 전 출력

# 리스트의 내부 항목을 수정해서 대입하면 바로 리스트가 수정됨
for item in d:
    for i in range(len(item)):
        item[i] = int(item[i])

print(d) # 수정 후 출력
```

```
[['1', '2'], ['10', '20', '30'], ['11', '22', '33']]
[[1, 2], [10, 20, 30], [11, 22, 33]]
```

가장 비슷한 곳의 인구 구조와 궁금한 지역의 인구 구조 시각화하기

알고리즘 버전 (“차이의 제곱의 합”이 가장 작은 지역)

- ⓐ 전국의 모든 지역 중 한 곳 (B)을 선택한다.
- ⓑ 궁금한 지역 A의 0세 인구 비율에서 B의 0세 인구 비율을 뺀다.
- ⓒ ⓑ를 “100세 이상 인구수”에 해당하는 값까지 반복한 후 각각의 차이의 제곱의 합을 구한다.
- ⓓ 전국의 모든 지역에 대해 반복하며 그 차이가 가장 작은 지역을 찾는다.

	0세	1세	2세	…	100세이상
A지역(away)	인구수/지역전체인구수	인구수/지역전체인구수	인구수/지역전체인구수	…	인구수/지역전체인구수
B지역(home)	인구수/지역전체인구수	인구수/지역전체인구수	인구수/지역전체인구수	…	인구수/지역전체인구수

```
np.sum((home - away) ** 2)
```

가장 비슷한 곳의 인구 구조와 궁금한 지역의 인구 구조 시각화하기

알고리즘 버전 (“차이의 제곱의 합”이 가장 작은 지역)

- ⓐ 전국의 모든 지역 중 한 곳 (B)을 선택한다.
- ⓑ 궁금한 지역 A의 0세 인구 비율에서 B의 0세 인구 비율을 뺀다.
- ⓒ ⓑ를 “100세 이상 인구수”에 해당하는 값까지 반복한 후 각각의 차이의 제곱의 합을 구한다.
- ⓓ 전국의 모든 지역에 대해 반복하며 그 차이가 가장 작은 지역을 찾는다.

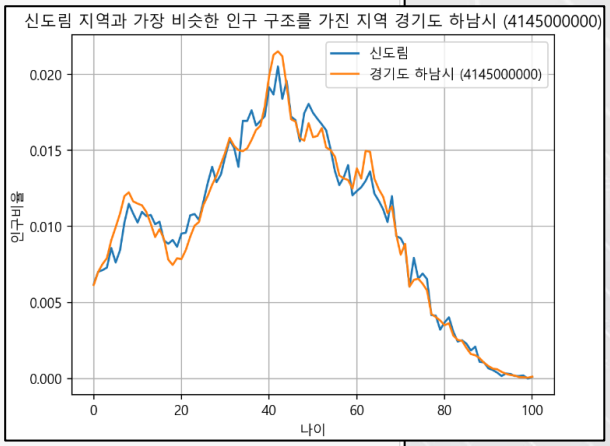
```
name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    # row[2]를 나누기에 사용하므로 먼저 정수로 변환
    row[2:] = map(int, map(lambda s: s.replace(',', ''), row[2:]))
    if name in row[0] :
        home = np.array(row[3:], dtype = int) / row[2]

for row in data :
    if name not in row[0] :
        if row[2] == 0:
            continue

        away = np.array(row[3:], dtype = int) / row[2]
        s = np.sum((home - away) ** 2)
        if s < min_val:
            min_val = s
            result_name = row[0]
            result = away
```

가장 비슷한 곳의 인구 구조와 궁금한 지역의 인구 구조 시각화하기

```
import csv
import numpy as np
import matplotlib.pyplot as plt

f = open('age.csv', encoding='cp949')
data = csv.reader(f)
header = next(data)
data = list(data)  #추가

min_val = 1
result_name = ''
result = 0

name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    # row[2]를 나누기에 사용하므로 먼저 정수로 변환
    row[2:] = map(int, map(lambda s: s.replace(',', ''), row[2:]))
    if name in row[0] :
        home = np.array(row[3:], dtype = int) / row[2]

for row in data :
    if name not in row[0] :
        if row[2] == 0:
            continue

        away = np.array(row[3:], dtype = int) / row[2]
        s = np.sum((home - away) ** 2)
        if s < min_val:
            min_val = s
            result_name = row[0]
            result = away

f.close()
```

```
plt.rc('font', family ='Malgun Gothic')
plt.plot(home, label=name)
plt.plot(result, label=result_name)
plt.title(f'{name} 지역과 가장 비슷한 인구 구조를 가진 지역 {result_name}')
plt.xlabel('나이')
plt.ylabel('인구비율')
plt.grid(True)
plt.legend()
plt.show()
```

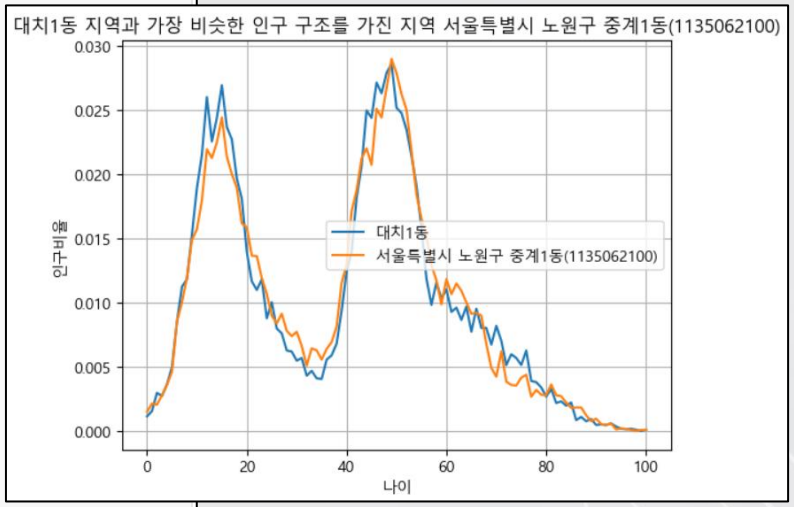
```
인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : 신도림
```



가장 비슷한 곳의 인구 구조와 궁금한 지역의 인구 구조 시각화하기

```
import csv
import numpy as np
import matplotlib.pyplot as plt

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

data = list(data)  #추가

min_val = 1
result_name = ''
result = 0

name = input('인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : ')
for row in data :
    # row[2]를 나누기에 사용하므로 먼저 정수로 변환
    row[2:] = map(int, map(lambda s: s.replace(',', ''), row[2:]))
    if name in row[0] :
        home = np.array(row[3:], dtype = int) / row[2]

for row in data :
    if name not in row[0] :
        if row[2] == 0:
            continue

        away = np.array(row[3:], dtype = int) / row[2]
        s = np.sum((home - away) ** 2)
        if s < min_val:
            min_val = s
            result_name = row[0]
            result = away

f.close()

plt.rc('font', family ='Malgun Gothic')
plt.plot(home, label=name)
plt.plot(result, label=result_name)
plt.title(f'{name} 지역과 가장 비슷한 인구 구조를 가진 지역 {result_name}')
plt.xlabel('나이')
plt.ylabel('인구비율')
plt.grid(True)
plt.legend()
plt.show()
```

인구 구조가 알고 싶은 지역의 이름(읍면동 단위)을 입력해주세요 : 대치1동



SUMMARY

학습정리

- 데이터분석을 위한 문제를 보다 명확하게 정의
  - 코드로 구현이 가능하도록
- csv.reader(f)의 결과
  - 클래스 _csv.reader로 반복자(iterator)
  - 한 번 반복하면 다시 반복이 불가능
- 차이 분석
  - '차이 값의 제곱의 합'이 작은 것이 차이가 작다
- 내장 함수 map을 사용한 '콤마 숫자 문자열' 숫자 변환
  - row[2:] = map(int, map(lambda s: s.replace(',', ''), row[2:]))
- data의 자료 형 변환
  - np.array(data, dtype=int)