제3유형_카이제곱 검정

✓ 분석 Case

Case 1. 적합도 검정 - 각 범주에 속할 확률이 같은지?

Case 2. 독립성 검정 - 두 개의 범주형 변수가 서로 독립인지?

✓ 가설검정 순서(중요!!)

> 1. 가설설정
> 2. 유의수준 확인
> 3. 검정실시(통계량, p-value 확인, 기대빈도 확인)
> 4. 귀무가설 기각여부 결정(채택/기각)

✓ 예제문제

Case 1. 적합도 검정 - 각 범주에 속할 확률이 같은지?

문제 1-1

랜덤 박스에 상품 A,B,C,D가 들어있다.

다음은 랜덤박스에서 100번 상품을 꺼냈을 때의 상품 데이터라고 할 때

상품이 동일한 비율로 들어있다고 할 수 있는지 검정해보시오.

In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
# 데이터 생성
row1 = [30, 20, 15, 35]
df = pd.DataFrame([row1], columns=['A','B','C', 'D'])
df
```

Out[2]:

	A	B	C	D
0	30	20	15	35

In [3]:
```
# 1. 가설설정
# H0 : 랜덤박스에 상품 A,B,C,D가 동일한 비율로 들어있다.
# H1 : 랜덤박스에 상품 A,B,C,D가 동일한 비율로 들어있지 않다.
```

In [4]:
```
# 2. 유의수준 확인 : 유의수준 5%로 확인
```

In [5]:
```
# 3. 검정실시(통계량, p-value)
from scipy.stats import chisquare
# chisquare(f_obs=f_obs, f_exp=f_exp)   # 관측빈도, 기대빈도

# 관측빈도와 기대빈도 구하기
f_obs = [30, 20, 15, 35]
# f_obs = df.iloc[0]
f_exp = [25, 25, 25, 25]

statistic, pvalue = chisquare(f_obs=f_obs, f_exp=f_exp)
print(statistic)
print(pvalue)
# 자유도는 n-1 = 3
```



```
10.0
0.018566135463043 25
```

In [6]:
```
# 4. 귀무가설 기각여부 결정(채택/기각)
# p-value 값이 0.05보다 작기 때문에 귀무가설을 기각한다.
# 즉, 랜덤박스에 상품 A,B,C,D가 동일한 비율로 들어있지 않다고 할 수 있다.

# 답 : 기각
```


문제 1-2

랜덤 박스에 상품 A, B, C가 들어있다.

다음은 랜덤박스에서 150번 상품을 꺼냈을 때의 상품 데이터라고 할 때

상품별로 A 30%, B 15%, C 55% 비율로 들어있다고 할 수 있는지 검정해보시오.

In [7]:
```python
import pandas as pd
import numpy as np
```

In [8]:
```python
# 데이터 생성
row1 = [50,25,75]
df = pd.DataFrame([row1], columns=['A','B','C'])
df
```


Out[8]:

	A	B	C
0	50	25	75

In [9]:
```python
# 1. 가설설정
# H0 : 랜덤박스에 상품 A,B,C가 30%, 15%, 55%의 비율로 들어있다.
# H1 : 랜덤박스에 상품 A,B,C가 30%, 15%, 55%의 비율로 들어있지 않다.
```

In [10]:
```python
# 2. 유의수준 확인 : 유의수준 5%로 확인
```

In [11]:
```python
# 3. 검정실시(통계량, p-value)
from scipy.stats import chisquare
# chisquare(f_obs=f_obs, f_exp=f_exp)   # 관측빈도, 기대빈도

# 관측빈도와 기대빈도 구하기
f_obs = [50, 25, 75]
# f_obs = df.iloc[0]
a = 150*0.3
b = 150*0.15
c = 150*0.55
f_exp = [a, b, c]

statistic, pvalue = chisquare(f_obs=f_obs, f_exp=f_exp)
print(statistic)
print(pvalue)
# 자유도는 n-1 = 2
```

빅데이터 자격증 / AI강의는 메타코드M

```
1.5151515151515151
0.46880153914023537
```

In [12]:
```python
# 4. 귀무가설 기각여부 결정(채택/기각)
# p-value 값이 0.05보다 크기 때문에 귀무가설을 채택한다.
# 즉, 랜덤박스에 상품 A,B,C가 30%, 15%, 55%의 비율로 들어있다고 할 수 있다.

# 답 : 채택
```

Case 2. 독립성 검정 - 두 개의 범주형 변수가 서로 독립인지?

문제 2-1

연령대에 따라 먹는 아이스크림의 차이가 있는지 독립성 검정을 실시하시오.

In [13]:
```python
import pandas as pd
import numpy as np
```

In [14]:
```python
# 데이터 생성
row1, row2 = [200, 190, 250], [220, 250, 300]
df = pd.DataFrame([row1, row2], columns=['딸기','초코','바닐라'], index=['10대', '20대'])
df
```

Out[14]:

	딸기	초코	바닐라
10대	200	190	250
20대	220	250	300

빅데이터 자격증 / AI강의는 메타코드M

In [15]:
```python
# 1. 가설설정
# H0 : 연령대와 먹는 아이스크림의 종류는 서로 관련이 없다(두 변수는 서로 독립이다)
# H1 : 연령대와 먹는 아이스크림의 종류는 서로 관련이 있다(두 변수는 서로 독립이 아니다)
```

In [16]:
```python
# 2. 유의수준 확인 : 유의수준 5%로 확인
```

In [17]:
```python
# 3. 검정실시(통계량, p-value, 기대빈도 확인)
from scipy.stats import chi2_contingency
```

```
statistic, pvalue, dof, expected = chi2_contingency(df)
# 공식문서상에 : statistic(통계량), pvalue, dof(자유도), expected_freq(기대빈도)

# 아래와 같이 입력해도 동일한 결과값
# statistic, pvalue, dof, expected = chi2_contingency([row1, row2])
# statistic, pvalue, dof, expected = chi2_contingency([df.iloc[0],df.iloc[1]])

print(statistic)
print(pvalue)
print(dof)   # 자유도 = (행-1)*(열-1)
print(np.round(expected, 2) ) # 반올림하고 싶다면 np.round()

# (참고) print(chi2_contingency(df))
```



```
1.708360126075226
0.4256320394874311
2
[[190.64 199.72 249.65]
 [229.36 240.28 300.35]]
```

In [18]:
```
# 4. 귀무가설 기각여부 결정(채택/기각)
# p-value 값이 0.05보다 크기 때문에 귀무가설을 채택한다.
# 즉, 연령대와 먹는 아이스크림의 종류는 서로 관련이 없다고 할 수 있다.

# 답 : 채택
```

(추가) 만약 데이터 형태가 다를경우?

In [19]:
```
# ★ tip : pd.crosstab() 사용방법
# (Case1) 만약 데이터가 아래와 같이 주어진다면?
df = pd.DataFrame({
    '아이스크림' : ['딸기','초코','바닐라','딸기','초코','바닐라'],
    '연령' : ['10대','10대','10대','20대','20대','20대'],
    '인원' : [200,190,250,220,250,300]
    })
df
```

Out[19]:

	아이스크림	연령	인원
0	딸기	10대	200
1	초코	10대	190
2	바닐라	10대	250
3	딸기	20대	220
4	초코	20대	250
5	바닐라	20대	300

In [20]:
```
# pd.crosstab(index =  , columns = , values = , aggfunc=sum)
table = pd.crosstab(index=df['연령'], columns=df['아이스크림'], values=df['인원'], aggfunc=sum)
table
# 주의 : index, columns에 순서를 꼭 확인하기
# print(table)
```

Out[20]:

아이스크림	딸기	바닐라	초코
연령			
10대	200	250	190
20대	220	300	250

In [21]:
```
# 3. 검정실시(통계량, p-value, 기대빈도 확인)
from scipy.stats import chi2_contingency

# 위와 같이 교차표 만들어서 입력
statistic, pvalue, dof, expected = chi2_contingency(table)
# 공식문서상에 : statistic(통계량), pvalue, dof(자유도), expected_freq(기대빈도)

print(statistic)
print(pvalue)
print(dof)   # 자유도 = (행-1)*(열-1)
print(np.round(expected, 2) ) # array 형태 : 반올림하고 싶다면 np.round()
```



```
1.708360126075226
0.4256320394874311
2
[[190.64 249.65 199.72]
 [229.36 300.35 240.28]]
```

In [22]:
```
# (Case2) 만약 데이터가 아래와 같이 주어진다면?
# (이해를 위한 참고용입니다, 빈도수 카운팅)
df = pd.DataFrame({
    '아이스크림' : ['딸기','초코','바닐라','딸기','초코','바닐라'],
    '연령' : ['10대','10대','10대','20대','20대','20대'],
```

```
    })
df
```

Out[22]:

	아이스크림	연령
0	딸기	10대
1	초코	10대
2	바닐라	10대
3	딸기	20대
4	초코	20대
5	바닐라	20대

In [23]:
```
# pd.crosstab(index, columns)
pd.crosstab(df['연령'], df['아이스크림'])
```

Out[23]:

아이스크림	딸기	바닐라	초코
연령			
10대	1	1	1
20대	1	1	1

문제 2-2

타이타닉에 데이터에서 성별(sex)과 생존여부(survived) 변수간

독립성 검정을 실시하시오.

In [24]:
```
import seaborn as sns
df = sns.load_dataset('titanic')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   survived     891 non-null    int64
 1   pclass       891 non-null    int64
 2   sex          891 non-null    object
 3   age          714 non-null    float64
 4   sibsp        891 non-null    int64
 5   parch        891 non-null    int64
 6   fare         891 non-null    float64
 7   embarked     889 non-null    object
 8   class        891 non-null    category
 9   who          891 non-null    object
 10  adult_male   891 non-null    bool
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object
 13  alive        891 non-null    object
 14  alone        891 non-null    bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```


In [25]:
```
print(df.head())
```

```
   survived  pclass     sex   age  sibsp  parch     fare embarked  class  \
0         0       3    male  22.0      1      0   7.2500        S  Third
1         1       1  female  38.0      1      0  71.2833        C  First
2         1       3  female  26.0      0      0   7.9250        S  Third
3         1       1  female  35.0      1      0  53.1000        S  First
4         0       3    male  35.0      0      0   8.0500        S  Third

     who  adult_male deck  embark_town alive  alone
0    man        True  NaN  Southampton    no  False
1  woman       False    C    Cherbourg   yes  False
2  woman       False  NaN  Southampton   yes   True
3  woman       False    C  Southampton   yes  False
4    man        True  NaN  Southampton    no   True
```


In [26]:
```
# pd.crosstab(index, columns)
table = pd.crosstab(df['sex'], df['survived'])
print(table)
```

```
survived    0    1
sex
female     81  233
male      468  109
```

In [27]:
```
# 1. 가설설정
# H0 : 성별과 생존 여부는 서로 관련이 없다(두 변수는 서로 독립이다)
```

```
# H1 : 성별과 생존 여부는 서로 관련이 있다(두 변수는 서로 독립이 아니다)
```

In [28]:
```
# 2. 유의수준 확인 : 유의수준 5%로 확인
```

In [29]:
```
# 3. 검정실시(통계량, p-value, 기대빈도 확인)
from scipy.stats import chi2_contingency

# 위와 같이 교차표 만들어서 입력
statistic, pvalue, dof, expected = chi2_contingency(table)
# 공식문서상에 : statistic(통계량), pvalue, dof(자유도), expected_freq(기대빈도)

print(statistic)
print(pvalue)
print(dof)   # 자유도 = (행-1)*(열-1)
print(np.round(expected, 2) ) # array 형태 : 반올림하고 싶다면 np.round()
```



```
260.71702016732104
1.1973570627755645e-58
1
[[193.47 120.53]
 [355.53 221.47]]
```

In [30]:
```
# 4. 귀무가설 기각여부 결정(채택/기각)
# p-value 값이 0.05보다 작기 때문에 귀무가설을 기각한다.
# 즉, 성별과 생존여부는 서로 관련이 있다고 할 수 있다.

# 답 : 기각
```

* 데이터를 변경해보면서 이해해봅시다.

In [31]:
```
# 임의 데이터 생성
sex, survived = [160, 160], [250, 220]
table = pd.DataFrame([sex, survived], columns=['0','1'], index=['female', 'male'])
print(table)
```

```
          0    1
female  160  160
male    250  220
```

In [32]:
```
# 3. 검정실시(통계량, p-value, 기대빈도 확인)
from scipy.stats import chi2_contingency

# 위와 같이 교차표 만들어서 입력
statistic, pvalue, dof, expected = chi2_contingency(table)
# 공식문서상에 : statistic(통계량), pvalue, dof(자유도), expected_freq(기대빈도)

print(statistic)
print(pvalue)
print(dof)   # 자유도 = (행-1)*(열-1)
print(np.round(expected, 2) ) # array 형태 : 반올림하고 싶다면 np.round()
```



```
0.6541895872879862
0.41861876333789727
1
[[166.08 153.92]
 [243.92 226.08]]
```

In [33]:
```
# 4. 귀무가설 기각여부 결정(채택/기각)
# p-value 값이 0.05보다 크기 때문에 귀무가설을 채택한다.
# 즉, 성별과 생존여부는 서로 관련이 없다고 할 수 있다.

# 답 : 채택
```