파이썬 EDA

1. 범주형 변수의 기술통계

- 데이터셋 불러오기 : sns.load_dataset(“mpg”) → df.shape
- 요약하기 : info *object로 되어있는 변수가 범주형
- 결측치 확인 : df → df.isnull
- 기술통계 : df.describe(include=”object”) ← 범주형 기술통계
- 최빈값 : top, 최빈값의 빈도수 : freq

```
[ ]  # describe 를 통해 범주형 변수에 대한 기술통계를 보기
     df.describe(include="object")
```

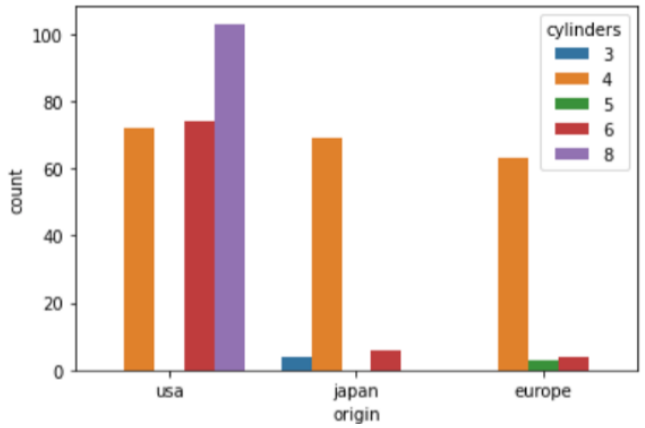
	origin	name
count	398	398
unique	3	305
top	usa	ford pinto
freq	249	6

2. 범주형 변수의 빈도수

- 유일값의 빈도수 구하기 : df.nunique
- origin 빈도수 시각화 : sns.countplot(data=df, x=”origin”) *y는 알아서 빈도수로 함
- 빈도수 구하기 : df[”origin”].value_counts *1개의 변수
- origin빈도수 시각화하고 cyliders색상 표현 : sns.countplot(data=df, x=”origin”, hue=”cylinders”) *hue=색상

```
[ ]  # countplot 으로 origin 의 빈도수를 시각화 하고 cylinders 로 다른 색상으로 표현하기
     sns.countplot(data=df, x="origin", hue="cylinders")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f3ab3b79c50>
```



- 2개 이상의 변수 빈도수 구하기 : pd.crosstab(df[”origin”], df[”cylinders”]) *앞은 인덱스 뒤는 칼럼

```
[ ]  # pd.crosstab 으로 시각화한 값 직접 구하기
     pd.crosstab(df["origin"], df["cylinders"])
```

cylinders	3	4	5	6	8
origin					
europe	0	63	3	4	0
japan	4	69	0	6	0
usa	0	72	0	74	103

3. 범주형과 수치형 변수를 막대그래프로 시각화 하기

- groupby통해 그룹화하고 평균 구하기 : df.groupby(”origin”)[”mpg”].mean()

```
[ ]  # groupby를 통해 origin 별로 그룹화 하고 mpg 의 평균 구하기
     df.groupby("origin")["mpg"].mean()
```

```
origin
europe    27.891429
japan     30.450633
usa       20.083534
Name: mpg, dtype: float64
```

- pivot_table(직관적) : pd.pivot_table(data=df, index="origin", values="mpg") * 데이터프레임 형식으로 변환

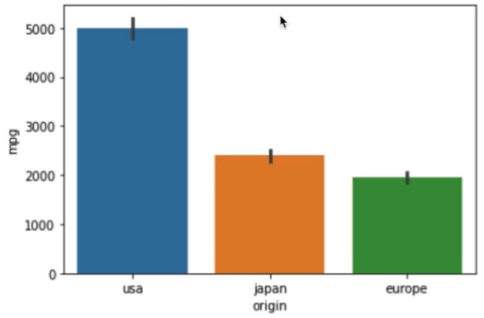
```
[ ]  # pivot_table 로 같은 값 구하기
     pd.pivot_table(data=df, index="origin", values="mpg")
```

origin	mpg
europe	27.891429
japan	30.450633
usa	20.083534

- barplot으로 합계 값 구하기 : sns.barplot(data=df, x="origin", y="mpg", estimator=np.sum, ci=None)

```
# barplot 으로 합계 값 구하기
sns.barplot(data=df, x="origin", y="mpg", estimator=np.sum)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f393a94f9d0>
```



- barplot에 hue 를 사용하여 색상 다르게 표현 : sns.barplot(data=df, x="cylinders", y="mpg", ci=None, hue = "origin")
- groupby 통해 시각화에 대한 값 구하기 : df.groupby(["cylinders", "origin"])("mpg").mean().unstack() *list형식으로 해야 함

```
# groupby 를 통해 위 시각화에 대한 값을 구하기
df.groupby(["cylinders", "origin"])["mpg"].mean().unstack()
```

cylinders	europe	japan	usa
3	NaN	20.550000	NaN
4	28.411111	31.595652	27.840278
5	27.366667	NaN	NaN
6	20.100000	23.883333	19.663514
8	NaN	NaN	14.963107

- pivot_table 통해 시각화에 대한 값 구하기 : pd.pivot_table(data=df, index=”cylinders”, columns = “origin”, values=”mpg”)