

작성일 : 2022.05.29

acci.csv 데이터를 읽어 데이터 시각화(Plot 그리기)
: matplotlib 라이브러리 활용

휴먼지능정보공학과 202210829 김진석

목차

1. 발생건수, 사망자수, 부상자수, 증상, 경상, 부상신고 각각에 대해 평균 구하기

- 1.1 제목
- 1.2 해결방법
- 1.3 결과
- 1.4 결론
- 1.5 트러블 슈팅
 - 1.5.1 문제 원인
 - 1.5.2 해결 방안
 - 1.5.3 참고 자료

2. matplotlib를 이용하여 사망자 수를 bar 그래프로 표현하기(시각화)

- 2.1 제목
- 2.2 해결방법
- 2.3 결과
- 2.4 결론
- 2.5 트러블 슈팅
 - 2.5.1 문제 원인
 - 2.5.2 해결 방안
 - 2.5.3 참고 자료

3. matplotlib를 이용하여 부상자수, 증상, 경상을 line 그래프로 표현하기(시각화)

- 3.1 제목
- 3.2 해결방법
- 3.3 결과
- 3.4 결론

4. 참고 자료

5. 부록

1. 발생건수, 사망자수, 부상자수, 중상, 경상, 부상신고 각각에 대해 평균을 구한다.

1.1 제목

발생 건수, 사망자 수, 부상자 수, 중상자 수, 경상자 수, 부상신고 수의 평균

1.2 해결방법¹⁾

판다스 객체를 사용해 acci 변수에 'acci.csv' 파일을 'cp949'로 인코딩해서 읽어 온다. acci.csv 파일을 읽어 온 변수에 mean() 함수를 사용하여 발생 건수, 사망자 수, 부상자 수, 중상자 수, 경상자 수, 부상신고 수의 평균을 구한다. 각 열에 해당하는 값의 평균을 구하기 위해 mean() 함수 안에 매개변수 axis 값을 0으로 해준다. 그 결과 각 열에 해당하는 행의 값의 평균을 구할 수 있게 된다.

1.3 결과

```
발생건수    13978.5000
사망자수     337.0000
부상자수    21535.3125
중상         6356.4375
경상        14224.3750
부상신고     954.5000
dtype: float64
```

1.4 결론

판다스의 mean() 함수는 매개변수 axis의 값 0 또는 1에 따라 행의 평균이나 열의 평균을 구할 수 있다. 또한, 다른 매개변수를 통해 NaN 값을 제외하거나 float, int, boolean 열만 포함할 수 있는 세부적인 mean() 함수의 기능을 알게 되었다. 그리고 mean() 함수는 기본적으로 Series 형태로 반환한다는 것을 알았다.

1)'Pandas DataFrame DataFrame.mean() 함수', DelftStack, 2021.02.09., <<https://www.delftstack.com/ko/api/python-pandas/pandas-dataframe-dataframe.mean-function/>>

1.5 트러블 슈팅

1.5.1 문제 원인

mean 함수에 axis 매개변수만을 사용하면

“FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.”

이러한 경고문이 나온다. 경고문의 내용을 해석하면 numeric_only 매개변수는 더 이상 None 값이 올 수 없다는 뜻이었다. 향후 버전에서는 TypeError가 발생하므로 numeric_only 값을 설정 해주어야 한다는 뜻이다. 여기서 numeric_only 매개변수는 True 이면 float, int, boolean 열만을 포함한다는 뜻이다.

1.5.2 해결 방안

현재 사용하는데 지장은 없지만 오류문을 없애 주기 위해 매개변수 numeric_only 값을 지정해 주었다.

1.5.3 참고 자료

<https://stackoverflow.com/questions/70897794/finding-the-mean-of-nuisance-columns-in-dataframe-error>

2. matplotlib를 이용하여 사망자 수를 bar 그래프로 표현한다.

2.1 제목

bar 그래프로 사망자 수 그리기(시각화)

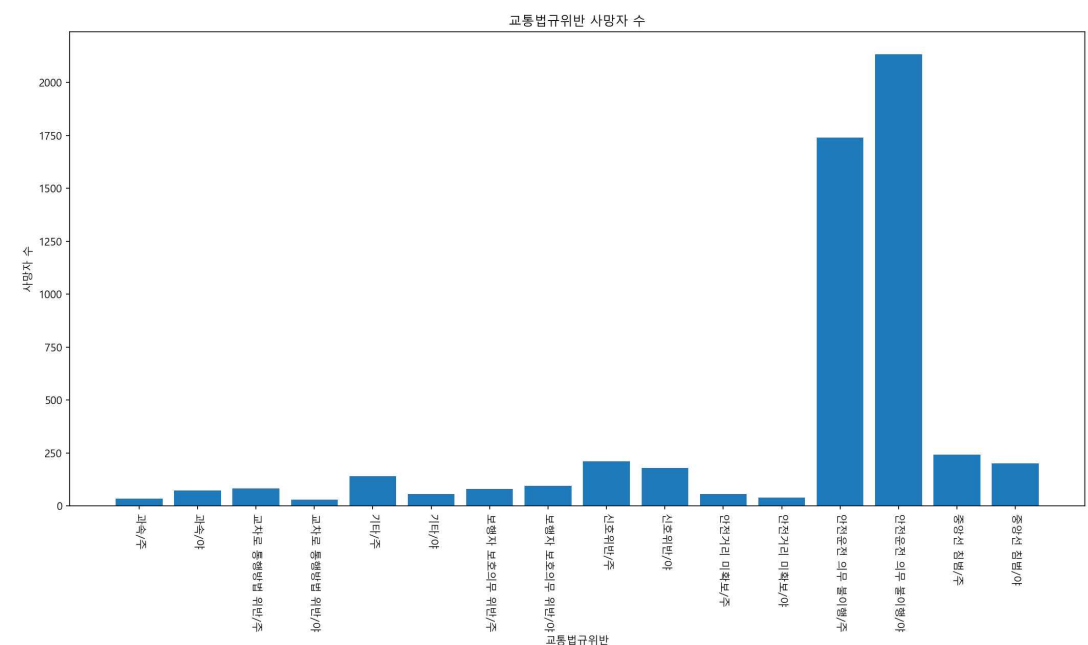
2.2 해결방법

acci 데이터프레임에서 ‘법규위반’과 ‘주야’를 선택하기 위해 딕셔너리에서 key를 선택해 value를 불러오는 것처럼 ‘법규위반’과 ‘주야’ 값을 선택한다. 법규위반 데이터는 ‘주’와 ‘야’로 구분되기 때문에 ‘법규위반’ 종류와 거기에 해당하는 ‘주야’ 값을 ‘+’ 연산자로 합쳐 acci.index 변수에 저장한다. 그리고 사망자 수 값을 위와 같이 열 이름으로 선택해 dead 변수에 저장한다.

bar 그래프를 그리기 위해 matplotlib 라이브러리의 bar 함수를 사용한다. bar 함수는 매

개변수를 `bar(x, y)`로 가지기 때문에 x축에 `accident.index`를 y축에 `dead` 변수를 집어넣으면 법규위반을 x축으로 가지는 사망자 수 bar 그래프를 시각화할 수 있다. matplotlib 라이브러리의 `title`, `xlabel`, `ylabel` 함수를 사용하여 제목과 x, y 레이블 값을 표시한다. 또한, x 축의 값이 길어 다른 x축 값과 겹치는 상황을 고려해 `xticks(rotation = 270)`을 사용해 x 축의 값을 270도 회전해서 표시한다.

2.3 결과



2.4 결론

x축 값을 설정할 때 ‘법규위반’ 값을 ‘주야’로 구분하기 위해서는 ‘+’ 연산자를 사용하면 ‘법규위반’과 ‘주야’ 값을 합칠 수 있다는 것을 알았다. 데이터프레임에서 열에 해당하는 값을 가져오는 방식이 딕셔너리의 `key`와 `value` 개념과 비슷하다. matplotlib 라이브러리의 `title`, `xlabel`, `ylabel`, `xticks`를 통해 제목, x,y 레이블 x축 값의 각도를 설정할 수 있다.

2.5 트러블 슈팅

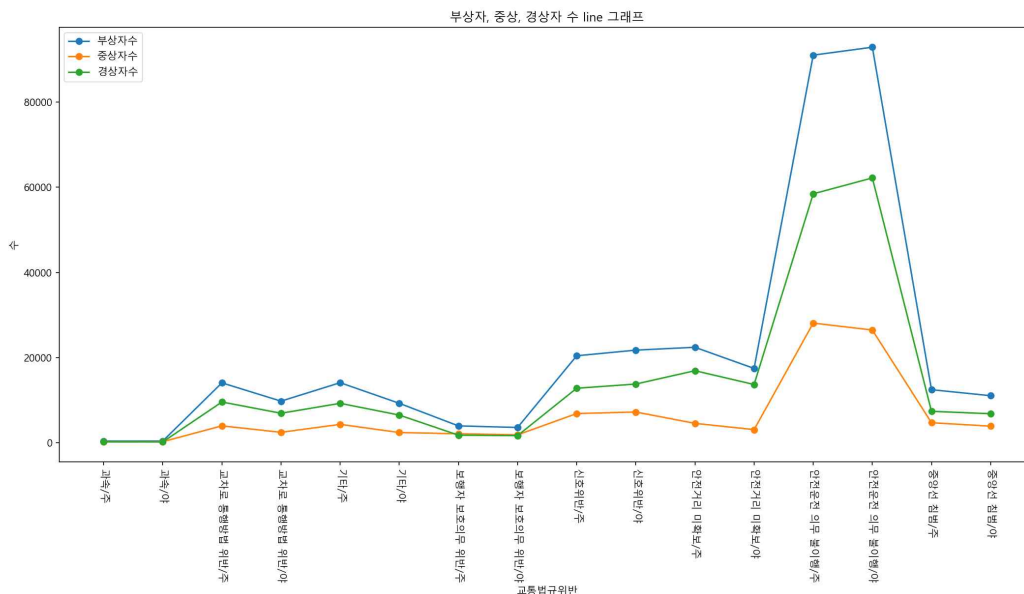
2.5.1 문제 원인

그래프를 출력했을 때 한글 폰트가 깨져서 나오는 상황

3.2 해결방법

x축은 위에서 저장한 acci.index 변수를 사용하고 y축은 '부상자 수', '중상', '경상' 데이터 값으로 각각 설정한다. marker의 형식을 O로 설정한다. 각 label 값을 해당 데이터에 맞게 이름을 설정한다. 위의 그래프의 x축 값의 길이가 길어서 xtick함수를 사용해 270도 회전한 결과로 표시한다. 3개의 line을 그래프 하나에 시각화하기 위해서는 모두 같은 x축 값을 가지고 있어야 한다. 3개의 line을 표시하므로 범례를 표시한다. 또한, 범례의 위치를 upper left로 설정해 그래프와 겹치지 않게 한다. 그리고 xlabel, ylabel, title을 사용해 x, y 레이블과 제목을 표시한다.

3.3 결과



3.4 결론

3개의 데이터를 각각 열값에 맞게 레이블과 데이터를 선택한다. x축의 값을 겹치지 않게 하기 위해 x축의 값을 270도 회전하였다. 3개의 line을 한 번에 표시하기 위해 x축 값을 통일해주어야 한다는 것을 알게 되었다. 범례를 표시하고 위치를 변경하는 법을 알게 되었다.

4. 참고 자료

'Pandas DataFrame DataFrame.mean() 함수', DelftStack, 2021.02.09., <<https://www.delftstack.com/ko/api/python-pandas/pandas-dataframe-dataframe.mean-function/>>

5. 부록

0. 라이브러리 import와 한글 폰트 설정 그리고 acci 변수에 acci.csv 값 읽어 옴. 마지막으로 어떤 값이 있는지 확인

```
import pandas as pd
import matplotlib.pyplot as plt

plt.rc('font', family='Malgun Gothic')

acci = pd.read_csv('acci.csv', encoding='cp949')#euc-kr
print(acci.head())
```

1. 발생건수, 사망자수, 부상자수, 중상, 경상, 부상신고 각각에 대해 평균

```
#1
print(acci.mean(axis=0, numeric_only = True))
```

2. matplotlib를 이용하여 사망자 수를 bar 그래프로 표현

```
#2
acci.index = acci['법규위반']+'/' + acci['주야']

dead = acci['사망자수']

plt.bar(
    acci.index,
    dead
)
plt.title("교통법규위반 사망자 수")
plt.xlabel("교통법규위반")
plt.ylabel("사망자 수")
plt.xticks(rotation = 270)
plt.show()
```

3. matplotlib를 이용하여 부상자수, 중상, 경상을 line 그래프로 표현


```
a = acci['부상자수']
plt.plot(acci.index,a,marker = 'o',label = "부상자수")
b = acci['중상']
plt.plot(acci.index,b,marker = 'o',label = "중상자수")
c = acci['경상']
plt.plot(acci.index,c,marker = 'o',label = "경상자수")

plt.xticks(rotation = 270)
plt.legend(loc='upper left',ncol = 1)
plt.xlabel("교통법규위반")
plt.ylabel("수")
plt.title("부상자, 중상, 경상자 수 line 그래프")

plt.show()
```