기초 통계 분석

기초 통계 분석은 데이터의 특성을 요약하고 해석하는 과정을 의미합니다. 주요한 통계 지표를 계산하고, 그래프를 생성하여 데이터를 이해하는 데 도움을 줍니다.

기술 통계 분석은 데이터의 중심 경향성, 퍼짐 정도 등을 설명하는 통계 지표를 계산하는 과정입니다. 기술 통계 분석의 예시로는 다음과 같은 지표들이 있습니다

- 평균 (Mean)
- 중앙값 (Median)
- 최빈값 (Mode)
- 표준편차 (Standard Deviation)
- 범위 (Range)

Scipy

과학적 및 기술적 컴퓨팅을 위한 파이썬 라이브러리

- Scipy는 수치 계산을 위한 확장 라이브러리로, 선형 대수, 최적화, 신호 처리, 통계 등 다양한 과학적 계산에 사용됩니다.
- 오픈 소스 프로젝트로서 NumPy와 함께 동작하며, 수학과 공학 분야의 복잡한 문제를 해결하는 데 도움을 줍니다.

Scipy 사용하는 이유?

- 다양한 기능의 통합:
  > Scipy를 사용하면 여러 개의 라이브러리를 사용하는 것보다 효율적이며, 일관된 인터페이스로 다양한 기능을 사용할 수 있습니다.
- 확장 가능성:
  > 모듈별로 필요한 기능만을 선택적으로 사용할 수 있으며, 자신만의 기능을 추가하여 사용할 수 있습니다.
- 생산성 향상:
  > 복잡한 수학적 계산을 쉽고 빠르게 수행할 수 있어, 프로그래머의 생산성을 향상시킵니다.

Scipy.stats

Scipy.stats는 확률 분포, 가설 검정, 상관 분석, 회귀 분석 등 통계 분석을 위한 다양한 기능을 제공합니다.

주요 기능:

- 확률 분포 모델링: 정규 분포, 이항 분포, 포아송 분포 등 다양한 분포에 대한 모델링과 시뮬레이션 가능
- 가설 검정: t-검정, 카이제곱 검정 등 통계적 가설 검정을 위한 함수 제공
- 상관 및 회귀 분석: 데이터 간의 상관 관계 분석과 선형 회귀 모델 구축 가능

Scipy.stats의 실무 활용 사례

- 데이터 분석 및 모델링:
  > > Scipy.stats를 사용하여 데이터를 분석하고 모델을 구축하여 의사 결정 및 예측에 활용합니다.
- 품질 관리 및 생산성 향상:
  > > 제품의 결함률을 분석하거나 생산 과정의 효율성을 평가하는 데 사용됩니다.
- 금융 및 마케팅:
  > > 금융 데이터 분석, 고객 행동 예측, 마케팅 전략 수립 등 다양한 분야에서 Scipy.stats의 통계 기능이 활용됩니다.

Scipy 설치 방법

In [1]:

```
Requirement already satisfied: scipy in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (1.10.0)
Requirement already satisfied: numpy<1.27.0,>=1.19.5 in /Users/youngjinseo/anaconda3/lib/python3.10/site-packages (from scipy) (1.23.5)
```

라이브러리 불러오기

In [2]:

```python
import numpy as np
from scipy import stats
```

가설 검정

가설 검정은 통계적 가설을 세우고, 주어진 데이터를 분석하여 가설이 참인지 아닌지를 결정하는 과정입니다. 가설 검정은 주어진 데이터에 대한 추론을 수행하는 데 사용됩니다. 가설 검정의 예시로는 t-검정, 카이제곱 검정, 회귀 분석 등이 있습니다.

- 두 집단 평균의 비교(예: 약물 효과)
- 표본과 모집단의 평균 비교

주요 함수

- scipy.stats.ttest_1samp: 한 표본 t-검정
- scipy.stats.ttest_ind: 두 표본 t-검정
- scipy.stats.ttest_rel: 대응 표본 t-검정

In [2]:

```python
# 두 집단의 데이터
group1 = [20, 22, 19, 24, 21]
group2 = [30, 31, 29, 32, 28]

# 두 집단 평균의 차이에 대한 t-검정
t_stat, p_value = stats.ttest_ind(group1, group2)
print(f"t-statistic: {t_stat}, p-value: {p_value}")
```

```
t-statistic: -7.9026332891780955, p-value: 4.768714309095669e-05
```

이 코드에서 사용된 ttest_ind 함수는 두 독립된 집단 간의 평균 차이를 검정하는 데 사용됩니다. 결과는 두 가지 값, 즉 t-통계량(t-statistic)과 p-값(p-value)으로 제공됩니다.

T-통계량 (t-statistic)

- 정의: 두 집단의 평균 차이를 표본 내 변동성으로 나눈 값입니다.
- 의미: t-통계량이 클수록 두 집단의 평균 차이가 통계적으로 의미가 있음을 나타냅니다. 일반적으로 t-통계량이 크면 두 집단 간의 차이가 더 크다는 것을 의미합니다.

p-값 (p-value)

- 정의:관측된 데이터와 같거나 더 극단적인 결과가 우연에 의해 발생할 확률입니다.
- 의미: p-값이 작을수록 두 집단 간의 평균 차이가 우연에 의해 발생할 가능성이 낮습니다. 일반적으로 p-값이 0.05보다 작으면 두 집단 간의 평균 차이가 통계적으로 유의미하다고 결론짓습니다.

예시 결과 해석

t-통계량: -7.9026

- 이 값은 두 집단 간의 평균 차이가 상당히 크다는 것을 나타냅니다.
- 음수 값은 첫 번째 집단의 평균이 두 번째 집단의 평균보다 낮다는 것을 의미합니다.

p-값: 3.36e-05 (약 0.0000336)

- p-값이 매우 작기 때문에, 두 집단의 평균 차이는 통계적으로 유의미하다고 볼 수 있습니다.
- 즉, 두 집단 간의 평균 차이가 우연에 의해 발생할 가능성은 매우 낮습니다.

두 집단의 평균 차이는 통계적으로 유의미하다고 결론짓고, 집단 간의 평균에 실질적인 차이가 있다고 해석할 수 있습니다.

연습문제

A 약을 복용한 그룹과 B 약을 복용한 그룹의 평균 혈압 차이가 있는지 t-검정을 통해 확인하세요. A 그룹의 데이터: [120, 122, 119, 121, 123], B 그룹의 데이터: [130, 128, 132, 129, 131].

In [ ]:

상관분석

상관 분석은 두 변수 간의 관계를 파악하는 통계적 분석 방법입니다. 상관 분석을 통해 두 변수 사이의 선형적인 관계, 상관 계수 등을 계산할 수 있습니다. 상관 분석의 예시로는 피어슨 상관 계수, 스피어만 상관 계수, 켄달의 순위 상관 계수 등이 있습니다.

피어슨 상관 계수 (Pearson Correlation Coefficient)

- 활용 예시:
  > 두 연속형 변수 간의 상관 관계 파악
  > > 예측 모델의 독립 변수 선택

scipy.stats.pearsonr: 피어슨 상관 계수와 p-value 계산

In [4]:

```python
# 두 변수 데이터
x = [1, 2, 3, 4, 5]
y = [2, 4, 6, 8, 10]

# 피어슨 상관 계수 계산
corr_coef, p_value = stats.pearsonr(x, y)
print(f"Pearson correlation coefficient: {corr_coef}, p-value: {p_value}")
```

```
Pearson correlation coefficient: 1.0, p-value: 0.0
```

피어슨 상관 계수 (Pearson correlation coefficient)

- 정의: 피어슨 상관 계수는 두 변수 간의 선형 관계의 강도와 방향을 측정하는 지표입니다.
- 값의 범위: -1에서 1까지의 값을 가집니다.
  > 1: 완벽한 양의 선형 관계 (한 변수가 증가하면 다른 변수도 일정한 비율로 증가) 0: 선형 관계가 없음 (두 변수 간에 직선 관계가 거의 없음) -1: 완벽한 음의 선형 관계 (한 변수가 증가하면 다른 변수는 일정한 비율로 감소)
- 해석: 이 경우, 피어슨 상관 계수가 1.0입니다.
  > 이는 두 변수 x와 y가 완벽한 양의 선형 관계를 가진다는 것을 의미합니다. 즉, x가 증가할 때 y도 일정한 비율로 정확히 증가합니다. 이 경우 y=2x의 관계가 명확하게 보입니다.

p-값 (p-value)

- 정의: p-값은 관찰된 상관 계수가 귀무 가설 하에서 우연히 발생할 확률을 나타냅니다.
- 귀무 가설 (H0): 두 변수 사이에 상관 관계가 없다.
- p-값의 해석:
  > 0.0: 이는 매우 강한 증거로, 상관 계수가 우연에 의해 발생했을 가능성이 거의 없음을 의미합니다.
  > > 통계적으로 유의한 결과로 간주되며, 보통 p-값이 0.05보다 작으면 귀무 가설을 기각합니다.

결론

- Pearson correlation coefficient가 1.0이므로 두 변수는 완벽한 양의 선형 관계를 가집니다.
- p-value가 0.0이므로 이 상관 관계는 통계적으로 유의미합니다.
- 따라서,x와 y 사이에 강한 양의 선형 관계가 있음을 확인할 수 있습니다.

연습문제

학생들의 공부 시간(x)과 시험 점수(y) 사이의 상관 관계를 피어슨 상관 계수를 이용하여 분석하세요. 공부 시간: [2, 3, 4, 5, 6], 시험 점수: [70, 75, 80, 85, 90].

In [ ]:

카이제곱 검정 (Chi-Square Test)

카이제곱 검정은 통계학에서 두 가지 주요한 유형의 분석을 위해 사용되는 검정입니다. 이 검정은 관찰된 데이터와 기대되는 데이터 간의 차이를 측정하여, 두 변수 사이의 독립성을 확인하거나, 관찰된 분포가 특정 이론적 분포와 일치하는지를 판단하는 데 사용됩니다.

- 범주형 데이터의 독립성 검정
- 기대 빈도와 관찰 빈도의 차이 검정

scipy.stats.chi2_contingency: 카이제곱 검정

In [3]:

```python
# 관찰 데이터
data = [[10, 20, 30],
        [20, 25, 15]]

# 카이제곱 검정
chi2, p, dof, expected = stats.chi2_contingency(data)
print(f"Chi-square statistic: {chi2}, p-value: {p}")
```

```
Chi-square statistic: 8.88888888888889, p-value: 0.011743628457021359
```

- 카이제곱 통계량 (Chi-square statistic) : 카이제곱 통계량은 관찰된 빈도와 기대 빈도의 차이를 측정하는 지표입니다.
- p-값 (p-value): p-값은 관찰된 카이제곱 통계량이 귀무 가설 하에서 우연히 발생할 확률을 나타냅니다.
- 자유도 (Degrees of Freedom, dof) : 자유도는 카이제곱 분포의 모양을 결정하는 데 사용됩니다.
- 기대 빈도 (Expected Frequencies): 각 셀에 대해 귀무 가설이 참일 때 예상되는 빈도입니다.

결과 해석

- 카이제곱 통계량은 8.89이면, 이는 관찰된 데이터와 기대 빈도 간의 차이가 통계적으로 유의미한지 여부를 결정하는 데 사용됩니다.
- p-값의 해석
  > p-값이 0.0117로, 일반적인 유의수준(α)인 0.05보다 작습니다.
  >
  > 이는 관찰된 데이터가 귀무 가설 하에서 발생할 확률이 매우 낮다는 것을 의미합니다.
  >
  > 따라서, 귀무 가설을 기각할 수 있으며, 이는 두 변수 간에 연관성이 있다고 결론지을 수 있습니다.

두 변수는 통계적으로 유의미한 연관성이 있으며, 이는 관찰된 데이터와 기대 빈도 간에 유의한 차이가 있음을 의미합니다.

연습문제

두 종류의 광고(A, B)에 대한 소비자 반응(긍정, 중립, 부정)을 카이제곱 검정을 통해 비교하세요. A 광고: [40, 35, 25], B 광고: [50, 30, 20].

In [ ]:

회귀 분석

회귀 분석은 종속 변수와 한 개 이상의 독립 변수 간의 관계를 모델링하는 데 사용되는 통계적 분석 기법입니다. 회귀 분석은 종속 변수의 값을 예측하는 모델을 구축하고, 독립 변수의 영향력과 관련성을 평가하는 데 사용됩니다. 이를 통해 변수 간의 상관 관계를 이해하고, 미래의 결과를 예측하는 데 도움을 줍니다.

선형 회귀 분석

선형 회귀 분석은 종속 변수와 독립 변수 간의 선형 관계를 모델링하는 가장 일반적인 회귀 분석 기법입니다. 선형 회귀 분석은 독립 변수의 가중치와 절편을 추정하여 종속 변수의 값을 예측하는 선형 모델을 생성합니다. 이를 통해 독립 변수의 변화에 따른 종속 변수의 변화를 예측할 수 있습니다.

In [2]:

```python
# 데이터 생성
x = np.array([1, 2, 3, 4, 5])
y = np.array([2, 4, 6, 8, 10])

# 선형 회귀 분석
slope, intercept, r_value, p_value, std_err = stats.linregress(x, y)

print(f"Slope: {slope}")
print(f"Intercept: {intercept}")
print(f"R-value: {r_value}")
print(f"P-value: {p_value}")
print(f"Standard Error: {std_err}")

# 회귀 직선 예측
predicted_y = slope * x + intercept
print(f"Predicted y: {predicted_y}")
```

```
Slope: 2.0
Intercept: 0.0
R-value: 1.0
P-value: 1.2004217548761408e-30
Standard Error: 0.0
Predicted y: [ 2.  4.  6.  8. 10.]
```

결과 해석

- 기울기(Slope)_2.0 : 기울기는 독립 변수 x가 한 단위 증가할 때 종속 변수 y가 얼마나 증가하는지를 나타내고, 이 경우 x가 1 증가할 때 y는 2 증가합니다.
- 절편(Intercept)_0.0: 절편은 x가 0일 때의 y값이므로, 이 경우 절편이 0이므로 회귀선은 원점을 통과합니다.
- 상관 계수 (R-value)_ 1.0 : -1과 1 사이의 값을 가지며, 1에 가까울수록 강한 양의 선형 관계를 의미합니다.여기서 R-value가 1.0이므로 x와 y사이에는 완벽한 양의 선형 관계가 있습니다.
- P-value_ 1.20042: 이 경우 p-값이 거의 0에 가까우므로, 기울기가 0이라는 귀무 가설을 기각할 수 있습니다.
- 표준 오차(Standard Error)_0.0 : 표준 오차는 회귀 직선의 기울기의 불확실성을 나타냅니다.값이 0.0이므로, 회귀 모델의 기울기에 대한 불확실성이 전혀 없음을 의미합니다.
- 회귀 직선 예측 (Predicted y) : 주어진 x값에 대해 선형 휘귀식 y=2x+0을 사용하여 y값을 예측합니다. 실제 y 값 [2,4,6,8,10]과 완벽하게 일치합니다.

결론

- 완벽한 선형 관계: x와 y 사이에 완벽한 양의 선형 관계가 있습니다.
- 모델의 유의성: p-값이 거의 0에 가까워, 이 관계는 통계적으로 매우 유의미합니다.
- 오차 없음: 표준 오차가 0.0이므로, 회귀 모델이 데이터에 대해 완벽한 적합성을 보입니다.
- 이 결과는 y=2x라는 함수적 관계를 정확하게 나타냅니다.

연습문제

1. 주택 크기(x)와 가격(y) 사이의 선형 회귀 모델을 생성하고, 크기가 85 제곱미터인 주택의 가격을 예측하세요. 주택 크기: [50, 60, 70, 80, 90], 가격: [3000, 3500, 4000, 4500, 5000].

In [ ]:

배운 내용을 가지고 데이터 분석해보세요.

데이터셋 : Fast Food Marketing Campaign (출처 : IBM Watson Analytics Marketing Campaign)

시나리오

패스트푸드 체인이 새로운 메뉴 항목을 추가할 계획입니다. 그러나 새 제품을 홍보하기 위한 세 가지 마케팅 캠페인 중에서 어느 것을 사용할지 결정하지 못했습니다. 판매에 가장 큰 영향을 미치는 프로모션을 결정하기 위해 새로운 메뉴를 무작위로 선택된 여러 시장의 매장에서 출시합니다. 각 매장에서는 다른 프로모션을 사용하며, 첫 4주 동안 새 메뉴의 주간 판매량을 기록합니다.

목표

A/B 테스트 결과를 평가하고, 어떤 마케팅 전략이 가장 효과적인지 결정하세요.

데이터셋 설명

- MarketID: 시장의 고유 식별자
- MarketSize: 판매에 따른 시장 영역 크기
- LocationID: 매장 위치의 고유 식별자
- AgeOfStore: 매장 운영 연수 (연도 단위)
- Promotion: 테스트된 세 가지 프로모션 중 하나
- week: 프로모션이 진행된 4주 중 한 주
- SalesInThousands: 특정 LocationID, Promotion, 및 주의 매출액 (천 단위)

In [8]:

```python
from scipy import stats
import seaborn as sns
import koreanize_matplotlib
import pandas as pd
```

In [10]:

```python
df = pd.read_csv("WA_Marketing-Campaign.csv")
```

In [11]:

```python
# 데어터셋 불러오기
# 결측치 확인을 위한  info 사용하기
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 548 entries, 0 to 547
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   MarketID          548 non-null    int64  
 1   MarketSize        548 non-null    object 
 2   LocationID        548 non-null    int64  
 3   AgeOfStore        548 non-null    int64  
 4   Promotion         548 non-null    int64  
 5   week              548 non-null    int64  
 6   SalesInThousands  548 non-null    float64
dtypes: float64(1), int64(5), object(1)
memory usage: 30.1+ KB
```

In [12]:

```python
#데어터셋 보기
df.head(4)
```

Out[12]:

	MarketID	MarketSize	LocationID	AgeOfStore	Promotion	week	SalesInThousands
0	1	Medium	1	4	3	1	33.73
1	1	Medium	1	4	3	2	35.67
2	1	Medium	1	4	3	3	29.03
3	1	Medium	1	4	3	4	39.25

AB test란?

A/B 테스트는 웹사이트, 앱, 광고, 이메일 마케팅 등의 다양한 디지털 제품과 마케팅 전략의 효과를 측정하기 위해 사용되는 실험 방법입니다. 이를 통해 두 가지 변형(A와 B)을 비교하여 어떤 변형이 더 나은 결과를 내는지 판단할 수 있습니다.

In [16]:

In [21]:

```python
# Promotion 1 VS Promotion 2
```

t-값이 높고 p-값이 0.05보다 작으므로, 95% 신뢰 구간에서 그룹 간 통계적 차이가 없다는 귀무가설이 기각됩니다. 따라서 프로모션 1과 프로모션 2는 유의미한 차이가 있으며, 전자가 후자보다 더 나은 성과를 보였습니다.

In [22]:

```python
#Promotion 1 VS Promotion 3
```

이 경우 t-값이 그리 높지 않고 p-값이 0.05보다 큽니다. 이는 이 시나리오가 우연히 발생할 확률이 높다는 것을 시사합니다. 따라서 95% 신뢰 구간에서 그룹 간 통계적 차이가 없다는 귀무가설을 기각하지 않습니다. 프로모션 1이 프로모션 3보다 평균 매출액이 더 높지만, 프로모션 1과 프로모션 3 사이에 유의미한 차이가 없습니다.

결과

프로모션 1과 프로모션 3은 프로모션 2보다 성과가 좋지만, 그들 사이의 차이는 통계적으로 유의미하지 않습니다.