

Visual Programming Assignment 5



사회학과

20173666

박하람

2019년 2학기

Assignment5

#02_Class/비주얼프로그래밍

1. 득표율 그래프 그리기

```
# pandas와 matplotlib를 import 해줌.
import pandas as pd
import matplotlib.pyplot as plt

# matplotlib가 한글이 깨져서 나오므로 한글이 표기될 수 있도록 해줌.
plt.rcParams['font.family'] = 'AppleGothic'

# while문을 이용해 excel이 수정될 때마다 데이터를 새롭게 가져올 수 있도록 해줌.
while True:
    excel_file = 'elec.xlsx'
    df = pd.read_excel(excel_file)

    names = df['이름']
    rate = df['득표율']

    colors = ['yellowgreen', 'lightskyblue', 'lightcoral', 'grey']
    explodes = (0.03, 0.03, 0.03, 0.03)

    plt.pie(rate,
            labels=names,
            colors=colors,
            autopct='%1.2f%%',
            explode=explodes
            )

    plt.title('20대 선거 득표율', fontsize=20)
    plt.draw()
    plt.pause(10)
    plt.clf()
```

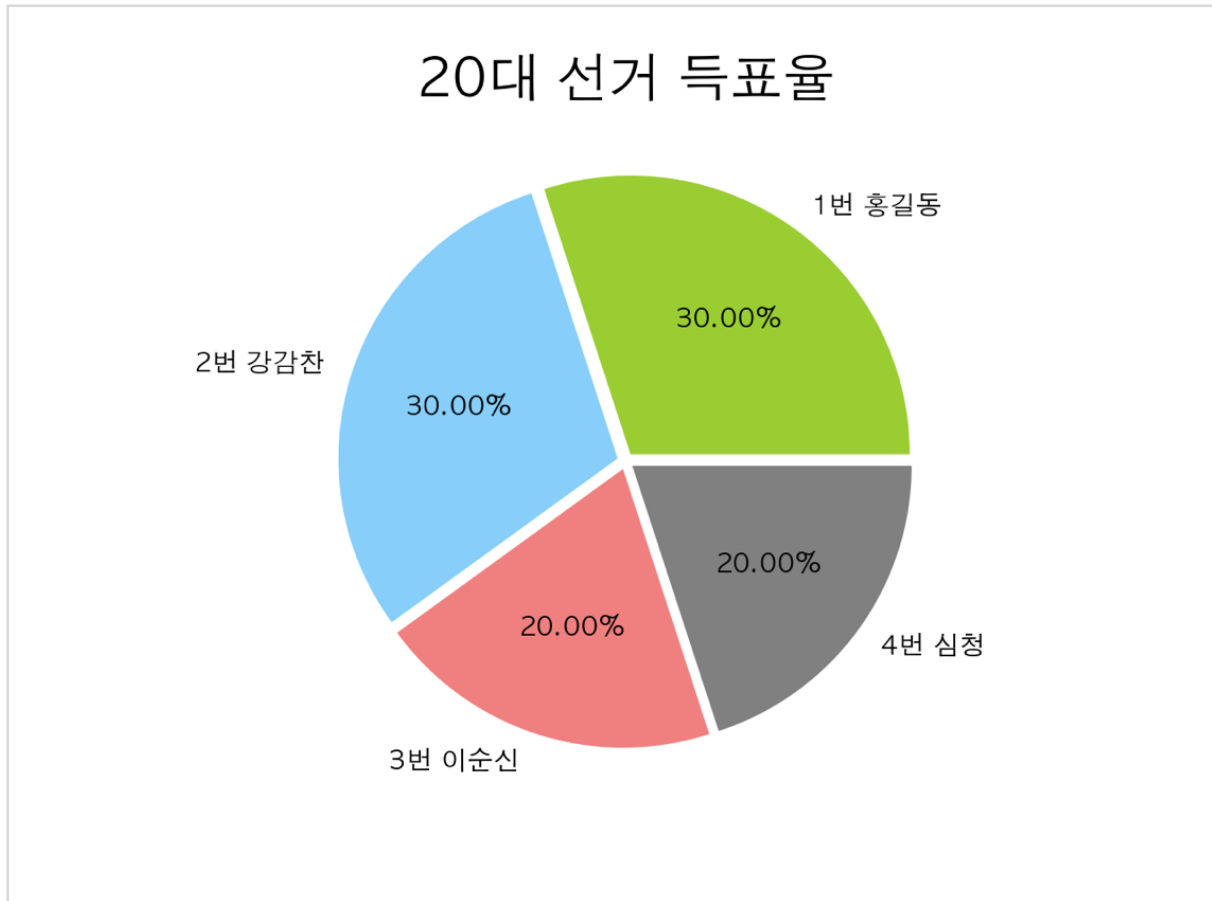
pandas와 matplotlib를 import 해주어 파이그래프를 그렸다. pandas의 read_excel를 사용하여 데이터프레임 df에다가 excel의 데이터를 담아주었다. names 변수에는 df의 '이름' 열만 가지고 오도록 하였고, rate 변수에는 df의 '득표율' 열만 가지고 오게 해주었다. colors를 사용해 각각 데이터의 색이 다르게

나올 수 있도록 했고, explode를 사용해 파이조각들이 서로 분리되게 하였다.

plt.draw()를 이용해 파이그래프를 그리고, 잠시 10초 동안 plt.pause(10)로 멈추게 한 후, plt.clf()를 이용해 깨끗하게 그래프를 지워서 새로운 데이터로 수정될 때에 다시 파이그래프로 반영될 수 있도록 하였다.

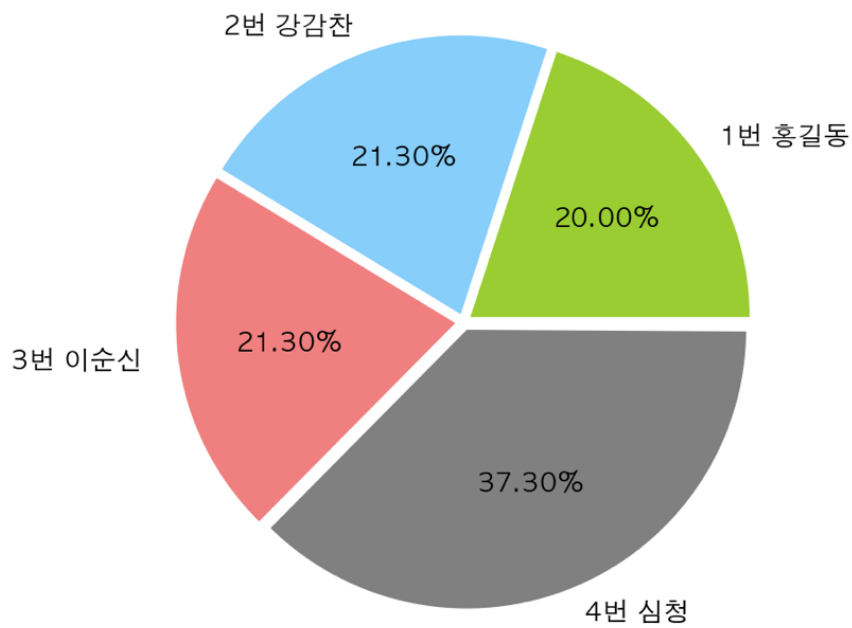
결과:

Excel 수정 전 데이터 :



Excel 수정 후 데이터 :

20대 선거 득표율



2. 정규분포의 랜덤값 생성

```
import numpy as np
import random
import matplotlib.pyplot as plt
import pandas as pd

# random.randint를 사용해 리스트 random1에다 넣어줌.
random1 = []
for i in range(100):
    number = random.randint(0, 100)
    random1.append(number)
random1.sort()

# count1에다 random1이 나온 빈도수만큼 넣어줌.
count1 = []
for i in random1:
    count1.append(random1.count(i))

# np.random.normal을 생성이 평균 50, 표준편차 10인 정규분포에서 난수 100개 생성.
```

```

random2 = np.random.normal(50, 10, size=100)
# 생성된 난수를 정수로 타입 변환시켜줌.
random2 = np.array(random2, dtype=np.int)
random2 = np.sort(random2)

# 생성된 난수에 대한 빈도수를 count2에다 담아줌.
count2 = np.bincount(random2, minlength=100)

# 판다스 DataFrame에다가 random2와 count2를 담음.
df1 = pd.DataFrame(random2)
df2 = pd.DataFrame(count2)
print(df1, df2)

# 그래프 그리기
plt.plot(random1, count1, marker='o')
plt.plot(df1, df2, marker='o')
plt.show()

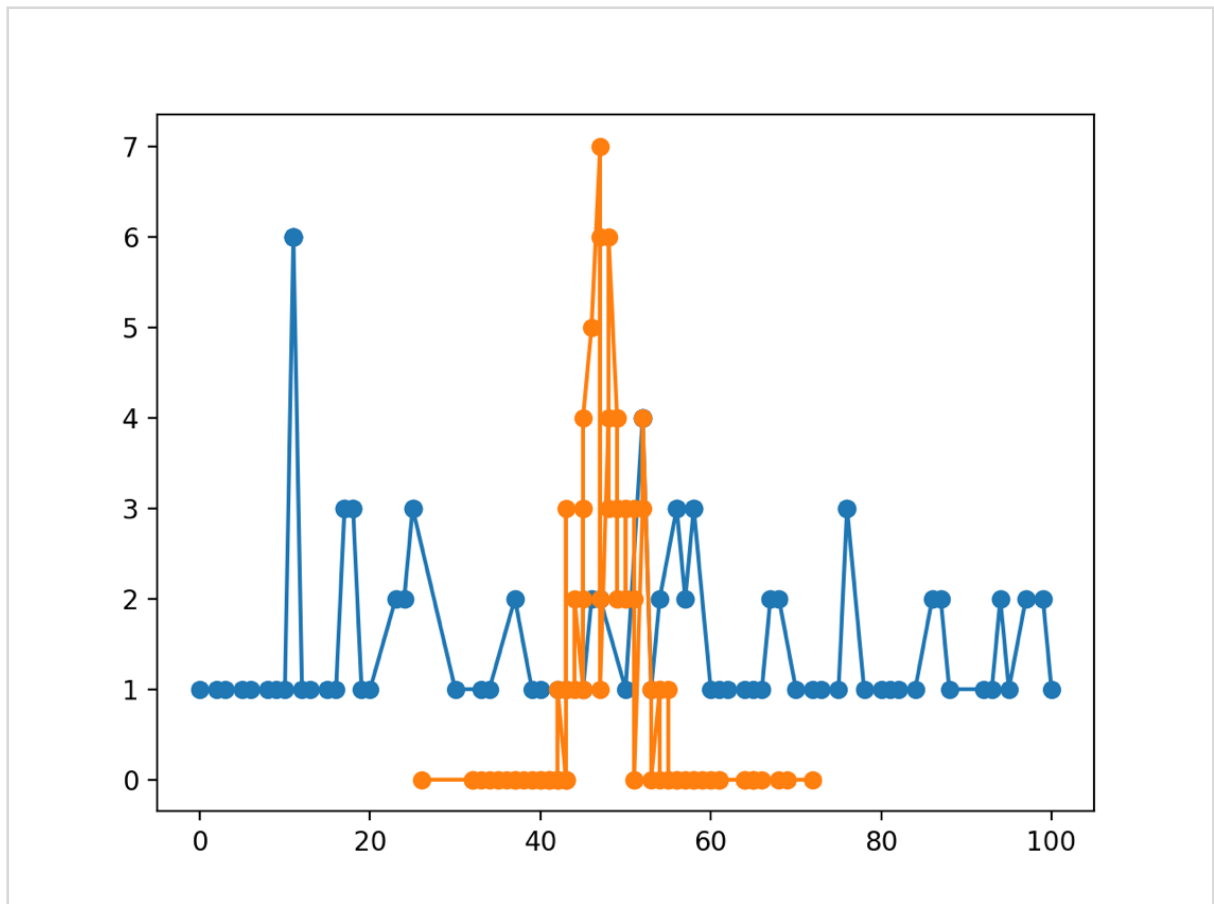
```

Numpy, random, matplotlib, pandas 모듈을 Import 해주었다.

random1에는 random.randint를 사용해 1-100까지의 난수를 생성해주었다. 또한 count1에는 random1의 각각의 숫자들의 빈도수를 측정하게 하였다. 그래프를 생성해줄 때, random1을 x축에, count1을 y축에 넣어주어 그래프를 그려주었다.

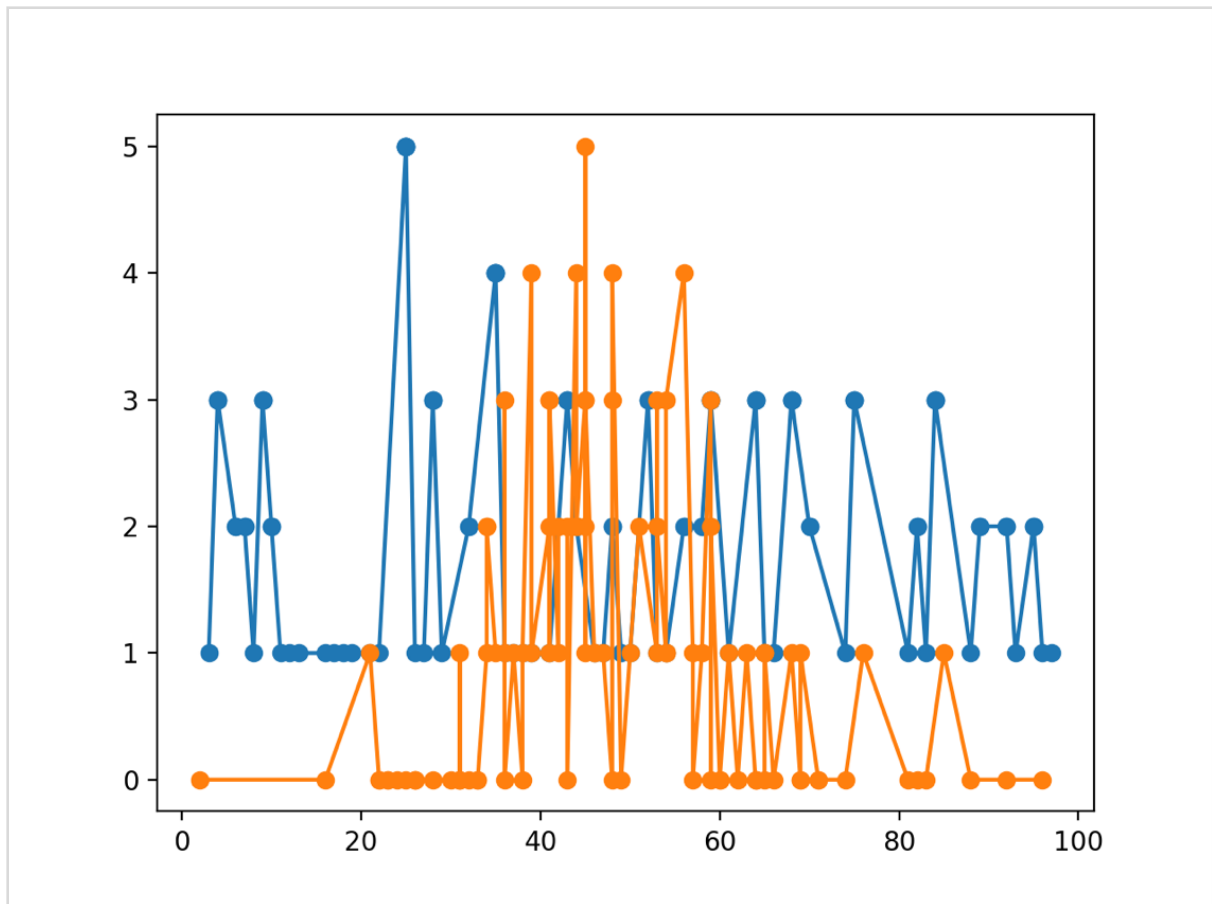
random2에는 numpy의 random normal을 이용해 평균이 50이고, 표준편차가 10인 정규분포 내에서 100개의 난수를 가지고 오게 하였다. 빈도수가 확실히 표현되도록 하기 위해서(정수여야 더 겹치는 수가 많아져서 빈도수 그래프가 명확하게 보이기 때문이다.) 타입을 정수로 변환해주었다. count2에는 bincount를 이용해 빈도수가 측정되도록 하였다. 이를 효과적으로 담기 위해서 random2와 count2를 모두 DataFrame에다가 담아주어 x축에는 df1, y축에는 df2가 담기어 보여주게 하였다.

결과 :



파란색 그래프가 random을 사용해 만든 난수이고, 노란색 그래프가 np.random.normal을 사용해서 만든 난수이다. 파란색은 01부터 100까지의 수 중에 빈도가 어떤 특정한 구간에 몰려있지 않고 비교적 산발적인 그래프를 보인다.

그래서 정규분포에서 가져온 난수의 경우에는 평균이 50이므로 50 주변으로 큰 빈도수를 보인다. 양 극단에는 거의 난수가 나오지 않는다고 할 수 있다.



random2의 표준편차를 18로 해주면, 비교적 앞의 노란색 그래프보다는 더 양극단으로 넓어진 그래프의 모습을 보인다. 소위말해 더 퍼져잡해지는 모습을 그린다. 이를 파란색 그래프와 비교해봤을 때, 노란색은 앞선 노란색 그래프보다 더 넓어진 모양새를 보이거나 기본적으로 평균 50을 중심으로 많은 빈도수를 보인다. 그러나 파란색 그래프는 특정 한곳에서만 난수가 많이 발현되는 것이 아니며, 산발적인 모습을 보인다.