

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

```

from pandas import read_excel, melt
from scipy.stats import shapiro, normaltest, ks_2samp, bartlett, fligner
from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm
import sys
import os

sys.path.append(os.path.dirname(os.path.dirname(os.getcwd())))
from helper import equal_variance_test

```

데이터 가져오기

어떤 지역에서 동일한 제품을 판매하고 있는 두 가게에 대한 일별 매출액

```

df = read_excel("https://data.hossam.kr/E02/store.xlsx")
df

```

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

	store1	store2
0	46	78
1	47	57
2	58	31
3	47	28
4	27	67
5	58	77
6	56	36
7	26	57
8	47	36
9	25	57

#02. 분산분석의 조건 충족 여부 검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

샘플의 수가 적을 때 정규성을 확인하는 검정

(샘플의 수가 대략 50개 미만인 경우, 중심극한 정리는 30개 미만을 권장하기도 함)

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

가설	내용
귀무가설	집단간 데이터 분포에는 차이가 없다(정규성을 따름)
대립가설	집단간 데이터 분포에는 차이가 있다(정규성을 따르지 않음)

```
shapiro(df['store1'])
```

```
ShapiroResult(statistic=0.8321117162704468, pvalue=0.035477906465530396)
```

```
shapiro(df['store2'])
```

```
ShapiroResult(statistic=0.8993193507194519, pvalue=0.21535511314868927)
```

2) Normal Test

```
normaltest(df['store1'])
```

```
c:\Users\leekh\AppData\Local\Programs\Python\Python311\Lib\site-packages  
warnings.warn("kurtosistest only valid for n ≥ 20 ... continuing ")
```

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

```
NormaltestResult(statistic=1.9891717867491527, pvalue=0.369876581177278)
```

```
normaltest(df['store2'])
```

```
NormaltestResult(statistic=2.081349912435389, pvalue=0.3532161960163575)
```

3) 콜모고로프-스미르노프 검정(Kolmogorov-Smirnov test)

정규분포에 국한되지 않고 두 표본이 같은 분포를 따르는지 확인할 수 있는 방법

한 번에 두 개씩 검사 가능

```
ks_2samp(df['store1'], df['store2'])
```

```
KstestResult(statistic=0.4, pvalue=0.41752365281777043, statistic_locati
```

2. 데이터의 등분산성 검정

1) Bartlett 검정

집단간 분산이 같은지 다른지 여부를 알아볼 때 사용

독립 2표본 t-검정 또는 일원분산분석(one-way ANOVA) 실시 전에 등분산성을 확인하는 용도

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

Bartlett 검정은 두 집단 뿐만 아니라 세 집단 이상에서도 사용할 수 있음

모든 변수가 정규분포를 따른다는 가정 하에서만 사용 가능함

가설	내용
귀무가설	집단간 분산이 차이가 없다(같다)
대립가설	집단간 분산이 차이가 있다(다르다)

```
bartlett(df['store1'], df['store2'])
```

```
BartlettResult(statistic=1.0488412011085946, pvalue=0.305774119649436)
```

2) fligner 검정

Fligner-Killeen test는 비모수 등분산 검정으로 각 독립 표본들이 정규분포를 따르지 않아도 사용한
검정 방법

가설	내용
귀무가설	집단간 분산이 차이가 없다(같다)
대립가설	집단간 분산이 차이가 있다(다르다)

```
fligner(df['store1'], df['store2'])
```

```
FlignerResult(statistic=1.3014081560908837, pvalue=0.2539561678380817)
```

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부 검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프 검정(Kolmogorov-Smirnov test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한 분산분석

해석

3) 레빈 검정(Levene's test)

통계학에서 등분산성(homoscedasticity)을 검증하기 위해 사용되는 방법

분석하려는 데이터의 그룹이 두 개 이상인 경우에 사용

다른 등분산성 검정 방법과 달리 레빈 검정은 정규성 가정을 거의 하지 않기 때문에 비모수적인 방법으로 적용할 수 있다.

가설	내용
귀무가설	집단간 분산이 차이가 없다(같다)
대립가설	집단간 분산이 차아가 있다(다르다)

```
levene(df['store1'], df['store2'])
```

```
LeveneResult(statistic=1.333315753388535, pvalue=0.2633161881599037)
```

함수로 한번에 처리하기

```
equal_variance_test(df['store1'], df['store2'])
```

	statistic	p-value	equal-var
Bartlett	1.048841	0.305774	True
Fligner	1.301408	0.253956	True

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

	statistic	p-value	equal-var
Levene	1.333316	0.263316	True

#04. 독립성 검정

```
chi2_contingency(df[['store1', 'store2']])
```

```
Chi2ContingencyResult(statistic=64.44306604494015, pvalue=1.851233643894
[47.29240375, 56.70759625],
[40.47138398, 48.52861602],
[34.10509886, 40.89490114],
[42.74505723, 51.25494277],
[61.38917794, 73.61082206],
[41.83558793, 50.16441207],
[37.74297607, 45.25702393],
[37.74297607, 45.25702393],
[37.28824142, 44.71175858]))
```

#03. scipy.stats 패키지를 사용한 분산분석

```
f_oneway(df['store1'], df['store2'])
```

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

```
F_onewayResult(statistic=1.4591624718860445, pvalue=0.24269553293319623)
```

해석

p-value 가 0.05보다 크므로 귀무가설을 기각할 수 없다.

즉, 두 가게의 일별 매출 평균은 같다.

#04. statsmodels 패키지를 사용한 일원분산분석

데이터 전처리

```
df2 = df.melt(var_name='store', value_name='sales')
df2
```

	store	sales
0	store1	46
1	store1	47
2	store1	58
3	store1	47
4	store1	27
5	store1	58

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

	store	sales
6	store1	56
7	store1	26
8	store1	47
9	store1	25
10	store2	78
11	store2	57
12	store2	31
13	store2	28
14	store2	67
15	store2	77
16	store2	36
17	store2	57
18	store2	36
19	store2	57

```
df2['store'] = df2['store'].astype('category')
df2.dtypes
```

One-way ANOVA (일원분산분석)

#01. 작업 준비

패키지 가져오기

데이터 가져오기

#02. 분산분석의 조건 충족 여부
검사

1. 데이터의 정규성 검정

1) shapiro wilk 검정

2) Normal Test

3) 콜모고로프-스미르노프
검정(Kolmogorov-Smirnov
test)

2. 데이터의 등분산성 검정

1) Bartlett 검정

2) fligner 검정

3) 레빈 검정(Levene's
test)

함수로 한번에 처리하기

#04. 독립성 검정

#03. scipy.stats 패키지를 사용한
분산분석

해석

```
store    category
sales      int64
dtype: object
```

```
lm = ols('sales ~ C(store)', data=df2).fit()
anova_lm(lm)
```

	df	sum_sq	mean_sq	F	PR(>F)
C(store)	1.0	378.45	378.450000	1.459162	0.242696
Residual	18.0	4668.50	259.361111	NaN	NaN