

Module5

ANOVA



◆ 학습목표

Pyhon을 이용해 ANOVA 분석방법을 학습한다.

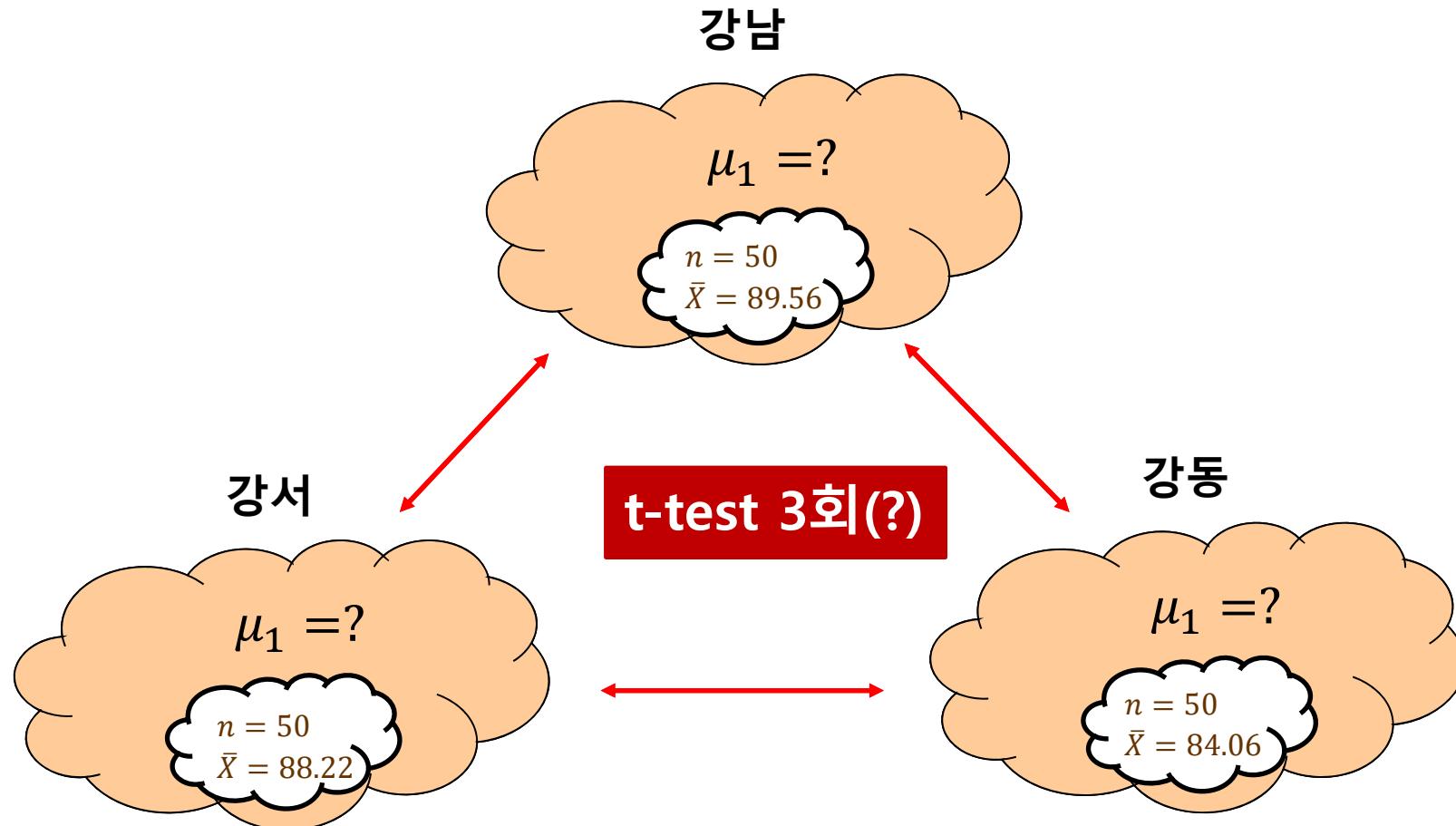
-
- I. One Way ANOVA
 - II. Repeated Measures ANOVA
 - III. Two Way ANOVA
 - IV. Two Way Repeated Measures ANOVA
-

I. One Way ANOVA

분산분석이란

평균의 차이를 보는데 왜 분산분석인가요?
t-test를 여러 번 하면 되지 않나요?

- ❖ 모집단에서 표본추출



❖ 다중검정

- t-test를 여러 번 하기 위한 귀무가설 (3개의 귀무가설)

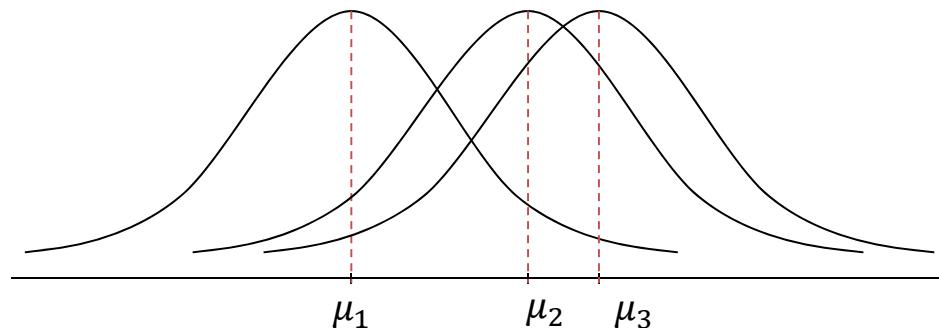
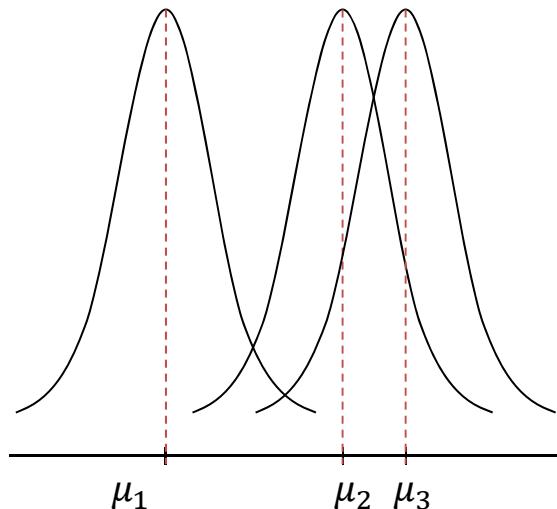
$$H_{0_1}: \mu_1 = \mu_2, H_{0_2}: \mu_1 = \mu_3, H_{0_3}: \mu_2 = \mu_3$$

- 여러 번 t-test를 해주게 되면 1종 오류(α)를 범할 확률이 증가
 - Bonferroni inequality
- $$P(H_{1_1} \cap H_{1_2} \cap H_{2_3}) = \max(n\alpha) = \max(3\alpha) = \max(15\%)$$
- 최대 15%까지 가능
 - 집단이 3개 이상일 때는 모든 집단의 평균이 같은지를(1=2=3) 분산을 이용하여 분석

* 평균의 차이를 분석하는 것인데 왜 분산을 이용하지요?

- ❖ ANOVA(Analysis of Variance)

- 그룹내 분산의 비율을 통해 평균의 차이를 비교할 수 있음
- 분산분석을 실시한 후에 집단 간에 평균이 차이가 있다고 결과가 나타나면, 각 집단간 평균을 비교
- Post-hoc 분석: Bonferroni, Duncan, Tukey 등 오류를 보정한 방법 이용



❖ 분산분석(ANOVA)

- 영국의 통계학자인 피셔에 의해 개발
- 세 그룹 이상의 분산을 검정하여, 각 집단의 모평균 차이를 비교 검정
- 변동(분산)에 영향을 주는 변수를 몇 개의 레벨로 나누어서 대상에게 실험
- 변동(분산)의 분해: 실험요인들에 의한 변동과 오차에 의한 변동으로 구분

❖ 실험계획법에서 가장 많이 사용되는 분석방법

- 사례) 통닭의 맛을 결정하는 온도는?
- 반응값(특성치) : 통닭 맛
- 요인 : 튀기는 온도
- 요인의 수준 수 : 4
 - 튀기는 온도가 120°C, 140°C, 160°C, 180°C인 경우
- 수준별 반복수 : 30
 - 각 수준의 요인에서 30명이 점수를 매긴 경우 : 30

❖ 일원분산분석

- 완전확률화설계(completely randomized design)

	요인A		
	강남	강서	강동
반응값	93	90	95
	91	89	91

❖ 이원분산분석

- 반복 없음

확률화블록설계(Randomized block design)

라틴방격설계(Latin square design)

- 반복 있음 (상호작용 효과검정)

요인실험(Factorial experiment)

- 범주형+연속형

ANCOVA

	요인B			
	강남	강서	강동	
요인A	A1	93	91	87
	A2	91	95	86
	A3

	요인B			
	강남	강서	강동	
요인A	A1	93 91 ...	90 89 ...	95 91 ...
	A2	87 83 ...	90 91 ...	88 89 ...
	A3

출처: SPSS를 이용한 실험설계와 분산분석, 한상태 외, spss 아카데미 참조

❖ 요인 수준 선택에 따른 ANOVA 종류

- 고정효과모형(fixed effect model) : 요인을 실험자가 선택
- 변량효과모형(random effect model) : 요인을 무작위로 선택
- 혼합모형(mixed effect model): 두 개의 요인 중에서 하나는 실험자가 하나는 무작위로 선택

❖ 예) G레스토랑은 서울에 10개의 매장을 가지고 있다.

- 강남, 강서, 강동의 3개의 매장에 대한 만족도 차이를 보고 싶다.
- 고정효과모형: 3개 매장의 평균에 관심

$$H_0: \mu_1 = \mu_2 = \mu_3$$

- 10개 매장 중에서 3개의 매장을 임의로 선택해서 K레스토랑 전체 만족도의 차이를 보고 싶다.
- 변량효과모형: 분산의 동질성에 대해 관심 (전체 평균에 관심)

$$H_0: \sigma_{\mu}^2 = 0$$

- 매장 간에는 메뉴가 서로 다양하다. 강남, 강서, 강동 3개의 매장을 설정하고 메뉴는 임의로 3 개씩 선택하였다.
- 혼합모형: 매장과 메뉴의 만족도

일원 분산분석

❖ 문제의 정의

- G커피회사는 강남(1), 강동(2), 강서(3)에 매장을 보유하고 있다. 매장별로 고객만족도가 차이가 있는지를 조사하기 위해 매장별 고객만족도를 조사하였다.
- 과연 3곳 매장의 고객만족도는 차이가 있는지? 있다면 어느 레스토랑의 서비스 만족도가 가장 안 좋은가 확인해보자
- 07_1.OWA.csv

❖ 가설

- 귀무가설(H_0): 3곳 매장의 고객만족도는 차이가 없다.

$$H_0: \mu_1 = \mu_2 = \mu_3$$

- 연구가설(H_1): 3곳 매장의 고객만족도 중 적어도 한 쌍은 차이가 있다.

$$H_1: \text{not } H_0, \mu_1 \neq \mu_2 \text{ or } \mu_1 \neq \mu_3 \text{ or } \mu_2 \neq \mu_3$$

❖ 가설

- 귀무가설(H_0): 3곳 매장의 고객만족도는 차이가 없다.

$$H_0: \mu_1 = \mu_2 = \mu_3$$

- 연구가설(H_1): 3곳 매장의 고객만족도 중 적어도 한 쌍은 차이가 있다.

$$H_1: \text{not } H_0, \mu_1 \neq \mu_2 \text{ or } \mu_1 \neq \mu_3 \text{ or } \mu_2 \neq \mu_3$$

- **다중비교** (Post-hoc)

$$H_{10}: \mu_1 = \mu_2$$

$$H_{11}: \mu_1 \neq \mu_2$$

$$H_{20}: \mu_1 = \mu_3$$

$$H_{21}: \mu_1 \neq \mu_3$$

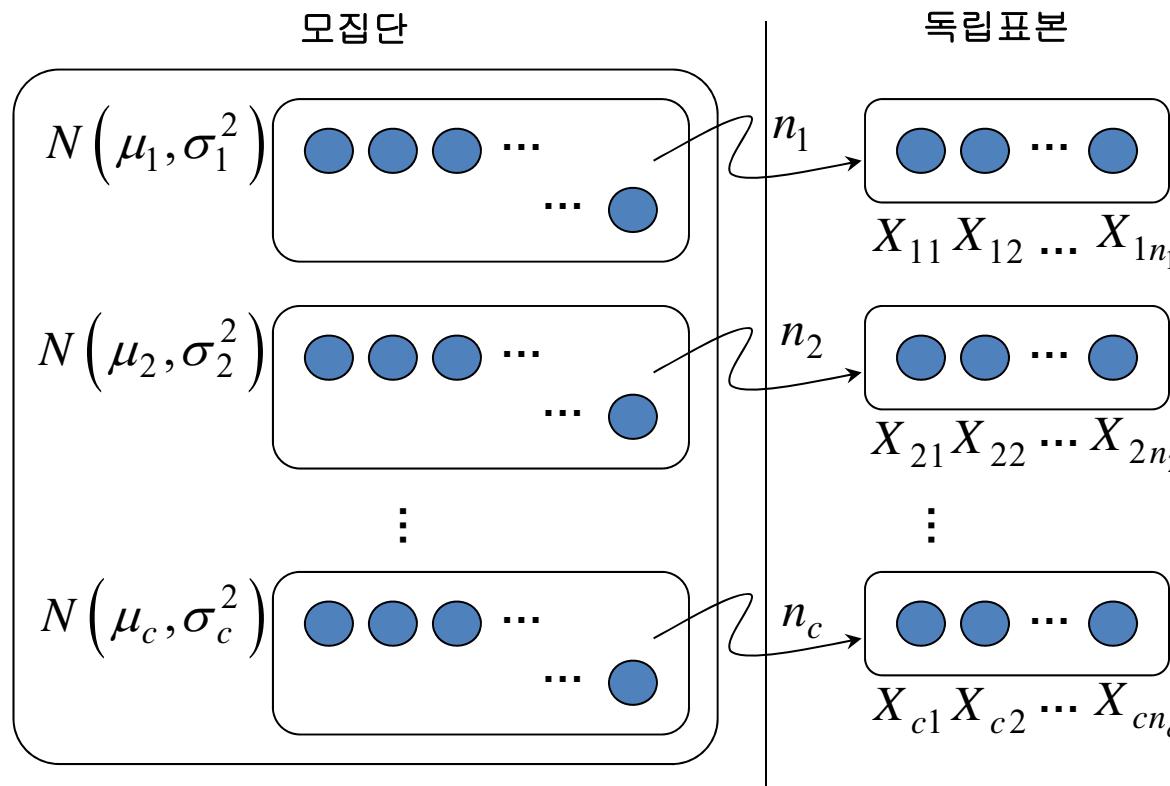
$$H_{30}: \mu_2 = \mu_3$$

$$H_{31}: \mu_2 \neq \mu_3$$

One-Way ANOVA

LGE Internal Use Only

❖ 표본추출

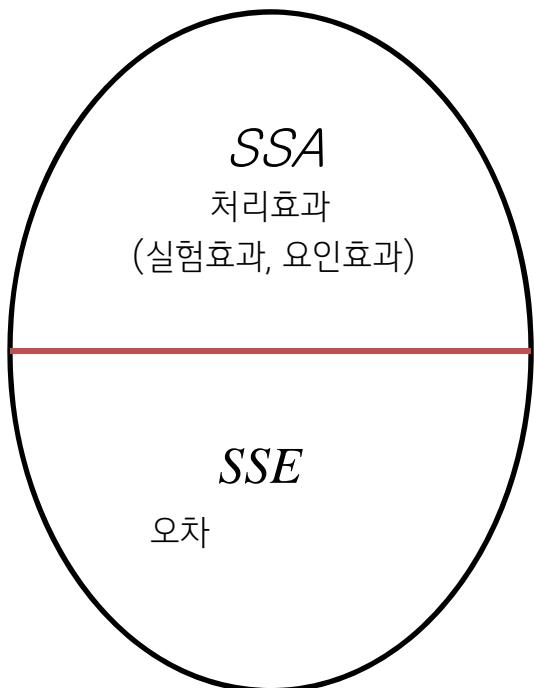


One-Way ANOVA

LGE Internal Use Only

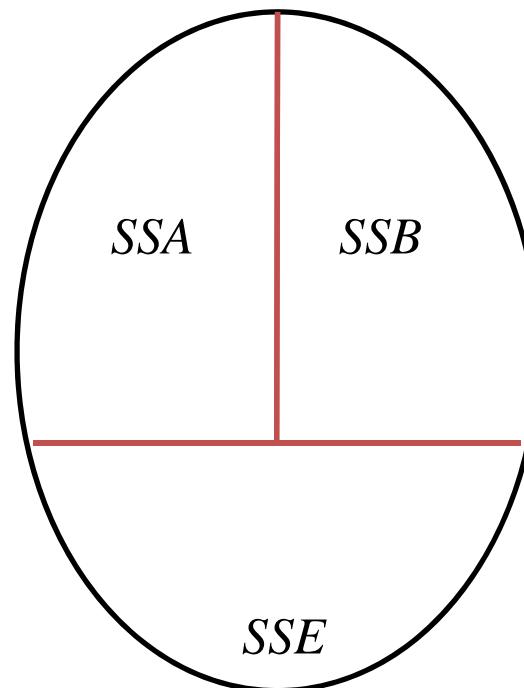
❖ 변동의 분해

일원배치 분산분석



$$SST = SSA + SSE$$

이원배치 분산분석



$$SST = SSA + SSB + SSE$$

❖ 변동(총제곱합)의 분해

- 총편차의 분해

$$y_{ij} - \bar{y} = (\bar{y}_j - \bar{y}) + (y_{ij} - \bar{y}_j)$$

총편차 = 처리효과 + 오차
(실험효과, 요인효과)

- 총제곱합의 분해

$$\sum_i \sum_j (y_{ij} - \bar{y})^2 = \sum_j (\bar{y}_j - \bar{y})^2 + \sum_i \sum_j (y_{ij} - \bar{y}_j)^2$$

$$SST = SSA + SSE$$

총제곱합 = 처리제곱합 + 오차제곱합

Total = between subject + within subject

One-Way ANOVA

LGE Internal Use Only

❖ 변동의 분해

	인자의 수준				
	A_1	A_2	...	A_j	
반응값 (반복실험)	y_{11}	y_{12}	...	y_{1j}	
	y_{21}	y_{22}	...	y_{2j}	
	y_{31}	y_{32}	...	y_{3j}	
	
	
N	n_1	n_2	...	n_j	
합계	T_1	T_2	...	T_j	
평균	\bar{y}_1	\bar{y}_2	...	\bar{y}_j	\bar{y}

One-Way ANOVA

LGE Internal Use Only

❖ 변동의 분해

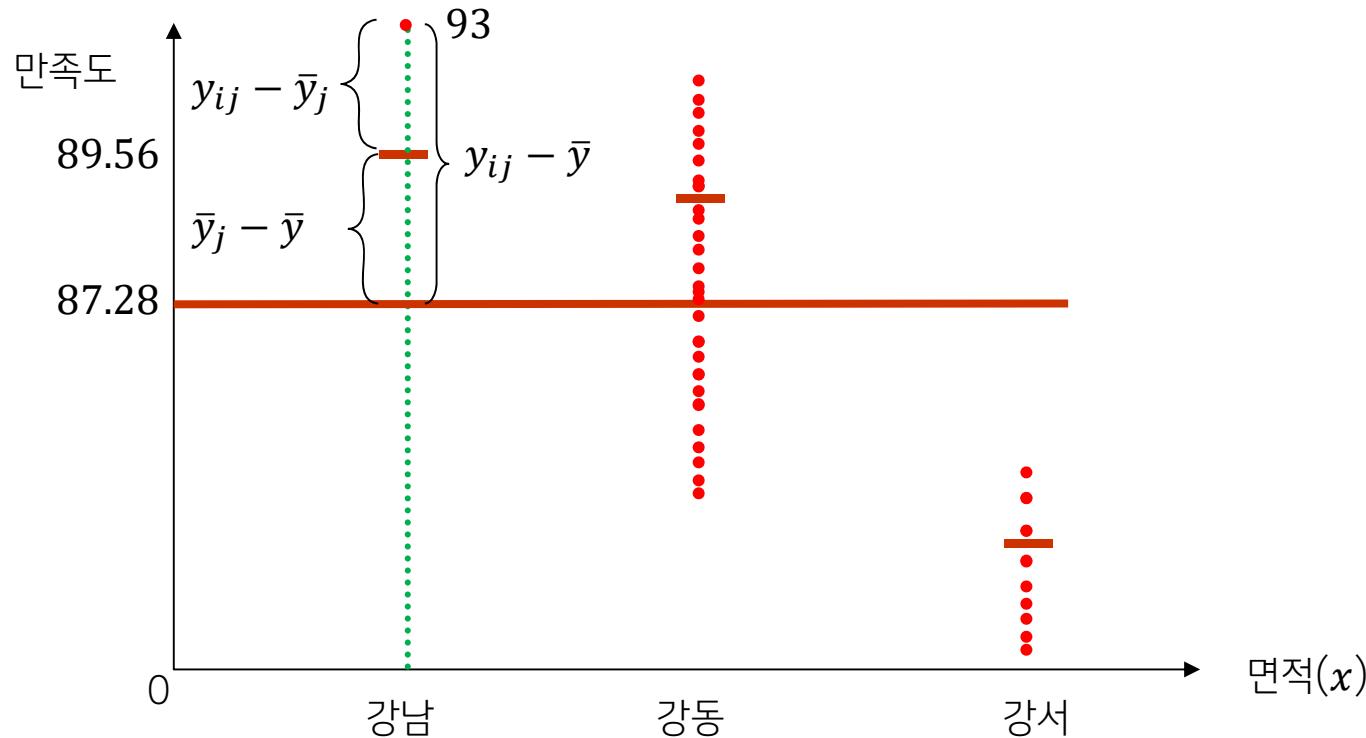
	인자의 수준			
	강남	강동	강서	
반응값 (반복실험)	93 94 95 86 87 ...	85 82 88 83 93 ...	86 85 84 93 90 ...	
N	50	50	50	150
합계	4,478	4,411	4,203	13,092
평균	89.56	88.22	84.06	87.28

$$\begin{aligned} y_{ij} - \bar{y} &= (\bar{y}_j - \bar{y}) + (y_{ij} - \bar{y}_j) \\ 5.72(93 - 87.28) &= 2.28(89.56 - 87.28) + 3.44(93 - 89.56) \\ \text{총편차} &= \text{처리효과} + \text{오차} \end{aligned}$$

One-Way ANOVA

LGE Internal Use Only

❖ 변동의 분해(분산분석)

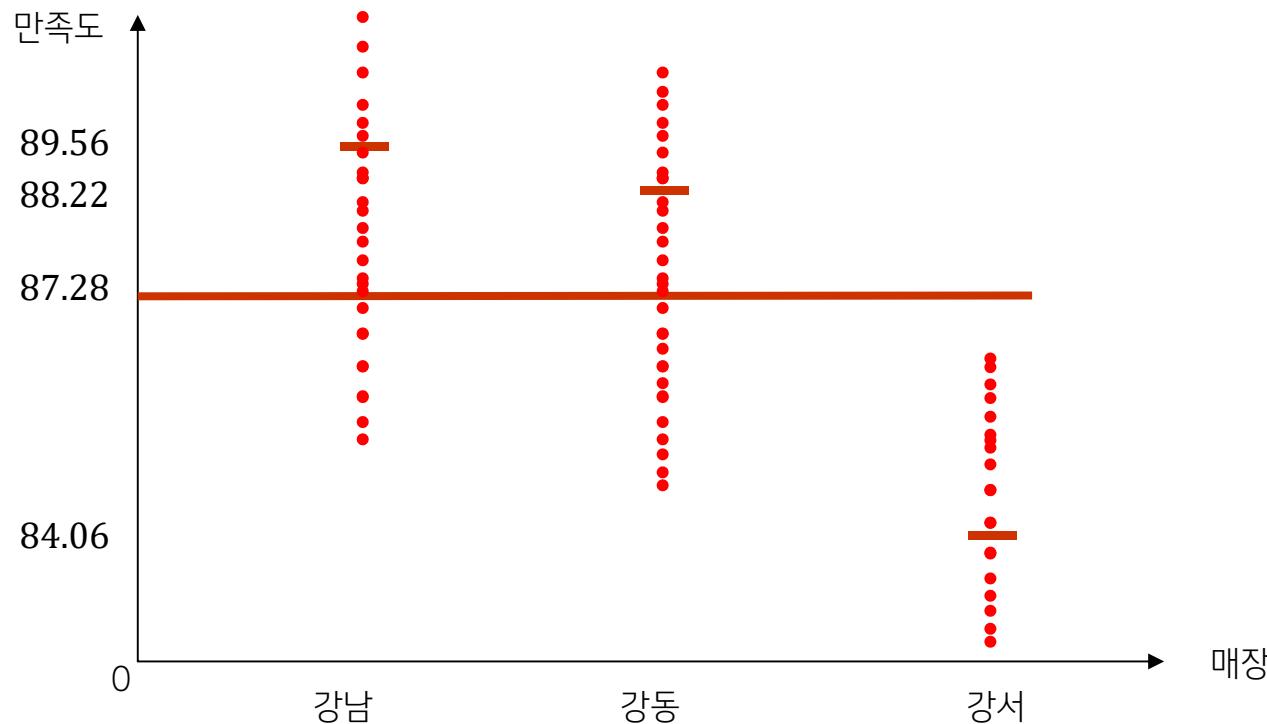


$$\begin{array}{lcl} y_{ij} - \bar{y} & = & (\bar{y}_j - \bar{y}) + (y_{ij} - \bar{y}_j) \\ 5.72(93 - 87.28) & = & 2.28(89.56 - 87.28) + 3.44(93 - 89.56) \\ \text{총편차} & = & \text{처리효과} + \text{오차} \end{array}$$

One-Way ANOVA

LGE Internal Use Only

❖ 변동의 분해(분산분석)



$$\sum_i \sum_j (y_{ij} - \bar{y})^2 = \sum_j (\bar{y}_j - \bar{y})^2 + \sum_i \sum_j (y_{ij} - \bar{y}_j)^2$$

$$\text{총제곱합} = \text{처리제곱합} + \text{오차제곱합}$$

One-Way ANOVA

❖ 분산분석표

요인	제곱합 (SS)	자유도(df)	평균제곱 (MS)	F
처리(SSA)	$SSA = \sum_{j=1}^k (\bar{y}_j - \bar{y})^2$	$k - 1$	$MSA = \frac{SSA}{k - 1}$	
잔차(SSE)	$SSE = \sum_{i=1}^n \sum_{j=1}^k (y_{ij} - \bar{y}_j)^2$	$n - k$	$MSE = \frac{SSE}{n - k}$	$F = \frac{MSA}{MSE}$
총계(SST)	$SST = \sum_{i=1}^n \sum_{j=1}^k (y_{ij} - \bar{y})^2$	$n - 1$		

❖ 검정통계량

$$F = \frac{MSA}{MSE} = \frac{\frac{SSA}{k-1}}{\frac{SSE}{n-k}} \sim F_{(k-1, n-k)}$$

$$F \uparrow = \frac{MSA \uparrow}{MSE \downarrow}$$

$$F \downarrow = \frac{MSA \downarrow}{MSE \uparrow}$$

One-Way ANOVA

❖ 분산분석표

요인	제곱합 (SS)	자유도(df)	평균제곱 (MS)	F
처리(SSA)	822.52	$2(k - 1 = 3 - 1)$	$411.26(= \frac{822.52}{2})$	14.804 $(= \frac{411.26}{27.78})$
잔차(SSE)	4,083.72	$147(n - k = 150 - 3)$	$27.78(= \frac{2,083.72}{147})$	
총계(SST)	4,906.24	$149(n - 1 = 150 - 1)$		

❖ 검정통계량

$$F_{cal} = \frac{411.26}{27.78} = 14.804 > F_{(0.05, 2, 149)} = 3.06$$

❖ p -value 계산

$$p-value = P(F > 14.804) = 0.000 < \alpha = 0.05$$

F분포는 분산(σ^2)을 검증하기 때문에 큰지만 검정: $P(F > 14.804)$

One-Way ANOVA

LGE Internal Use Only

❖ 다중비교

종 류	설 명
Fisher's LSD (least significant difference)	
Bonferroni's MSD	<ul style="list-style-type: none">• t-검정을 사용하여 집단 평균 간 모든 대응별 비교
Seidak	
Scheffe	<ul style="list-style-type: none">• F-검정 사용
R-E-G-W F	<ul style="list-style-type: none">• F-검정 사용 Ryan-Einot-Gabriel-Welsch의 F
R-E-R-W Q	<ul style="list-style-type: none">• Ryan-Einot-Gabriel-Welsch의 Q
S-N-K	<ul style="list-style-type: none">• 스튜던트화 범위 분포를 사용한 평균간 대응비교
Tukey'HSD	<ul style="list-style-type: none">• 평균(스튜던트화) 범위 통계량
Duncan	

One-Way ANOVA

LGE Internal Use Only

❖ 다중비교

종 류	설 명
Fisher's LSD	$t_{\frac{\alpha}{2}(n-k)}$
Bonferroni's MSD	$t_{(\frac{\alpha}{2k}, n-k)} \frac{q_{(\alpha, k, n-k)}}{\sqrt{2}}$
Scheffe	$\sqrt{(k-1)F_{(\alpha, k-1, n-k)}}$
Tukey'HSD	$\frac{q_{(\alpha, k, n-k)}}{\sqrt{2}} , q_{(k, n-k)} = \text{표분화법위분포}$

One-Way ANOVA

❖ Fisher's LSD(least significant difference)

- s_p^2 (공통분산)를 사용하지 않고, 분석후에 사용된 MSE를 이용하여 분석

$$T_{cal} = \frac{\bar{y}_i - \bar{y}_j}{\sqrt{MSE \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}} > t_{(\frac{\alpha}{2}, n-k)}$$

$$t_{cal} = \frac{(\bar{x}_2 - \bar{x}_1) - (\mu_2 - \mu_1)}{s_p^2 \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

$$T_{강남-강서} = \frac{|90.61 - 89.59|}{\sqrt{22.573 \left(\frac{1}{38} + \frac{1}{30} \right)}} = \frac{1.02}{1.16} = 0.880 > t_{(0.025, 135)} = 1.978$$

❖ Bonferroni

- LSD의 경우 가설 검정의 수에 따라 유의수준이 증가 → Bonferroni로 보정

$$t_{(\frac{\alpha}{2}, n-k)} \rightarrow t_{(\frac{\alpha}{2k}, n-k)}$$

$$t_{(\frac{\alpha}{2}, n-k)} = 1.978 \rightarrow t_{(\frac{\alpha}{2k}, 135)} = t_{(\frac{0.05}{2*6=12}, 135)} = 2.915$$

One-Way ANOVA

LGE Internal Use Only

- ❖ Tukey's test(Q)

$$T_{cal} = \frac{\bar{y}_i - \bar{y}_j}{\sqrt{MSE \left(\frac{1}{n_1} + \frac{1}{n_2} \right)}} > T_{(k,n-k)}$$

$$\begin{aligned} T_{cal} &= \frac{|90.61 - 89.59|}{\sqrt{22.573 \left(\frac{1}{38} + \frac{1}{30} \right)}} \\ &= \frac{1.02}{1.16} \\ &= 0.880 > T_{(4,135)} = 3.63 \end{aligned}$$

$$* t_{(0.025,135)} = 1.978$$

Critical Values for the Tukey Q Test

d.f. = N – K (ANOVA Error or Within d.f.) for ANOVA: Single Factor

d.f. = N – R*C (ANOVA Within d.f.) for ANOVA: Two-Factor With Replication

d.f. = (R – 1)*(C – 1) (ANOVA Error d.f.) for ANOVA: Two-Factor Without Replication

Number of Groups (Treatments) = K

Error df	Number of Groups (Treatments)									
	2	3	4	5	6	7	8	9	10	
1	17.97	26.98	32.82	37.08	40.41	43.12	45.40	47.36	49.07	
2	6.08	8.33	9.80	10.88	11.74	12.44	13.03	13.54	13.99	
3	4.50	5.91	6.82	7.50	8.04	8.48	8.85	9.18	9.46	
4	3.93	5.04	5.76	6.29	6.71	7.05	7.35	7.60	7.83	
5	3.64	4.60	5.22	5.67	6.03	6.33	6.58	6.80	6.99	
6	3.46	4.34	4.90	5.30	5.63	5.90	6.12	6.32	6.49	
7	3.34	4.16	4.68	5.06	5.36	5.61	5.82	6.00	6.16	
8	3.26	4.04	4.53	4.89	5.17	5.40	5.60	5.77	5.92	
9	3.20	3.95	4.41	4.76	5.02	5.24	5.43	5.59	5.74	
10	3.15	3.88	4.33	4.65	4.91	5.12	5.30	5.46	5.60	
11	3.11	3.82	4.26	4.57	4.82	5.03	5.20	5.35	5.49	
12	3.08	3.77	4.20	4.51	4.75	4.95	5.12	5.27	5.39	
13	3.06	3.73	4.15	4.45	4.69	4.88	5.05	5.19	5.32	
14	3.03	3.70	4.11	4.41	4.64	4.83	4.99	5.13	5.25	
15	3.01	3.67	4.08	4.37	4.59	4.78	4.94	5.08	5.20	
16	3.00	3.65	4.05	4.33	4.56	4.74	4.90	5.03	5.15	
17	2.98	3.63	4.02	4.30	4.52	4.70	4.86	4.99	5.11	
18	2.97	3.61	4.00	4.28	4.49	4.67	4.82	4.96	5.07	
19	2.96	3.59	3.98	4.25	4.47	4.65	4.79	4.92	5.04	
20	2.95	3.58	3.96	4.23	4.45	4.62	4.77	4.90	5.01	
24	2.92	3.53	3.90	4.17	4.37	4.54	4.68	4.81	4.92	
30	2.89	3.49	3.85	4.10	4.30	4.46	4.60	4.72	4.82	
40	2.86	3.44	3.79	4.04	4.23	4.39	4.52	4.63	4.73	
60	2.83	3.40	3.74	3.98	4.16	4.31	4.44	4.55	4.65	
120	2.80	3.36	3.68	3.92	4.10	4.24	4.36	4.47	4.56	
∞	2.77	3.31	3.63	3.86	4.03	4.17	4.29	4.39	4.47	

❖ ANOVA 모형의 가정(잔차검정)

- ANOVA는 분산을 이용해 검정하는 기법으로 잔차의 가정이 중요함
- 잔차의 등분산성: 잔차그림, 등분산검정(Levene test)
- **잔차의 정규성:** Shapiro-Wilk, jarque-Bera
- **잔차의 독립성:** 무작위로 표본 추출(ANOVA에서는 특별히 강조하지 않음)
- **변동의 분해**

$$y_{ij} - \bar{y} = (\bar{y}_j - \bar{y}) + (y_{ij} - \bar{y}_j)$$

총편차 = 처리효과 + 오차

- ANOVA 모형식

$$Y_{ij} - \mu = (\mu_i - \mu) + \varepsilon_{ij}$$

$$Y_{ij} = \mu + (\mu_i - \mu) + \varepsilon_{ij}$$

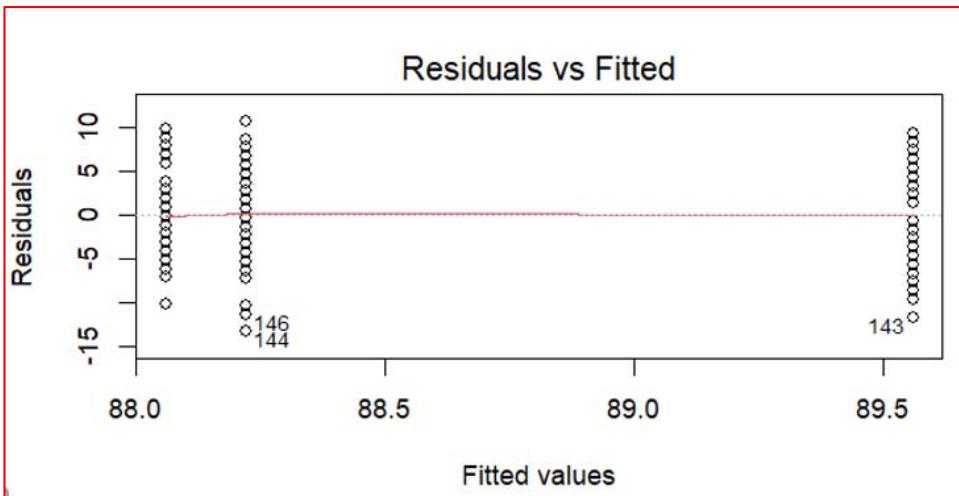
전체평균 + 처리효과 + 오차

- 오차 → 잔차의 가정

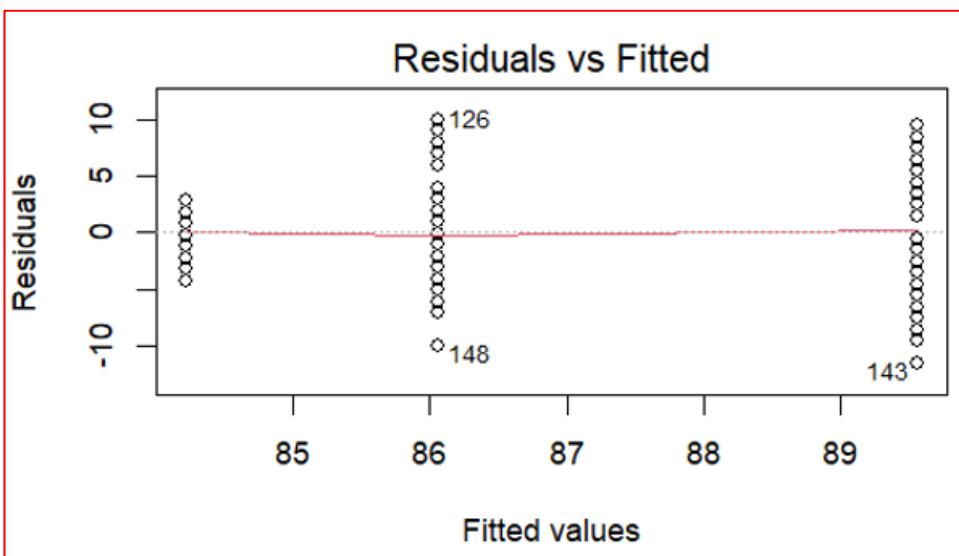
$$\varepsilon_{ij} \rightarrow e_{ij} = (y_{ij} - \bar{y}_j) \sim N(0, \sigma^2)$$

One-Way ANOVA

LGE Internal Use Only



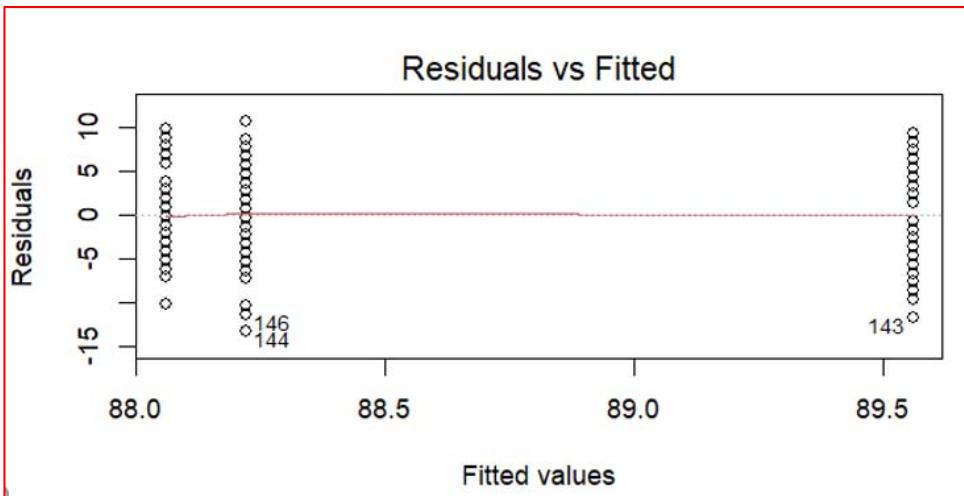
잔차의 등분산성(O)
잔차의 정규성(O)



잔차의 등분산성(X)
잔차의 정규성(O)

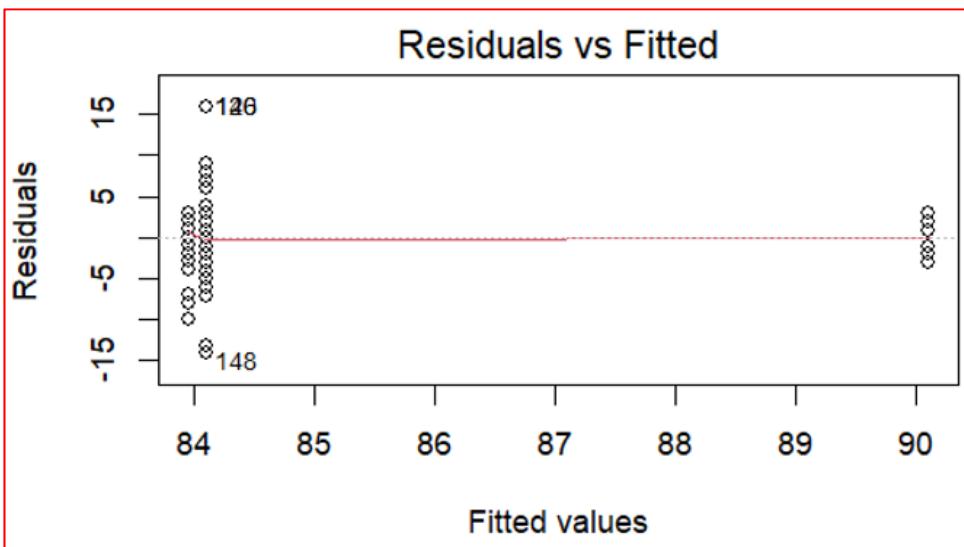
One-Way ANOVA

LGE Internal Use Only



잔차의 등분산성(O)

잔차의 정규성(O)



잔차의 등분산성(X)

잔차의 정규성(X)

One-Way ANOVA 분석절차

LGE Internal Use Only



07_1.Own-Way ANOVA

07_1.Own-Way ANOVA

- <https://pingouin-stats.org/build/html/generated/pingouin.anova.html#pingouin.anova>

1.기본 package 설정

```
[ ] # 그래프에서 한글 폰트 인식하기  
!sudo apt-get install -y fonts-nanum  
!sudo fc-cache -fv  
!rm ~/.cache/matplotlib -rf  
  
[ ] !pip install pingouin  
  
# *** 런타임 다시 시작
```

2초 [1] # 1.기본
import numpy as np # numpy 패키지 가져오기
import matplotlib.pyplot as plt # 시각화 패키지 가져오기
import seaborn as sns # 시각화

2.데이터 가져오기
import pandas as pd # csv -> dataframe으로 전환

3.통계분석 package
import pingouin as pg
from scipy import stats
import statsmodels.api as sm

0초 [2] # 기본세팅
테마 설정
sns.set_theme(style = "darkgrid")

0초 오후 8:40에 완료됨

2.데이터 불러오기

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following structure:

- Section 2.1:** 2.1 데이터 프레임으로 저장
 - 원본데이터(csv)를 dataframe 형태로 가져오기(pandas)
- Code Cell 3:** # 원본데이터(csv)를 dataframe 형태로 가져오기(pandas)
owa_df = pd.read_csv('https://raw.githubusercontent.com/leecho-bigdata/statistics-python/main/07_1.0WA.csv', encoding="cp949")
owa_df.head()
- Data Preview:** A table showing the first 5 rows of the DataFrame.

	매장	만족도1	만족도2	만족도3	만족도4	만족도5
0	2	85	85	85	84	84
1	2	82	82	82	81	81
2	3	90	86	88	88	86
3	2	88	88	88	87	87
4	1	93	93	93	93	93
- Section 2.2:** 2.2 범주형 변수 처리
 - 가변수 처리시 문자로 처리를 해야 변수명 구분이 쉬움
- Code Cell 4:** owa_df['매장'].replace({1:'강남', 2:'강동', 3:'강서'}, inplace=True)
owa_df['매장'] = owa_df['매장'].astype('category')

owa_df.head()
- Data Preview:** A table showing the first 5 rows of the DataFrame after categorical conversion.

	매장	만족도1	만족도2	만족도3	만족도4	만족도5
0	강동	85	85	85	84	84

2.데이터 불러오기

LGE Internal Use Only

Next steps: View recommended plots

▽ 2.2 범주형 변수 처리

- 가변수 처리시 문자로 처리를 해야 변수명 구분이 쉬움

✓ [4] owa_df[['매장']].replace({1:'강남', 2:'강동', 3:'강서'}, inplace=True)
owa_df['매장'] = owa_df['매장'].astype('category')

owa_df.head()

	매장	만족도1	만족도2	만족도3	만족도4	만족도5
0	강동	85	85	85	84	84
1	강동	82	82	82	81	81
2	강서	90	86	88	88	86
3	강동	88	88	88	87	87
4	강남	93	93	93	93	93

Next steps: View recommended plots

▽ 2.3 자료구조 살펴보기

✓ [5] owa_df.shape
(150, 6)

✓ [6] owa_df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column   Non-Null Count  Dtype  
--- 
 0   매장      150 non-null   category
  ✓  0초 오후 8:40에 완료됨
```

3. 기술통계

LGE Internal Use Only

```
✓ 0초 [8] # 그룹별 기술통계
owa_df.groupby('매장')[["만족도1"]].describe().round(3)

    count   mean   std   min   25%   50%   75%   max
매장
강남    50.0  89.56  5.610  78.0  86.0  89.0  94.00  99.0
강동    50.0  88.22  5.396  75.0  85.0  88.5  92.00  99.0
강서    50.0  88.06  4.770  78.0  85.0  88.0  90.75  98.0

✓ 0초 [9] # 분석변수가 여러개 일 때
num_feature = ['만족도1', '만족도2', '만족도3', '만족도4', '만족도5']
for num in num_feature:
    print("----", num, "----")
    results = owa_df.groupby('매장')[num].describe().round(3)
    print(results, "#n")

---- 만족도1 ----
    count   mean   std   min   25%   50%   75%   max
매장
강남    50.0  89.56  5.610  78.0  86.0  89.0  94.00  99.0
강동    50.0  88.22  5.396  75.0  85.0  88.5  92.00  99.0
강서    50.0  88.06  4.770  78.0  85.0  88.0  90.75  98.0

---- 만족도2 ----
    count   mean   std   min   25%   50%   75%   max
매장
강남    50.0  89.56  5.610  78.0  86.0  89.0  94.00  99.0
강동    50.0  88.22  5.396  75.0  85.0  88.5  92.00  99.0
강서    50.0  84.06  4.770  74.0  81.0  84.0  86.75  94.0

---- 만족도3 ----
    count   mean   std   min   25%   50%   75%   max
매장
강남    50.0  89.56  5.610  78.0  86.0  89.0  94.00  99.0
강동    50.0  87.90  4.958  75.0  85.0  88.5  92.00  95.0
강서    50.0  86.06  4.770  76.0  83.0  86.0  88.75  96.0
```

4.ANOVA

```
▼ 4.ANOVA
  ▼ 4.1 ANOVA(차이 없는 경우)

✓ 0초 [10] # 등분산일때
    pg.anova(dy = '만족도1',
              between = '매장',
              data = owa_df,
              detailed = True).round(3)

    Source      SS   DF     MS      F  p-unc  np2
    0   매장    67.853   2  33.927  1.221  0.298  0.016
    1   Within  4083.720 147 27.780  NaN    NaN    NaN

✓ 0초 [11] # 그래프
    sns.catplot(x = "매장",
                  y = "만족도1",
                  kind = "point",
                  data = owa_df)
    plt.show()


```

4.ANOVA

▼ 4.2 ANOVA(차이 있을 경우)

- 만족도2: 강남=강동=강서 차이가 있음 (a,a,b)
- 만족도3: 강남-강서만 차이 있음(a, ab, b)

▼ 4.2.1 ANOVA

✓ [12] # 등분산일때

```
pg.anova(dv = '만족도2',
          between = '매장',
          data = owa_df,
          detailed = True).round(3)
```

Source	SS	DF	MS	F	p-unc	np2
0 매장	822.52	2	411.26	14.804	0.0	0.168
1 Within	4083.72	147	27.78	NaN	NaN	NaN

✓ [13] # 등분산일때

```
pg.anova(dv = '만족도3',
          between = '매장',
          data = owa_df,
          detailed = True).round(3)
```

Source	SS	DF	MS	F	p-unc	np2
0 매장	306.52	2	153.26	5.834	0.004	0.074
1 Within	3861.64	147	26.27	NaN	NaN	NaN

▼ 4.2.2 사후분석(post-hoc, Multicom)

✓ [14] # post-hoc test: LSD

```
pg.pairwise_tests(data = owa_df,
```

✓ 0초 오전 10:15에 완료됨

4.ANOVA

▼ 4.2.2 사후분석(post-hoc, Multicom)

```
[14] # post-hoc test: LSD
pg.pairwise_tests(data = owa_df,
                   dv = '만족도2',
                   between = '매장').round(3)
```

Contrast	A	B	Paired	Parametric	T	dof	alternative	p-unc	BF10	hedges	
0	매장	강동	강서	False	True	4.085	98.0	two-sided	0.000	241.199	0.811
1	매장	강동	강남	False	True	-1.217	98.0	two-sided	0.226	0.407	-0.242
2	매장	강서	강남	False	True	-5.281	98.0	two-sided	0.000	1.77e+04	-1.048

```
[15] # post-hoc test: Tukey
pg.pairwise_tukey(data = owa_df,
                   dv = '만족도2',
                   between = '매장').round(3)
```

A	B	mean(A)	mean(B)	diff	se	T	p-tukey	hedges	
0	강동	강서	88.22	84.06	4.16	1.054	3.946	0.000	0.811
1	강동	강남	88.22	89.56	-1.34	1.054	-1.271	0.414	-0.242
2	강서	강남	84.06	89.56	-5.50	1.054	-5.218	0.000	-1.048

```
[16] # post-hoc test: Tukey
pg.pairwise_tukey(data = owa_df,
                   dv = '만족도3',
                   between = '매장').round(3)
```

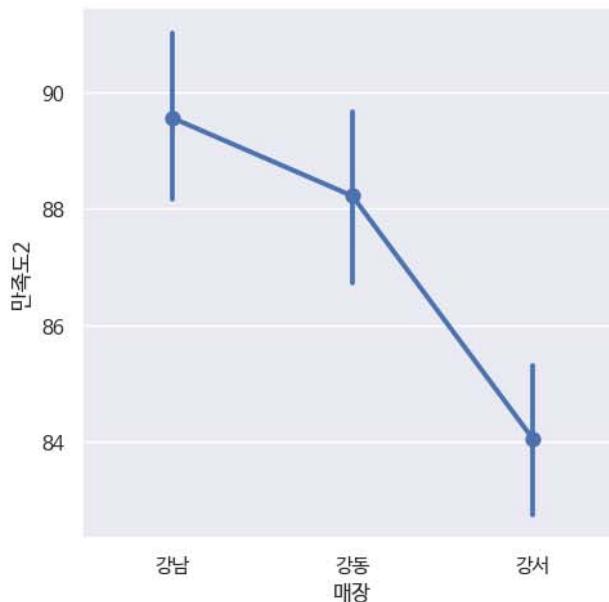
A	B	mean(A)	mean(B)	diff	se	T	p-tukey	hedges	
0	강동	강서	87.90	86.06	1.84	1.025	1.795	0.175	0.375
1	강동	강남	87.90	89.56	-1.66	1.025	-1.619	0.241	-0.311
2	강서	강남	86.06	89.56	-3.50	1.025	-3.414	0.002	-0.667

✓ 0초 오전 10:15에 완료됨

4.ANOVA

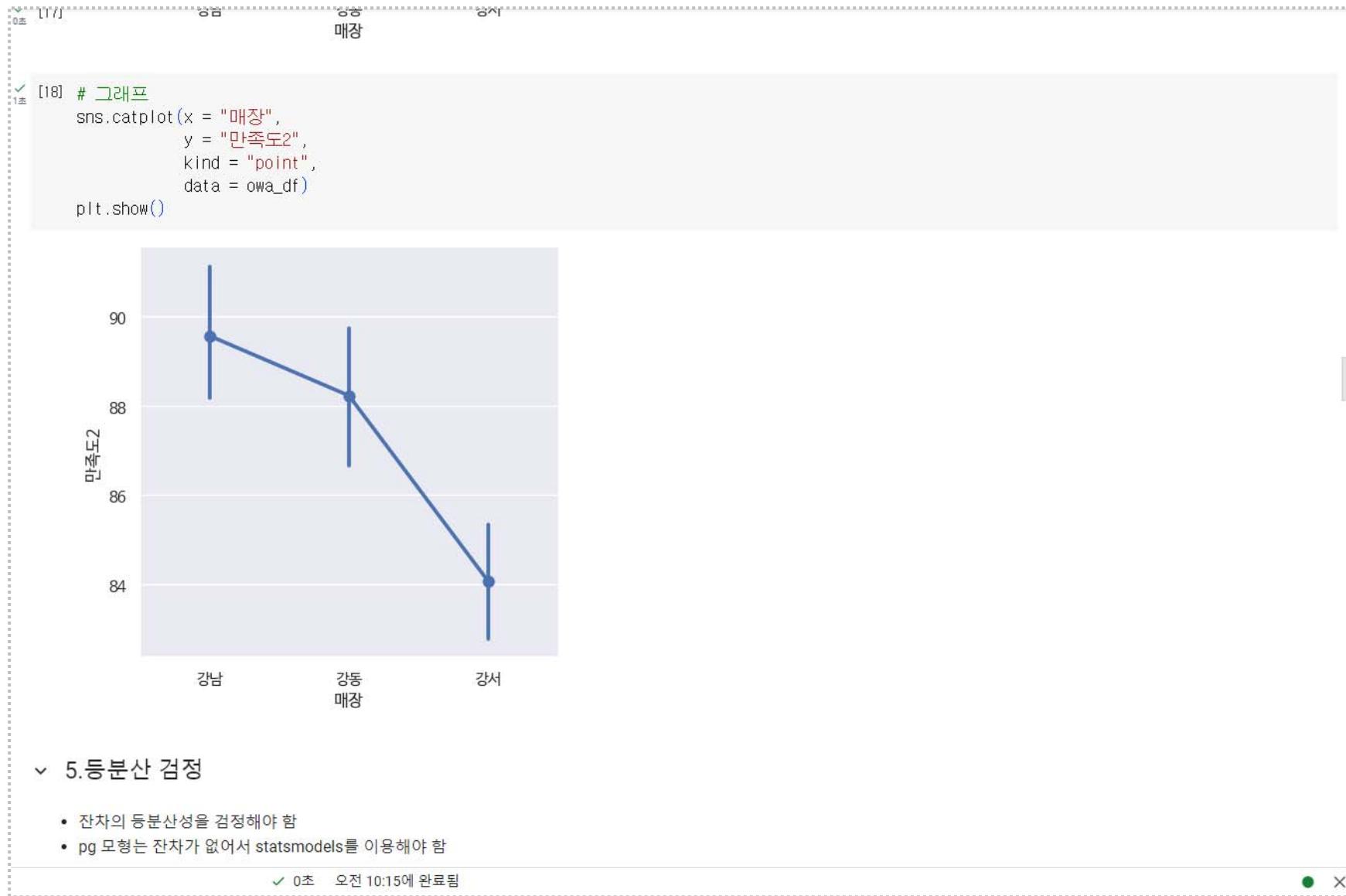
```
✓ [16] 1 강동 강남 87.90 89.56 -1.66 1.025 -1.619 0.241 -0.311  
      2 강서 강남 86.06 89.56 -3.50 1.025 -3.414 0.002 -0.667
```

```
✓ [17] # 그래프  
sns.catplot(x = "매장",  
            y = "만족도2",  
            kind = "point",  
            data = owa_df)  
plt.show()
```



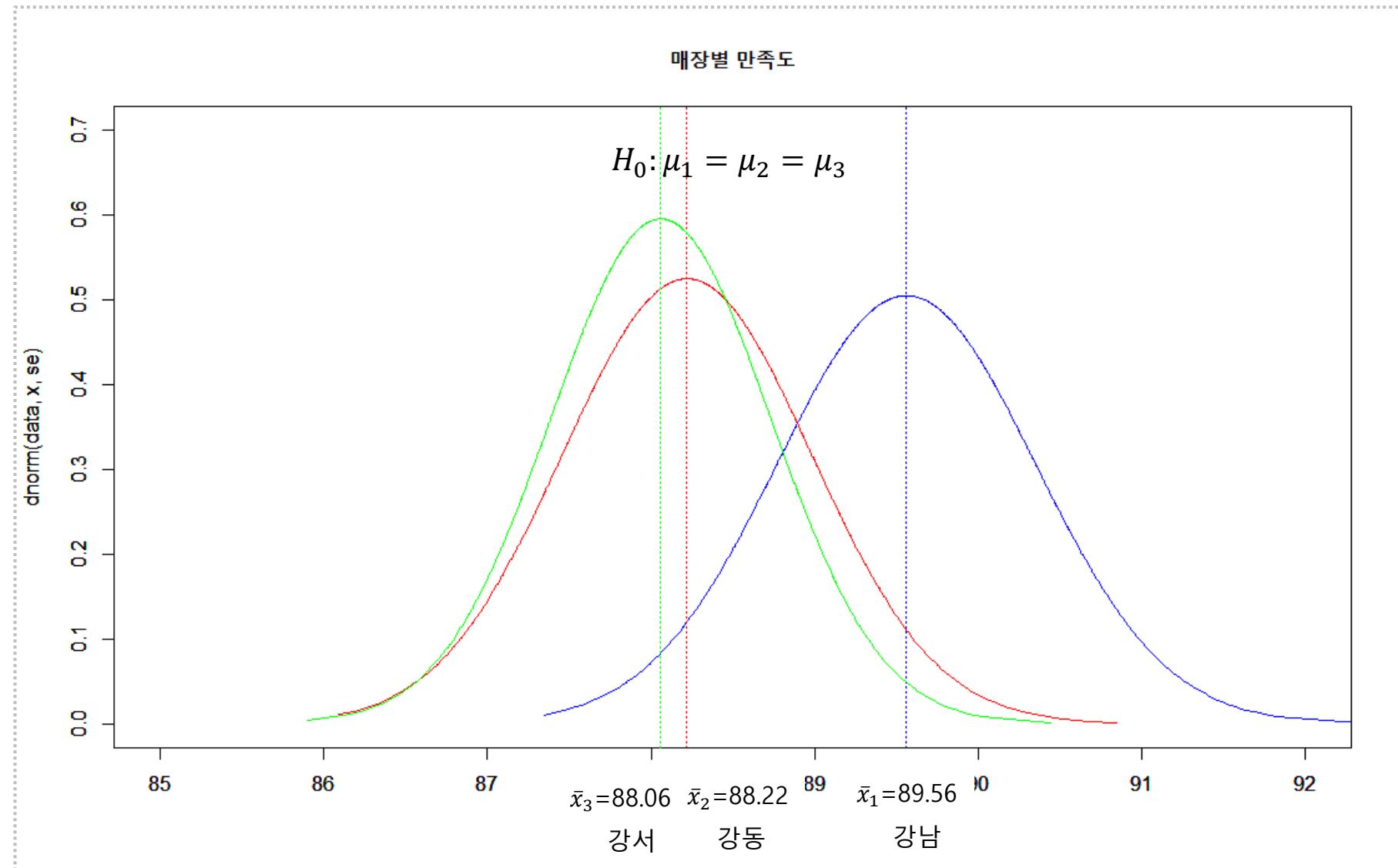
```
✓ [18] # 그래프  
sns.catplot(x = "매장",  
            y = "만족도2",  
            kind = "point",  
            data = owa_df)
```

4.ANOVA



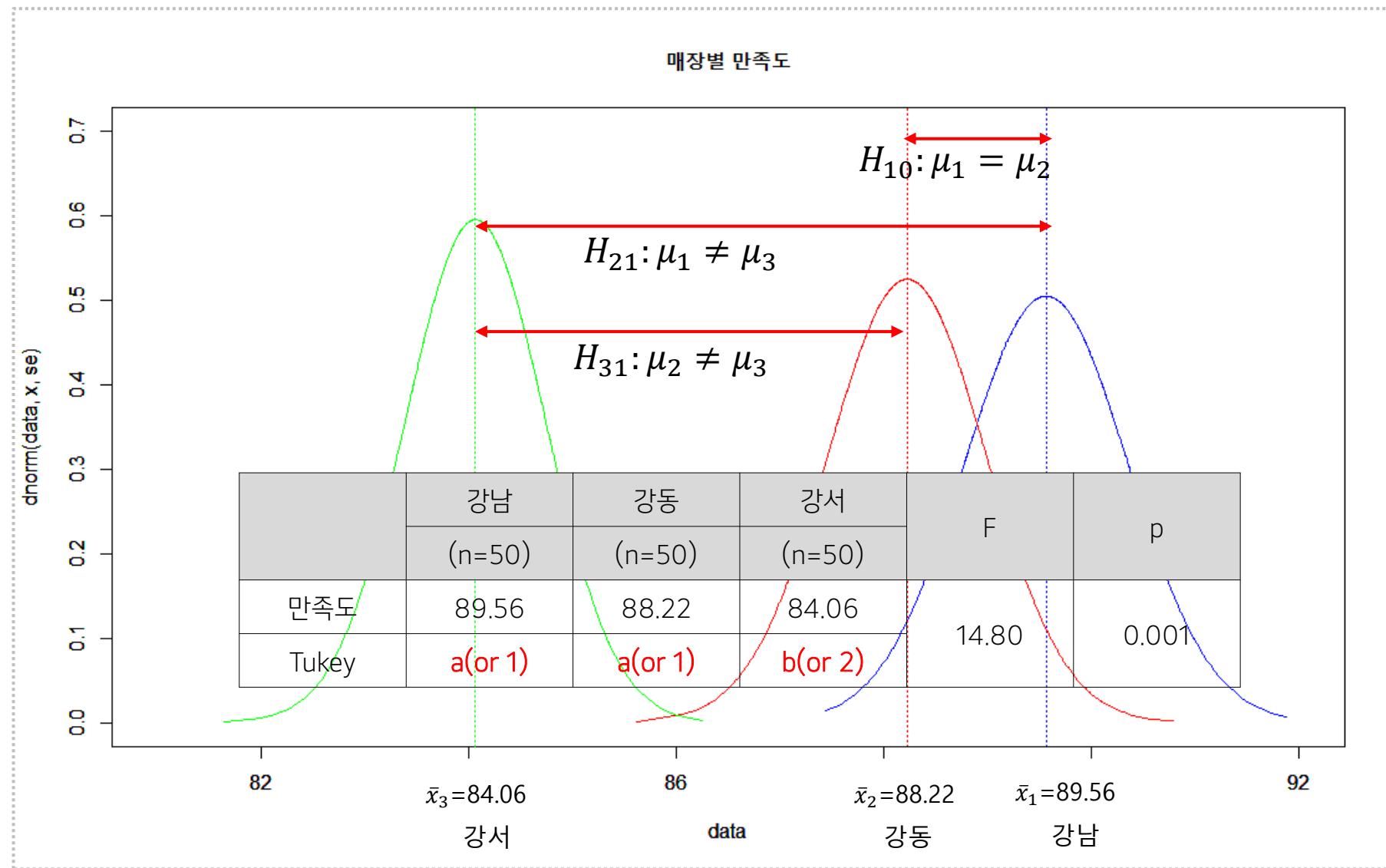
One-Way ANOVA

LGE Internal Use Only

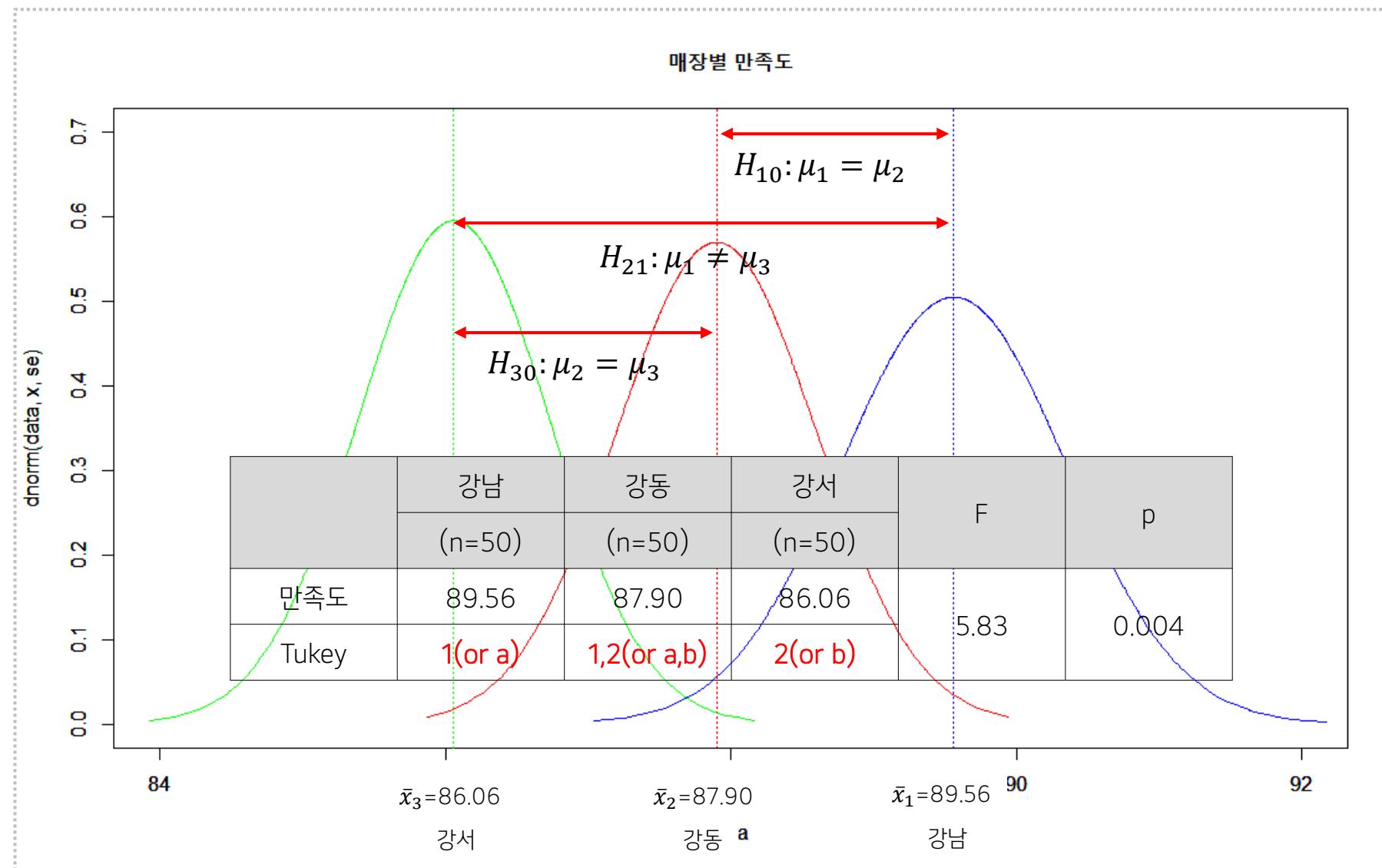


One-Way ANOVA

LGE Internal Use Only



One-Way ANOVA



One-Way ANOVA

LGE Internal Use Only

❖ 다중비교 정리

- 만족도2일 경우에 정리

	강남	강동	강서	F	p
	(n=50)	(n=50)	(n=50)		
만족도	89.56	88.22	84.06		
Tukey	a	a	b	14.80	0.001

- 만족도3일 경우에 정리

	강남	강동	강서	F	p
	(n=50)	(n=50)	(n=50)		
만족도	89.56	87.90	86.06		
Tukey	1	1, 2	2	5.83	0.004

5. 등분산 검정

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following content:

- Section 5. 등분산 검정**
 - 잔차의 등분산성을 검정해야 함
 - pg 모형은 잔차가 없어서 statsmodels를 이용해야 함
- Section 5.1 등분산 검정**
- Code and Output:**
 - [19] # 등분산이면 지금까지 분석한 것이 문제 없음
pg.homoscedasticity(owa_df, dv="만족도1", group="매장")

	pval	equal_var
levene	1.531605	0.219618
 - [20] num_feature = ['만족도1', '만족도2', '만족도3', '만족도4', '만족도5']
for num in num_feature:
 print("----", num, "----")
 results = pg.homoscedasticity(owa_df, dv=num, group="매장")
 print(results, "\n")

---- 만족도1 ----

	pval	equal_var
levene	1.531605	0.219618

---- 만족도2 ----

	pval	equal_var
levene	1.531605	0.219618

---- 만족도3 ----

	pval	equal_var
levene	1.608568	0.203677

---- 만족도4 ----

	pval	equal_var
levene	18.485413	6.898834e-08

---- 만족도5 ----

	pval	equal_var
levene	11.235009	0.000029

5. 등분산 검정

LGE Internal Use Only

```
▼ 5.2 이분산일때(Welch's test)
  ▼ 5.2.1 ANOVA(Welch's test)

  [21] pg.homoscedasticity(owa_df, dv="만족도4", group="매장")
    ┌─────────┐ pval equal_var ┌─────────┐
    ┌─────────┐ levene 18.485413 6.898834e-08 ┌─────────┐
    ┌─────────┐
    [22] # Welch's test
    pg.welch_anova(dv = '만족도4',
                    between = '매장',
                    data = owa_df).round(3)
    ┌─────────┐ Source ddof1 ddof2   F p-unc   np2 ┌─────────┐
    ┌─────────┐ 0     매장      2 83.173 20.622  0.0 0.203 ┌─────────┐
    ┌─────────┐
    [23] # 등분산일때와 비교
    pg.anova(dv = '만족도4',
              between = '매장',
              data = owa_df,
              detailed = True).round(3)
    ┌─────────┐ Source   SS   DF   MS   F p-unc   np2 ┌─────────┐
    ┌─────────┐ 0     매장 735.853  2 367.927 18.678  0.0 0.203 ┌─────────┐
    ┌─────────┐ 1     Within 2895.720 147 19.699  NaN  NaN  NaN ┌─────────┐
    ┌─────────┐
  ▼ 5.2.2 이분산 사후분석(post-hoc, Multicom)

  [24] # 이분산 post-hoc test
  scrollTo... ┌─────────┐
  ┌─────────┐ 0초 오전 10:15에 완료됨 ┌─────────┐
  ┌─────────┐
  ┌─────────┐
```

5. 등분산 검정

LGE Internal Use Only

▼ 5.2.2 이분산 사후분석(post-hoc, Multicom)

[24] # 이분산 post-hoc test
pg.pairwise_gameshowell(data = owa_df,
 dv = '만족도4',
 between = '매장').round(3)

	A	B	mean(A)	mean(B)	diff	se	T	df	pval	hedges
0	강동	강서	84.22	86.06	-1.84	0.743	-2.476	69.054	0.041	-0.491
1	강동	강남	84.22	89.56	-5.34	0.853	-6.263	63.805	0.000	-1.243
2	강서	강남	86.06	89.56	-3.50	1.041	-3.361	95.527	0.003	-0.667

[25] # 그래프
sns.catplot(x = "매장",
 y = "만족도4",
 kind = "point",
 data = owa_df)
plt.show()



scrollTo... ✓ 0초 오전 10:15에 완료됨

6. 정규성 검정

LGE Internal Use Only

▼ 6. 정규성 검정

- 잔차의 정규성을 검정해야 함
- pg 모형은 잔차가 없어서 statsmodels를 이용해야 함

▼ 6.1 정규분포 검정

✓ [26] # 매장별 정규성을 검정하면 강동은 정규성이 아닌 것으로 나타남
잔차의 정규성을 검정해야 함
pg.normality(owa_df, dv='만족도4', group='매장')

	pval	normal
매장		
강동	0.891831	0.000261
강서	0.974630	0.353244
강남	0.969113	0.212952

✓ [27] # 잔차의 정규성을 검정한 결과 잔차는 정규분포로 나타남
import statsmodels.api as sm
from statsmodels.formula.api import ols

model = ols('만족도4 ~ C(매장)', data = owa_df)
result = model.fit()
pg.normality(result.resid).round(3)

	pval	normal
0	0.989	0.268
		True

▼ 6.2 비모수일때(Kruskal Wallis)

✓ 0초 오전 10:15에 완료됨● ×

6.정규성 검정

LGE Internal Use Only

```
▼ 6.2 비모수일때(Kruskal Wallis)
  ▼ 6.2.1 Kruskal Wallis
    [28] pg.normality(owa_df, dv='만족도5', group='매장')

      # pval normal
      매장
      강동 0.838372 0.000008 False
      강서 0.957795 0.071865 True
      강남 0.889787 0.000225 False

    [29] # 잔차의 정규성
        import statsmodels.api as sm
        from statsmodels.formula.api import ols

        model = ols('만족도5 ~ C(매장)', data = owa_df).fit()
        pg.normality(model.resid).round(3)

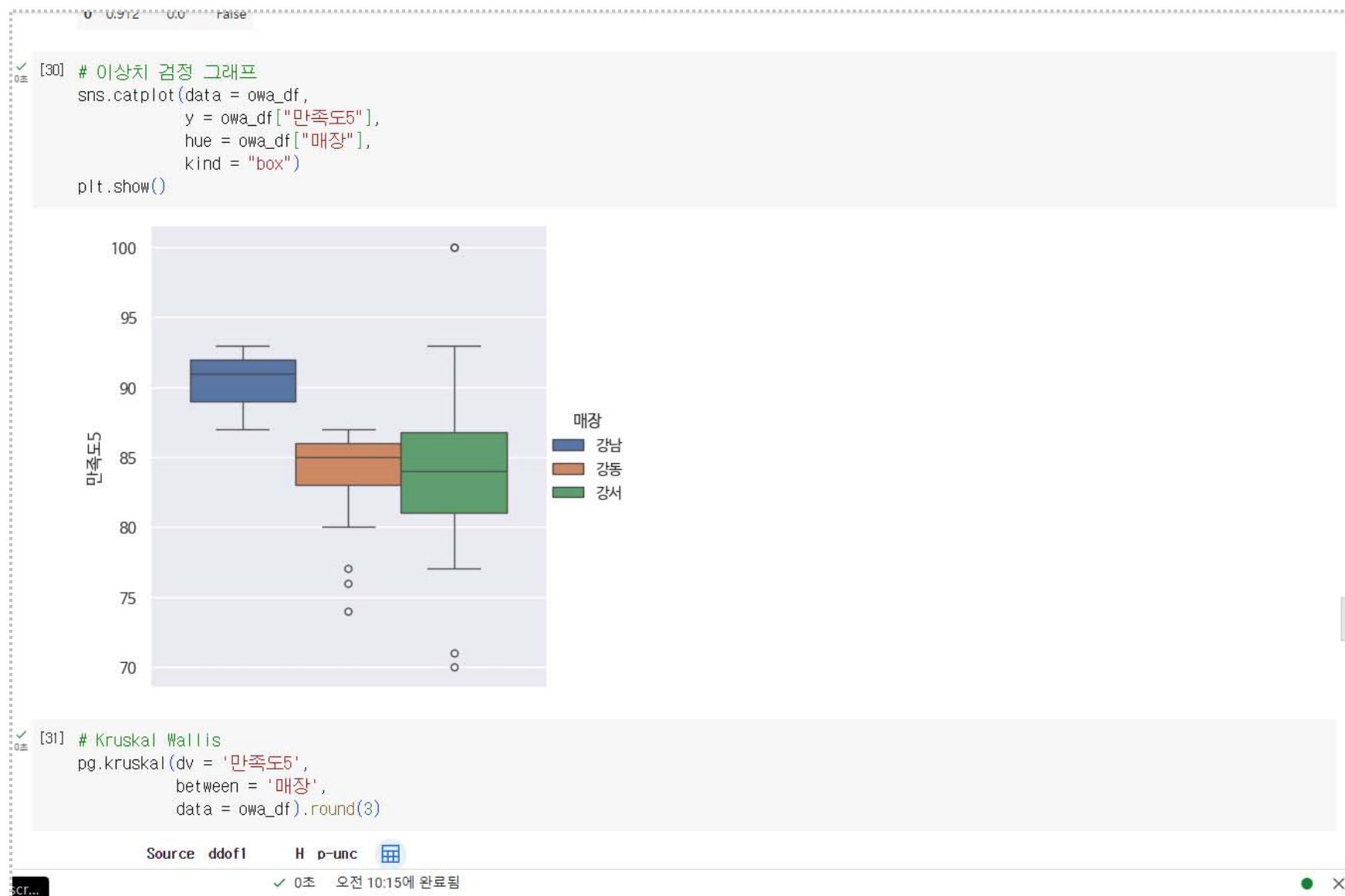
      # pval normal
      0 0.912 0.0 False

    [30] # 이상치 검정 그래프
        sns.catplot(data = owa_df,
                     y = owa_df["만족도5"],
                     hue = owa_df["매장"],
                     kind = "box")
        plt.show()

  100
  ✓ 0초 오전 10:15에 완료됨
```

6. 정규성 검정

LGE Internal Use Only



6. 정규성 검정

LGE Internal Use Only

```
✓ [30] 
```

```
✓ [31] # Kruskal Wallis
pg.kruskal(dv = '만족도5',
            between = '매장',
            data = owa_df).round(3)
```

Source	ddof1	H	p-unc	
Kruskal	매장	2	73.725	0.0

```
✓ [32] # 모수통계(t-test)와 비교
pg.anova(dv = '만족도5',
          between = '매장',
          data = owa_df,
          detailed = True).round(3)
```

Source	SS	DF	MS	F	p-unc	np2	
0	매장	1228.653	2	614.327	39.801	0.0	0.351
1	Within	2268.920	147	15.435	NaN	NaN	NaN

▼ 6.2.2 비모수 사후분석(post-hoc, Multicom)

```
✓ [33] # 비모수 post-hoc test: LSD
pg.pairwise_tests(data = owa_df,
                   dv = '만족도5',
                   between = '매장',
                   parametric = False).round(3)
```

Contrast	A	B	Paired	Parametric	U-val	alternative	p-unc	hedges	
0	매장	강동	강서	False	False	1300.0	two-sided	0.731	-0.030

✓ 0초 오전 10:15에 완료됨

6.정규성 검정

LGE Internal Use Only

6.2.2 비모수 사후분석(post-hoc, Multicom)

```
[33] # 비모수 post-hoc test: LSD
pg.pairwise_tests(data = owa_df,
                   dv = '만족도5',
                   between = '매장',
                   parametric = False).round(3)
```

Contrast	A	B	Paired	Parametric	U-val	alternative	p-unc	hedges	
0	매장	강동	강서	False	False	1300.0	two-sided	0.731	-0.030
1	매장	강동	강남	False	False	31.5	two-sided	0.000	-2.410
2	매장	강서	강남	False	False	325.0	two-sided	0.000	-1.366

```
[34] # 그래프
sns.catplot(x = "매장",
             y = "만족도5",
             kind = "point",
             data = owa_df)
plt.show()
```

91
90
89
88
87
86
85

매장

✓ 0초 오전 10:15에 완료됨

6.정규성 검정

LGE Internal Use Only



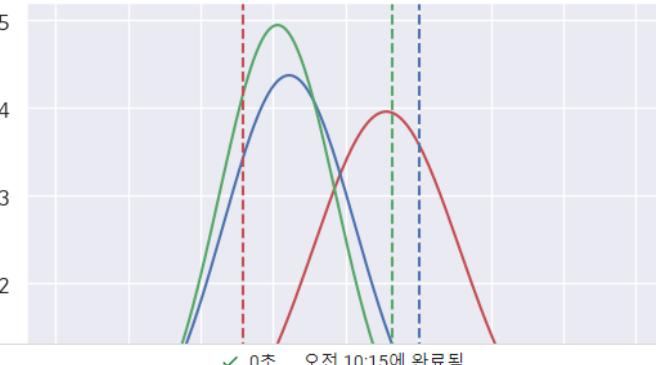
▼ 7.검증결과 그래프

```
[32] from scipy.stats import norm # 정규분포
[33] 0초 오후 8:40에 완료됨
```

7. 검증결과 그래프

LGE Internal Use Only

```
✓ 1초 [35] from scipy.stats import norm # 정규분포  
x_data = np.linspace(85, 93, 200)  
  
x1 = 89.560  
x2 = 88.220  
x3 = 88.060  
  
se1 = 5.610/np.sqrt(31) # 표준오차(표준편차/sqrt(n))  
se2 = 5.396/np.sqrt(35) # 표준오차(표준편차/sqrt(n))  
se3 = 4.770/np.sqrt(35) # 표준오차(표준편차/sqrt(n))  
  
plt.plot(x_data, norm.pdf(x_data, loc = x1, scale = se1), 'r-')  
plt.plot(x_data, norm.pdf(x_data, loc = x2, scale = se2), 'b-')  
plt.plot(x_data, norm.pdf(x_data, loc = x3, scale = se3), 'g-')  
  
plt.axvline(x = x1 - 1.96 * se1, color='r', linestyle='--')  
plt.axvline(x = x2 + 1.96 * se2, color='b', linestyle='--')  
plt.axvline(x = x3 + 1.96 * se3, color='g', linestyle='--')  
plt.show()
```



8.statsmodels 이용

LGE Internal Use Only

```
✓ [36] import statsmodels.api as sm
       from statsmodels.formula.api import ols

       model = ols('만족도1 ~ 매장', data = owa_df).fit()
       aov_table = sm.stats.anova_lm(model, typ=2)
       aov_table.round(3)



|          | sum_sq   | df    | F     | PR(>F) |
|----------|----------|-------|-------|--------|
| 매장       | 67.853   | 2.0   | 1.221 | 0.298  |
| Residual | 4083.720 | 147.0 | Nan   | Nan    |



✓ [37] from statsmodels.stats.multicomp import pairwise_tukeyhsd

       posthoc = pairwise_tukeyhsd(owa_df["만족도1"],
                                    owa_df["매장"])
       print(posthoc)

Multiple Comparison of Means - Tukey HSD, FWER=0.05
=====
group1 group2 meandiff p-adj    lower   upper  reject
강남    강동     -1.34 0.4137 -3.8359 1.1559  False
강남    강서     -1.5  0.3318 -3.9959 0.9959  False
강동    강서     -0.16 0.9874 -2.6559 2.3359  False

✓ [38] fig = posthoc.plot_simultaneous()

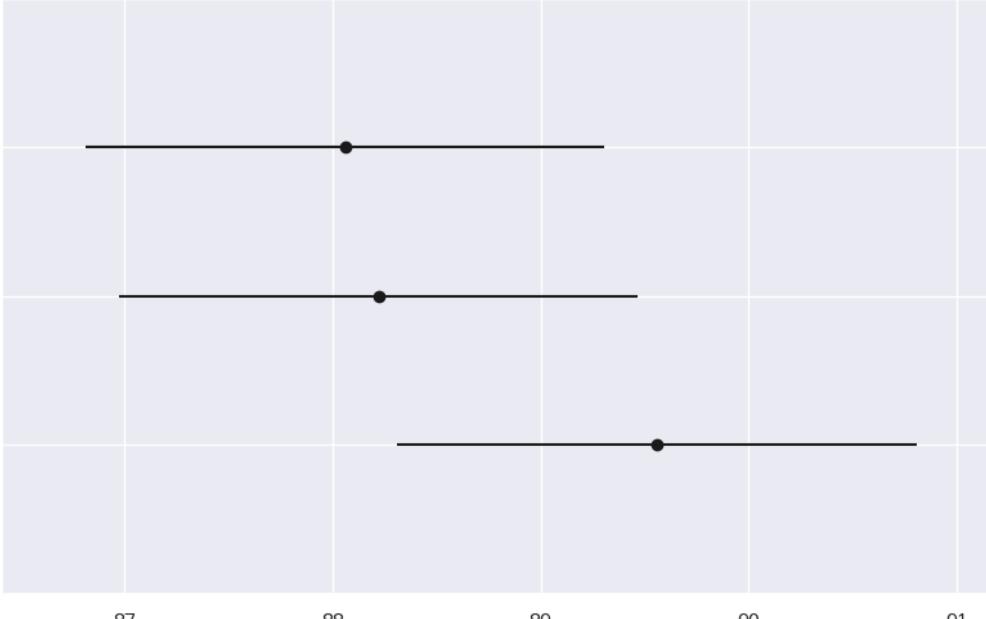

```

8.statsmodels 이용

```
✓ 0초  강남    강동      -7.34 0.4137 -3.8559 1.1559 False
    강남    강서      -1.5  0.3318 -3.9959 0.9959 False
    강동    강서     -0.16 0.9874 -2.6559 2.3359 False

✓ 1초 [38] fig = posthoc.plot_simultaneous()

Multiple Comparisons Between All Pairs (Tukey)

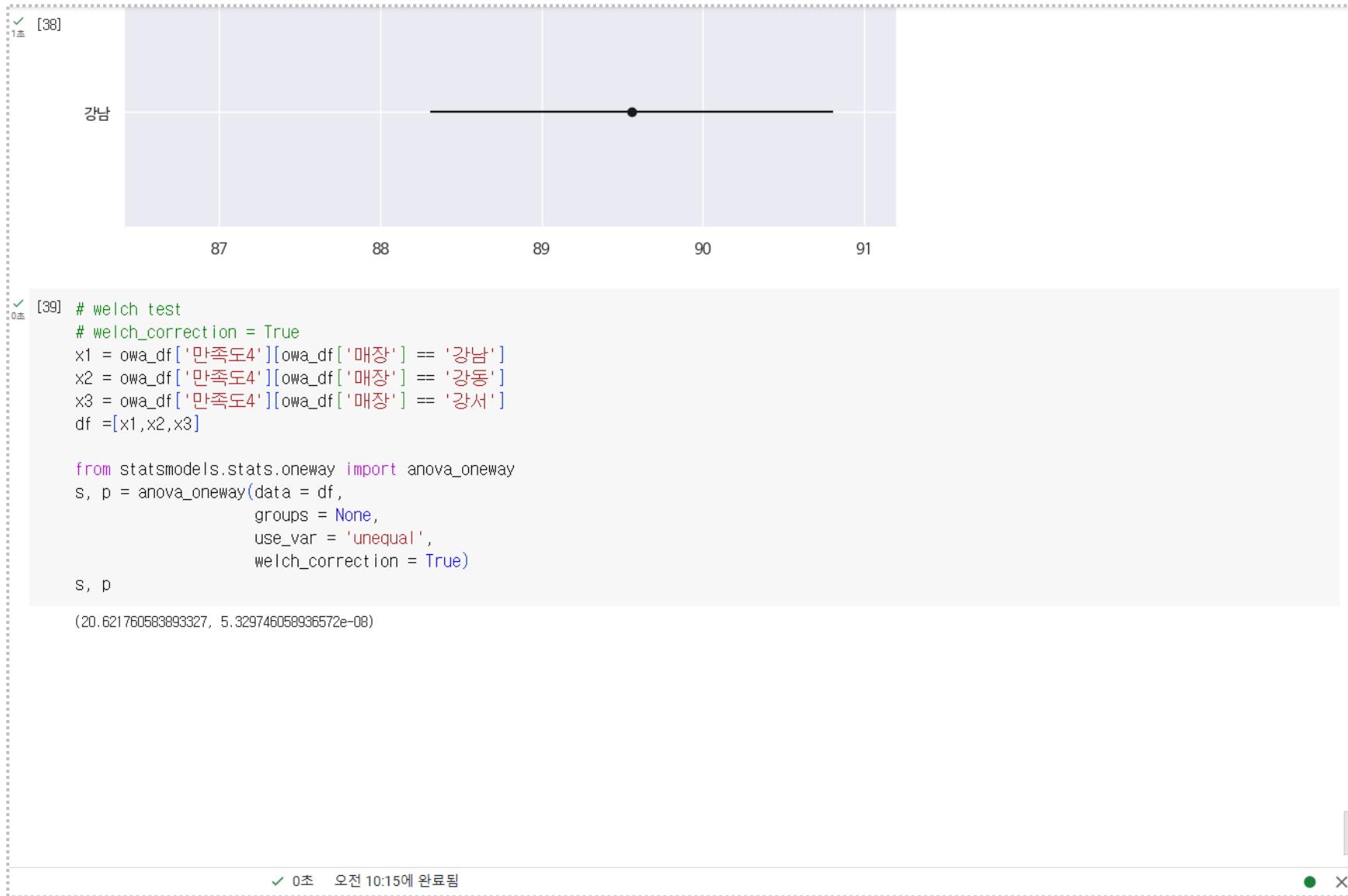


```
✓ 0초 [39] # welch test
welch_correction = True
x1 = owa_df['만족도4'][owa_df['매장'] == '강남']
x2 = owa_df['만족도4'][owa_df['매장'] == '강동']
x3 = owa_df['만족도4'][owa_df['매장'] == '강서']
df =[x1,x2,x3]

✓ 0초 오전 10:15에 완료됨
```


```

8.statsmodels 이용



One-Way ANOVA

- ❖ 매장에 따른 소비자만족도를 분석한 결과, 강남매장의 소비자만족도($M=89.56$), 강동매장의 소비자만족도($M=88.22$), 강서매장의 소비자만족도($M=88.06$) 간에는 통계적으로 유의한 차이가 없는 것으로 나타났다($F=1.22$, $p=0.298$).
- ❖ 만족도1 해석

	강남 (n=50)	강동 (n=50)	강서 (n=50)	F	p
만족도	89.56	88.22	88.06	1.22	0.298
Tukey	1	1	1		

One-Way ANOVA

LGE Internal Use Only

- ❖ 매장에 따른 소비자만족도를 분석한 결과, 강남매장의 소비자만족도($M=89.56$), 강동매장의 소비자만족도($M=87.90$), 강서매장의 소비자만족도($M=86.06$) 간에는 통계적으로 유의한 차이가 있는 것으로 나타났다($F=5.83$, $p=0.004$).
- ❖ 추가로 사후분석을 한 결과, 강남매장의 소비자 만족도가 가장 높았고, 강서매장의 소비자 만족도가 가장 낮았다. 강동매장은 강남과 강서매장과 차이가 없게 나타났다.
- ❖ 만족도2 해석

	강남	강동	강서	F	p
	(n=50)	(n=50)	(n=50)		
만족도	89.56	87.90	86.06	5.83	0.004
Tukey	1	1,2	2		

연습문제

연습문제1

❖ 문제의 정의

- G커피회사는 강남(1), 강동(2), 강서(3), 강북(4)에 매장을 보유하고 있다. 매장별로 고객만족도가 차이가 있는지를 조사하기 위해 매장별 고객만족도를 조사하였다.
- 1:강남, 2:강동, 3:강서, 4:강북
- 1. 만족도1, 만족도2는 매장간에 차이가 있는가?
- 2. 잔차의 등분산 가정이 만족하는가? 만족하지 않다면 Welch's test를 사용하세요.
- 2. 정규분포가정을 만족하는가? 만족하지 않다면 비모수 통계를 사용하세요.
- 07_2.OWA(4).csv

	매장	만족도1	만족도2
1	2	85	85
2	2	82	82
3	3	90	86
4	2	88	88
5	1	93	93
6	2	83	83
7	3	89	85
8	3	88	84
9	1	94	94
10	2	93	93
11	3	97	93
12	3	94	90
13	1	95	95
14	2	84	84
15	3	82	78
16	2	87	87
17	1	86	86
18	2	85	85
19	3	91	87
20	2	92	92
21	1	87	87
22	2	86	86
23	3	86	82
24	1	89	89
25	1	88	88

연습문제2

LGE Internal Use Only

❖ 문제의 정의

- G대학에서는 재학생 만족도 조사를 실시하였다.
- 학과: 1:빅경, 2:사복, 3:간호, 4:음악
- 학년: 1:1학년, 2:2학년, 3:3학년, 4:4학년
- 1. 교양만족도, 전공만족도, 비교과만족도, 전체만족도는 학과에 따라 차이가 있는가?
- 2. 교양만족도, 전공만족도, 비교과만족도, 전체만족도는 학년에 따라 차이가 있는가?
- 3. 잔차의 등분산 가정이 만족하는가? 만족하지 않다면 Welch's test를 사용하세요.
- 4. 정규분포가정을 만족하는가? 만족하지 않다면 비모수통계를 사용하세요.
- 07_3.Education.csv

	학과	학년	교양만족도	전공만족도	
1	1	1	47.6	40.5	
2	1	1	33.3	35.7	
3	1	1	50.0	52.4	
4	1	1	35.7	28.5	
5	1	1	54.7	92.8	
6	1	1	39.3	53.6	
7	1	1	46.4	46.4	
8	1	1	42.9	67.9	
9	2	1	42.9	50.0	
10	2	1	32.1	28.6	
11	2	1	50.0	50.0	
12	2	1	50.0	50.0	
13	2	1	78.6	64.3	
14	2	1	42.9	53.6	
15	3	1	42.8	73.8	
16	3	1	80.5	90.5	
17	3	1	50.0	46.4	
18	3	1	25.0	32.1	
19	3	1	39.3	50.0	
20	3	1	39.3	57.1	
21	3	1	28.6	39.3	
22	3	1	50.0	75.0	
23	3	1	85.7	92.9	
24	3	1	25.0	46.4	
25	3	1	50.0	50.0	

II. Repeated Measures ANOVA

❖ 문제의 정의

- G대학에서 운영하는 6개월 어학 프로그램이 있다.
- 어학 프로그램의 효과를 측정하기 위해 학습 프로그램에 참여한 학생을 대상으로 프로그램 참여전, 중간시험, 최종시험 영어실력을 테스트하였다.
- 관련 어학프로그램은 효과가 있는지? 있다면 언제부터 효과가 나타났을지를 검증해 보자
- 08_1.RMA.csv

❖ 가설

- 귀무가설(H_0): 어학프로그램은 시점에 따라 차이가 없다.

$$H_0: \delta_1 = \delta_2 = \delta_3$$

- 연구가설(H_1): 어학프로그램은 시점에 따라 차이가 있다.

$$H_1: \text{not } H_0$$

Repeated Measures ANOVA

LGE Internal Use Only

❖ 일원 반복측정 분산분석

개체	인자의 수준		
	실시전	3주후	6주후
1	10	16	25
2	11	17	26
3	9	15	21
4	9	14	20
5	9	20	22

평균	11.08	19.15	29.46

Repeated Measures ANOVA

LGE Internal Use Only

❖ 변량분해

$$TSS = SSB$$

총제곱합 = 개체간제곱합

Total = Between subject + Within subject (Treatment + Error)

$$+ SSW (SST + SSE)$$

+ 개체내제곱합 (처리제곱합 + 오차제곱합)

❖ 검정통계량

$$F = \frac{MST}{MSE} = \frac{\frac{SST}{k-1}}{\frac{SSE}{(n-1)(K-1)}} \sim F_{(\alpha, k-1, (n-1)(k-1))}$$

Mauchly's sphericity test

❖ Mauchly's sphericity test

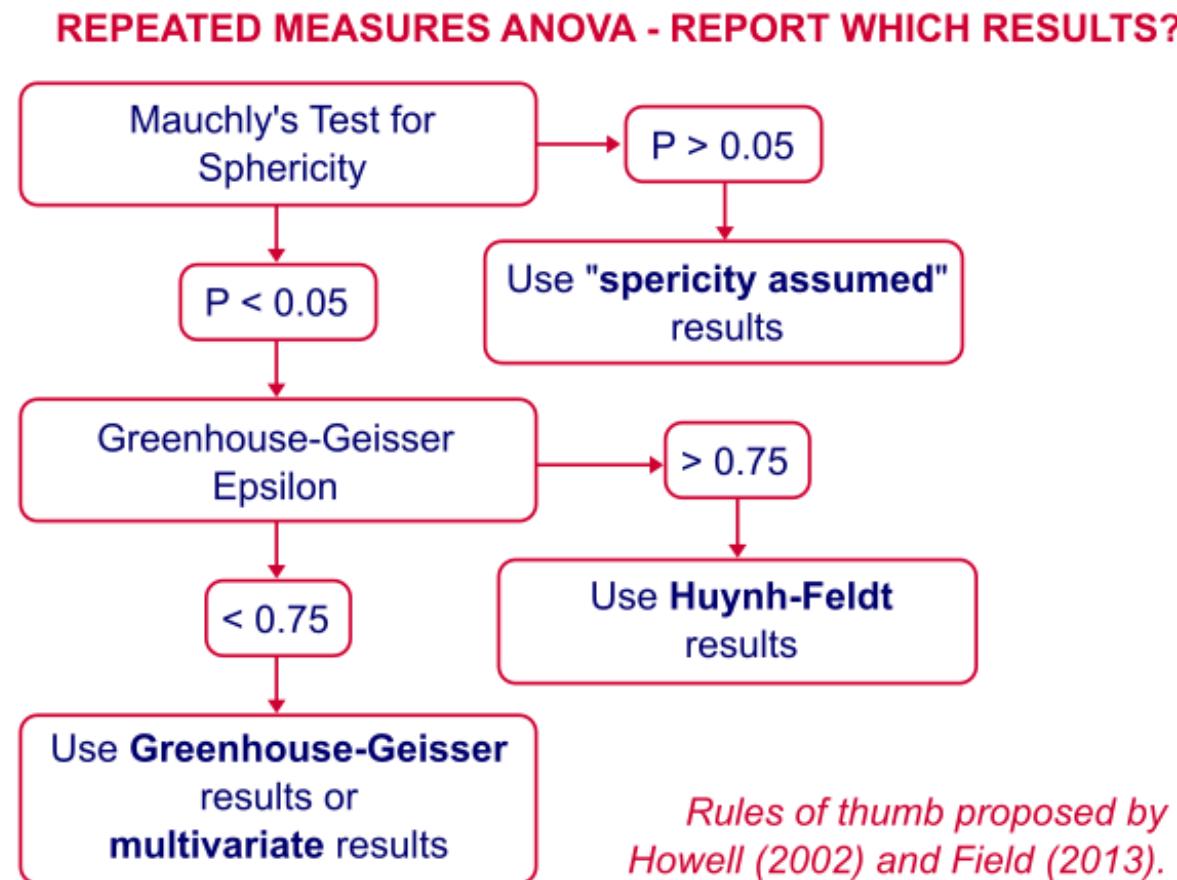
- Mauchly' W = 1
- 1940년 John Mauchly에 의해 개발
- 공분산행렬 Σ 의 복합대칭성
$$\sigma_{jk} = Cov(y_j, y_k) = \begin{cases} \sigma_s^2 + \sigma^2, & \text{if } y_j = y_k \\ \sigma_s^2, & \text{if } y_j \neq y_k \end{cases}$$
- 구형성 검정 가설 $H_0: \sigma_{TXA-TXB}^2 = \sigma_{TXA-TXC}^2 = \sigma_{TxB-TxC}^2$

Patient	Tx A	Tx B	Tx C	Tx A – Tx B	Tx A – Tx C	Tx B – Tx C
1	30	27	20	3	10	7
2	35	30	28	5	7	2
3	25	30	20	-5	5	10
4	15	15	12	0	3	3
5	9	12	7	-3	2	5
Variance:				17	>	10.3 ≈ 10.3

출처: https://en.wikipedia.org/wiki/Mauchly%27s_sphericity_test

Mauchly's sphericity test

- ❖ 구형성 조건에 따른 해석 방법 및 절차



출처: <https://www.spss-tutorials.com/spss-repeated-measures-anova-example-2/comment-page-1/>

Repeated Measures ANOVA

LGE Internal Use Only

❖ 구형성 검정

No	사전	3개월	6개월	사전-3개월	사전-6개월	3개월-6개월
1	63	63	68	0	-5	-5
2	60	60	67	0	-7	-7
3	61	61	70	0	-9	-9
4	57	57	68	0	-11	-11
5	58	58	68	0	-10	-10
6	58	Assumptions				
7	53	Tests of Sphericity				
8	58	<hr/> Mauchly's W p Greenhouse-Geisser ϵ Huynh-Feldt ϵ <hr/>				
9	58	시점 0.908 0.126 0.916 0.954				
43	56	59	67	-3	-11	-8
44	59	57	66	2	-7	-9
45	60	62	67	-2	-7	-5
분산			6.32	≈	5.22	≈ 7.57

Repeated Measures ANOVA

LGE Internal Use Only

❖ 분산분석표

총제곱합 = 개체간제곱합+개체내제곱합 (처리제곱합 + 오차제곱합)

$$TSS = SSB + SSW \quad (SST + SSE)$$

❖ 검정통계량

$$F = \frac{MST}{MSE} = \frac{\frac{12.978}{3 - 1}}{\frac{229.022}{(45 - 1)(3 - 1)}} = 2.493 < 3.848 = F_{(\alpha, k-1, (n-1)(k-1))}$$

$$p-value = P(F > 2.493) = 0.088 > \alpha = 0.05$$

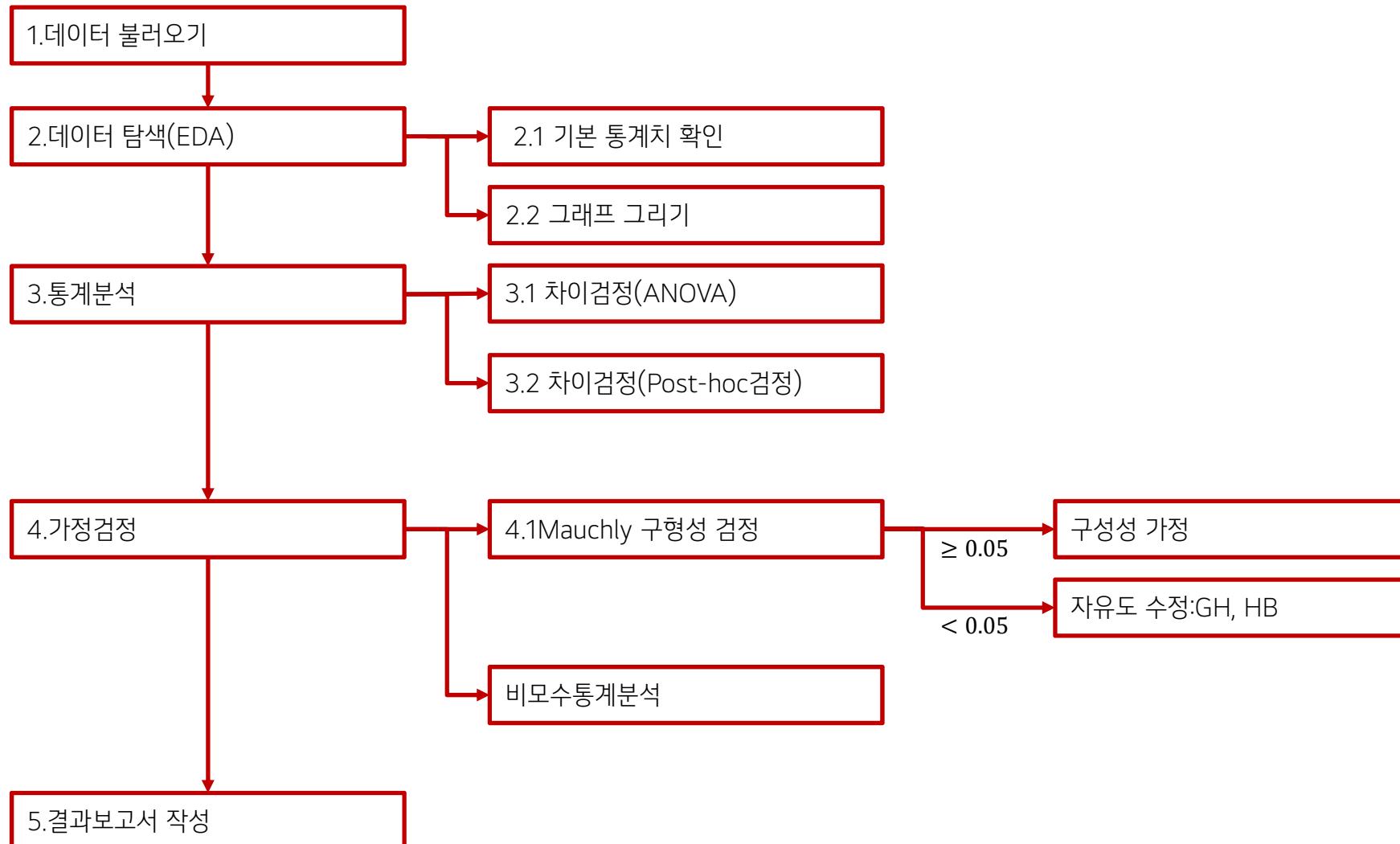
Within Subjects Effects

	Sphericity Correction	Sum of Squares	df	Mean Square	F	p
시점	None	12.978	2	6.489	2.493	0.088
	Greenhouse-Geisser	12.978	1.832	7.084	2.493	0.094
	Huynh-Feldt	12.978	1.908	6.803	2.493	0.091
Residual	None	229.022	88	2.603		
	Greenhouse-Geisser	229.022	80.607	2.841		
	Huynh-Feldt	229.022	83.934	2.729		

Note. Type 3 Sums of Squares

Repeated Measures ANOVA 분석절차

LGE Internal Use Only



08_1.Repeated Measures ANOVA

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following content:

- Section 1: 08_1.Repeated Measures ANOVA**
 - [Link to pingouin.stats.org](https://pingouin-stats.org/build/html/generated/pingouin.rm_anova.html#pingouin.rm_anova)
- Section 2: 1. 기본 package 설정**

```
[1] # 그래프에서 한글 폰트 인식하기  
!sudo apt-get install -y fonts-nanum  
!sudo fc-cache -fv  
!rm ~/.cache/matplotlib -rf  
  
[2] !pip install pingouin  
  
# *** 세션 다시 시작
```
- Section 3: # 1.기본**

```
[1] # 1.기본  
import numpy as np # numpy 패키지 가져오기  
import matplotlib.pyplot as plt # 시각화 패키지 가져오기  
import seaborn as sns # 시각화
```
- Section 4: # 2.데이터 가져오기**

```
[1] import pandas as pd # csv -> dataframe으로 전환
```
- Section 5: # 3.통계분석 package**

```
[1] import pingouin as pg  
from scipy import stats  
import statsmodels.api as sm
```
- Section 6: # 기본세팅**

```
[2] # 기본세팅  
# 테마 설정  
sns.set_theme(style = "darkgrid")
```

At the bottom of the notebook, there is a status bar showing "To=08_1.R.." and "0초 오전 10:23에 완료됨". The top right corner of the window has a toolbar with various icons.

2.데이터 불러오기

LGE Internal Use Only

2.데이터 불러오기

2.1 데이터 프레임으로 저장

- 원본데이터(csv)를 dataframe 형태로 가져오기(pandas)

[3] rma_df = pd.read_csv('https://raw.githubusercontent.com/leecho-bigdata/statistics-python/main/08_1.RMA.csv', encoding="cp949")
rma_df.head()

	id	시간	점수1	점수2	점수3
0	1	1	63	63	63
1	2	1	60	60	60
2	3	1	61	61	61
3	4	1	57	57	57
4	5	1	58	58	58

Next steps: View recommended plots

2.2 범주형 변수 처리

- 가변수 처리시 문자로 처리를 해야 변수명 구분이 쉬움

[4] rma_df['시간'].replace({1:'사전점수', 2:'중간점수', 3:'최종점수'}, inplace=True)
rma_df['시간'] = rma_df['시간'].astype('category')

rma_df.head()

	id	시간	점수1	점수2	점수3
0	1	사전점수	63	63	63
1	2	사전점수	60	60	60

✓ 0초 오후 8:49에 완료됨

3. 기술통계

LGE Internal Use Only

```
✓ 0초 [8] # 그룹별 기술통계
rma_df.groupby('시간')[["점수1"]].describe().round(3)

    count      mean      std      min      25%      50%      75%      max
시간
사전점수   45.0  60.200  3.216  53.0  58.0  60.0  63.0  67.0
중간점수   45.0  60.956  3.796  53.0  58.0  60.0  64.0  70.0
최종점수   45.0  60.644  2.451  53.0  59.0  61.0  62.0  66.0

✓ 0초 [9] # 분석변수가 여러개 일 때
num_feature = ['점수1', '점수2', '점수3']
for num in num_feature:
    print("----", num, "----")
    results = rma_df.groupby("시간")[num].describe().round(3)
    print(results, "\n")

---- 점수1 ----
    count      mean      std      min      25%      50%      75%      max
시간
사전점수   45.0  60.200  3.216  53.0  58.0  60.0  63.0  67.0
중간점수   45.0  60.956  3.796  53.0  58.0  60.0  64.0  70.0
최종점수   45.0  60.644  2.451  53.0  59.0  61.0  62.0  66.0

---- 점수2 ----
    count      mean      std      min      25%      50%      75%      max
시간
사전점수   45.0  60.200  3.216  53.0  58.0  60.0  63.0  67.0
중간점수   45.0  60.956  3.796  53.0  58.0  60.0  64.0  70.0
최종점수   45.0  68.133  1.829  65.0  67.0  68.0  69.0  73.0

---- 점수3 ----
    count      mean      std      min      25%      50%      75%      max
시간
사전점수   45.0  60.200  3.216  53.0  58.0  60.0  63.0  67.0
중간점수   45.0  60.956  3.796  53.0  58.0  60.0  64.0  70.0
최종점수   45.0  66.911  4.912  50.0  66.0  68.0  69.0  73.0
```

4.ANOVA

▼ 4.ANOVA(RM)

▼ 4.1 구형성일때, 차이가 없을때

```
[10] # 구형성일때, 차이가 없을때
# correction = False
pg.rm_anova(dv = '점수1',
            within = '시간',
            subject = "id",
            correction = False,
            data = rma_df,
            detailed = True).round(3)
```

Source	SS	DF	MS	F	p-unc	ng2	eps
0 시간	12.978	2	6.489	2.493	0.088	0.009	0.916
1 Error	229.022	88	2.603	NaN	NaN	NaN	NaN

```
[11] # 그래프
g = sns.catplot(x = "시간",
                 y = "점수1",
                 kind = "point",
                 data = rma_df)
```

4.ANOVA

▼ 4.2 구형성일때, 차이가 있을때

▼ 4.2.1 ANOVA(RM)

```
[12] # 구형성일때, 차이가 있을때
# correction = False
pg.rm_anova(dv = '점수2',
             within = '시간',
             subject = "id",
             correction = False,
             data = rma_df,
             detailed = True).round(3)
```

Source	SS	df	MS	F	p-unc	ng2	eps
0 시간	1725.437	2	862.719	264.93	0.0	0.583	0.956
1 Error	286.563	88	3.256	Nan	Nan	Nan	Nan

▼ 4.2.2 사후분석(post-hoc, Multicom)

```
[13] # post-hoc test(RMA)
pg.pairwise_tests(dv = '점수2',
                   within = '시간',
                   subject = "id",
                   data = rma_df).round(3)
```

Contrast	A	B	Paired	Parametric	T	dof	alternative	p-unc	BF10	hedges
0 시간	사전점수	중간점수	True	True	-1.994	44.0	two-sided	0.052	0.983	-0.213
1 시간	사전점수	최종점수	True	True	-23.038	44.0	two-sided	0.000	9.047e+22	-3.006
2 시간	중간점수	최종점수	True	True	-17.307	44.0	two-sided	0.000	1.456e+18	-2.389

✓ [14] # 그림 표

✓ 0초 오후 8:49에 완료됨

4.ANOVA

▼ 4.2.2 사후분석(post-hoc, Multicom)

```
[13] # post-hoc test(RMA)
pg.pairwise_tests(dv = '점수2',
                   within = '시간',
                   subject = "id",
                   data = rma_df).round(3)
```

Contrast	A	B	Paired	Parametric	T	dof	alternative	p-unc	BF10	hedges
0	시간	사전점수	중간점수	True	True	-1.994	44.0	two-sided	0.052	0.983 -0.213
1	시간	사전점수	최종점수	True	True	-23.038	44.0	two-sided	0.000	9.047e+22 -3.006
2	시간	중간점수	최종점수	True	True	-17.307	44.0	two-sided	0.000	1.456e+18 -2.389

```
[14] # 그래프
sns.catplot(x = "시간",
             y = "점수2",
             kind = "point",
             data = rma_df)

<seaborn.axisgrid.FacetGrid at 0x7fa0c52153f0>
```

The figure is a Seaborn FacetGrid plot. The x-axis is labeled '시간' (Time) with categories '0초' (0 seconds) and '오후 8:49에 완료됨' (Completed at 8:49 PM). The y-axis is labeled '점수2' (Score 2) with numerical ticks at 62, 64, 66, and 68. A single data point is plotted, showing a value of approximately 61.5 at 0 seconds and a value of approximately 68.5 at the completed time. Error bars are present on both points.

5. 구형성 검정

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following structure:

- Section 5. 구형성 검정
- Section 5.1 구형성 검정
- Cell 15: # 구형성이면 지금까지 분석한 것이 문제 없음
pg.sphericity(dv = '점수1',
 within = '시간',
 subject = "id",
 data = rma_df)
SpherResults(spher=True, II=0.9082850717962428, chi2=4.136470739071457, dof=2, pval=0.12640864956146516)
- Section 5.2 구형성이 아닐 때(Greenhouse-Geisser)
- Section 5.2.1 ANOVA(Greenhouse-Geisser)
- Cell 16: # 구형성이 아닐때: Greenhouse-Geisser corrected p-value
pg.sphericity(dv = '점수3',
 within = '시간',
 subject = "id",
 data = rma_df)
SpherResults(spher=False, II=0.5076989303880011, chi2=29.14826654385579, dof=2, pval=4.683109812153781e-07)
- Cell 17: # 구형성이 아닐때: Greenhouse-Geisser corrected p-value
correction = True
pg.rm_anova(dv = '점수3',
 within = '시간',
 subject = "id",
 correction = True,
 data = rma_df,
 detailed = True).round(3)

At the bottom of the notebook, there is a status bar with the text "✓ 0초 오후 8:49에 완료됨".

5. 구형성 검정

LGE Internal Use Only

```
[0초] 0초
SpherResults(spher=False, W=0.5076989303880011, chi2=29.14826654385579, dof=2, pval=4.683109812153781e-07)

✓ [17] # 구형성이 아닐때: Greenhouse-Geisser corrected p-value
# correction = True
pg.rma_anova(dv = '점수3',
              within = '시간',
              subject = "id",
              correction = True,
              data = rma_df,
              detailed = True).round(3)

  Source   SS  DF    MS      F  p-unc  p-GG-corr    ng2    eps  sphericity  W-spher  p-spher
  0   시간 1216.178  2  608.089  56.181     0.0        0.0  0.361  0.67      False    0.508     0.0
  1   Error  952.489 88  10.824    NaN    NaN       NaN    NaN    NaN      NaN    NaN    NaN
  ✓ 5.2.2 구형성아닐때 사후분석(post-hoc, Multicom)

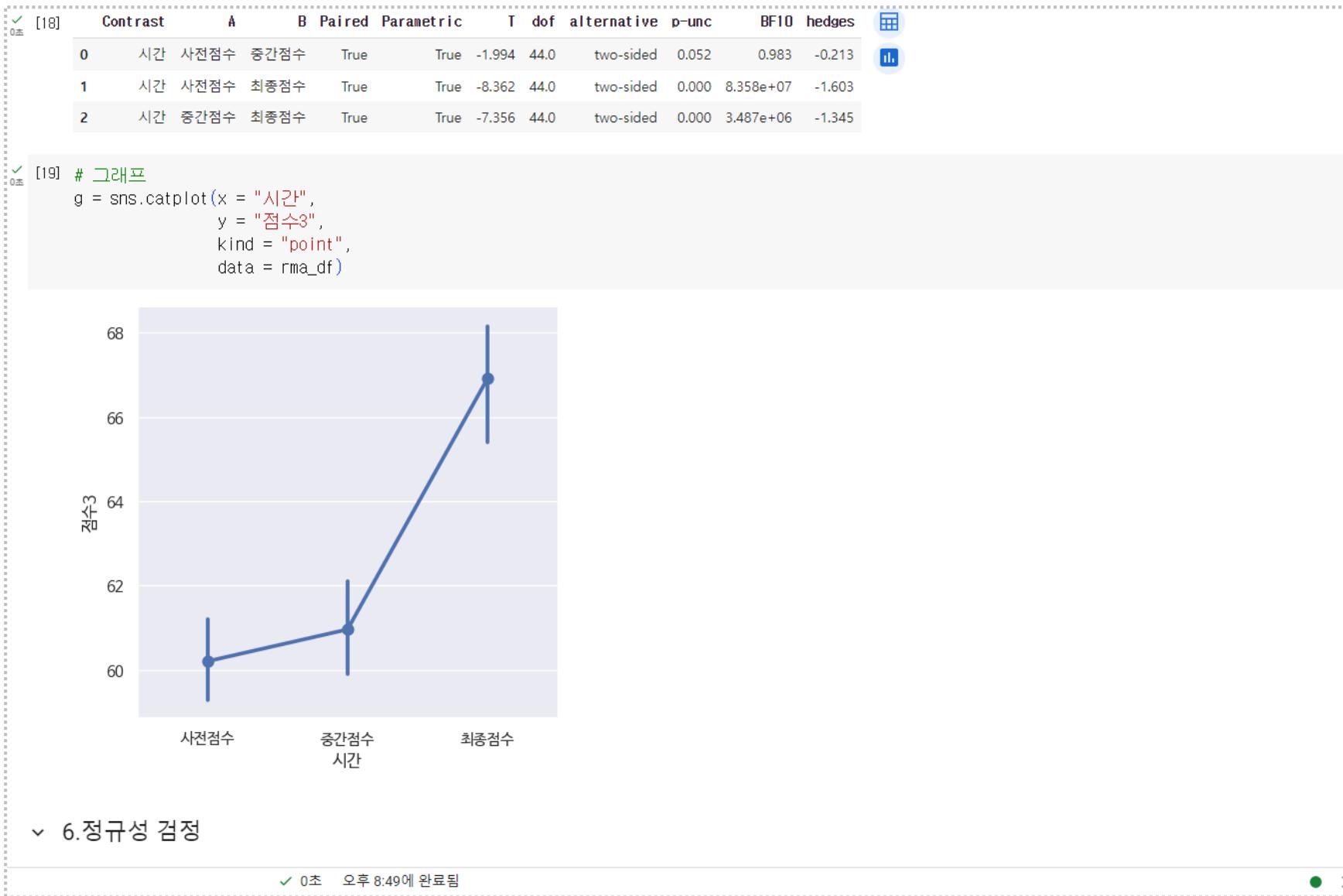
✓ [18] # post-hoc test(RMA)
pg.pairwise_tests(dv = '점수3',
                   within = '시간',
                   subject = "id",
                   data = rma_df).round(3)

  Contrast      A      B Paired Parametric      T  dof alternative  p-unc    BF10  hedges
  0   시간  사전점수  중간점수  True      True -1.994  44.0  two-sided  0.052    0.983 -0.213
  1   시간  사전점수  최종점수  True      True -8.362  44.0  two-sided  0.000  8.358e+07 -1.603
  2   시간  중간점수  최종점수  True      True -7.356  44.0  two-sided  0.000  3.487e+06 -1.345
  ✓ [19] # 그래프
g = sns.catplot(x = "시간",
                 y = "점수3",
                 kind = "point",
                 data = rma_df)
```

✓ 0초 오후 8:49에 완료됨

5. 구형성 검정

LGE Internal Use Only



6.정규성 검정

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following structure:

- Section 6. 정규성 검정**
 - Section 6.1 정규분포 검정**
 - Cell 20:** `[20] ## Python으로 검증하는 package 없음`
 - Section 6.2 비모수일때(friedman)**
 - Cell 21:** `[21] # friedman
pg.friedman(dv = '점수3',
 within = '시간',
 subject = "id",
 data = rma_df).round(3)`
Output:

Source	ddof1	Q	p-unc
Friedman	시간	0.594	2 53.491 0.0
 - Cell 22:** `[22] # 구형성일때, 차이가 없을때
correction = False
pg.rm_anova(dv = '점수3',
 within = '시간',
 subject = "id",
 correction = False,
 data = rma_df,
 detailed = True).round(3)`
Output:

Source	SS	DF	MS	F	p-unc	ng2	eps	
0	시간	1216.178	2	608.089	56.181	0.0	0.361	0.67
1	Error	952.489	88	10.824	NaN	NaN	NaN	NaN

At the bottom of the notebook, there is a status message: **✓ 0초 오후 8:49에 완료됨**.

6.정규성 검정

```

✓ 0초 [22] # 구형성일때, 차이가 없을때
# correction = False
pg.rm_anova(dv = '점수3',
             within = '시간',
             subject = "id",
             correction = False,
             data = rma_df,
             detailed = True).round(3)

      Source   SS   DF    MS     F  p-unc   ng2   eps
0   시간 1216.178   2  608.089  56.181    0.0  0.361  0.67
1   Error  952.489  88   10.824    NaN    NaN    NaN    NaN

✓ 0초 [23] # Non-parametric pairwise paired test (wilcoxon)
pg.pairwise_tests(dv = '점수3',
                   within = '시간',
                   subject = "id",
                   parametric = False,
                   data = rma_df).round(3)

/usr/local/lib/python3.10/dist-packages/scipy/stats/_morestats.py:4088: UserWarning: Exact p-value calculation does not work if there are zeros. Switching to normal approximation.
warnings.warn("Exact p-value calculation does not work if there are "
      Contrast      A      B Paired Parametric  W-val alternative p-unc hedges
0   시간 사전점수 중간점수    True    False  146.5 two-sided  0.076 -0.213
1   시간 사전점수 최종점수    True    False  119.0 two-sided  0.000 -1.603
2   시간 중간점수 최종점수    True    False  124.0 two-sided  0.000 -1.345

```

7.검증결과 그래프

▼ 8.statsmodels 이용

||To=5... ✓ 0초 오후 8:49에 완료됨

8.statsmodels 이용

LGE Internal Use Only

7. 검증결과 그래프

▼ 8.statsmodels 이용

```
[24] import statsmodels.api as sm
     from statsmodels.stats.anova import AnovaRM

     res = AnovaRM(data = rma_df,
                    depvar = '점수3',
                    subject = 'id',
                    within=['시간'])

     print(res.fit())

     Anova
=====
      F Value Num DF Den DF Pr > F
시간 56.1810 2.0000 88.0000 0.0000
=====
```

✓ 0초 오후 8:49에 완료됨

Repeated Measures ANOVA

- ❖ 3개월 어학 프로그램의 효과를 측정하기 위해 사전, 중간점수, 최종점수로 영어실력을 테스트하였다. 사전($M = 60.20$), 중간($M = 60.96$), 최종($M = 60.64$) 등 프로그램은 효과가 없는 것으로 나타났다 ($F = 2.49$, $p=0.088$). (최종점수1 해석)

	사전	중간	최종	F	p
점수	60.20	60.96	60.64	2.49	0.088
Tukey	1	1	1		

Repeated Measures ANOVA

LGE Internal Use Only

- ❖ 3개월 어학 프로그램의 효과를 측정하기 위해 사전, 중간점수, 최종점수로 영어실력을 테스트하였다. 사전($M = 60.20$), 중간($M = 60.96$), 최종($M = 68.13$) 등 시간이 지남에 따라 영어점수가 변화가 있는 것으로 나타났다($F = 264.93, p=0.001$).
- ❖ 시점 간을 비교 분석한 결과, 교육 직후와 중간까지는 프로그램의 효과가 없었으나, 3개월 후 최종점수는 상승한 것으로 나타났다. (최종점수2 해석)

	사전	중간	최종	F	p
점수	60.20	60.96	68.93	264.93	0.001
Tukey	1	1	2		

연습문제

연습문제1

❖ 문제의 정의

- 다음은 호흡과 뇌파와의 관계를 연구한 자료이다. 한사람에게 호흡을 3번(흡호=1:1, 7:3, 3:7)변화하게 하여 뇌파를 측정하였다.
- Ch1_11: 흡호=1:1
- Ch1_73: 흡호=7:3
- Ch1_37: 흡호=3:7
- 뇌파: Ch1, Ch2
- 1. 호흡의 변화에 따라 Ch1, Ch2는 차이가 있는가?
- 2. 구형성 가정이 만족하는가? 만족하지 않다면 GH, HF를 사용하세요.
- 08_2.EEG.csv

성별	ch1_11	ch1_73	ch1_37	ch2_11
2	0.027	0.007	0.029	0
2	0.055	0.032	0.070	0
2	0.050	0.044	0.044	0
2	0.009	0.046	0.069	0
2	0.040	0.044	0.029	0
2	0.028	0.026	0.034	0
2	0.021	0.074	0.092	0
2	0.044	0.049	0.057	0
2	0.026	0.061	0.019	0
2	0.034	0.090	0.065	0
2	0.042	0.046	0.064	0
2	0.012	0.016	0.043	0
2	0.062	0.054	0.037	0
2	0.041	0.038	0.029	0
2	0.026	0.025	0.073	0
2	0.027	0.048	0.023	0
1	0.045	0.067	0.020	0
2	0.048	0.048	0.060	0
1	0.029	0.061	0.053	0
2	0.084	0.032	0.055	0
2	0.034	0.027	0.054	0
2	0.054	0.041	0.033	0
2	0.021	0.064	0.027	0
2	0.035	0.042	0.057	0
2	0.048	0.035	0.051	0

III. Two Way ANOVA

❖ 문제의 정의

- 치킨의 맛을 결정하는 두 가지 요인은 튀길 때의 기름 온도와 튀기는 방법이다.
- 튀길 때의 온도를 200도(1)와 300도(2)로 하고, 튀기는 방법을 오븐(1)과 기름(2)으로 하여 치킨을 튀긴 후에 사람들에게 맛을 평가하도록 하였다.
- 과연 온도와 방법이 맛을 결정하는데 중요한 요인인가? 이 두 가지 요인들 간의 상호작용효과는 없었는가?
- 09_1.TWA.csv

❖ 가설1

- 귀무가설(H_0): 튀기는 온도와 방법은 상호작용이 없다.

$$H_0: \alpha\beta = 0$$

- 연구가설(H_1): 튀기는 온도와 방법은 상호작용이 있다.

$$H_1: \alpha\beta \neq 0$$

❖ 가설2

- 귀무가설(H_0): 튀기는 온도에 따라 맛은 차이가 없다.

$$H_0: \mu_{A1} = \mu_{A2}$$

- 연구가설(H_1): 온도에 따라 맛은 차이가 있다.

$$H_1: \mu_{A1} \neq \mu_{A2}$$

❖ 가설3

- 귀무가설(H_0): 튀기는 방법에 따라 맛은 차이가 없다.

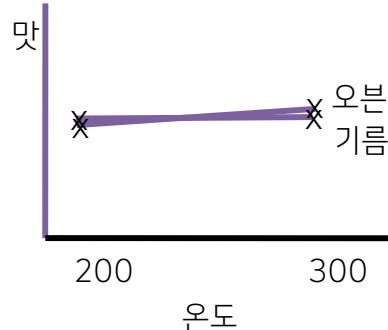
$$H_0: \mu_{B1} = \mu_{B2}$$

- 연구가설(H_1): 튀기는 방법에 따라 맛은 차이가 있다.

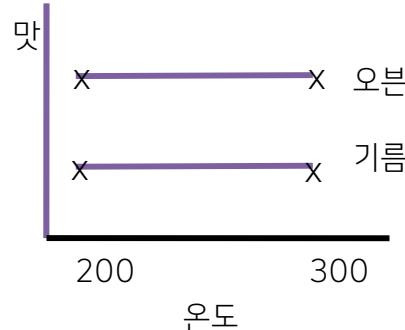
$$H_1: \mu_{B1} \neq \mu_{B2}$$

Two-Way ANOVA

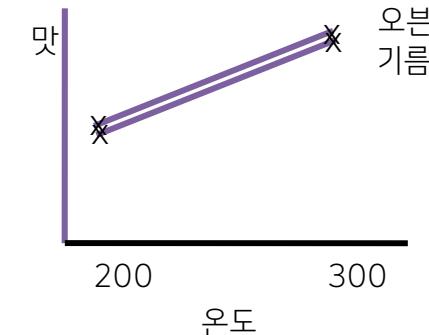
LGE Internal Use Only



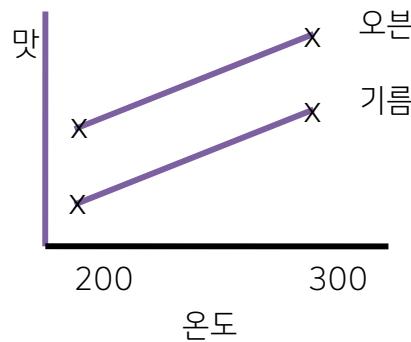
- 방법: X
- 온도: X
- 상호작용: X



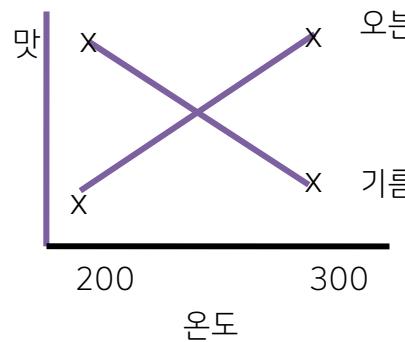
- 방법: O
- 온도: X
- 상호작용: X



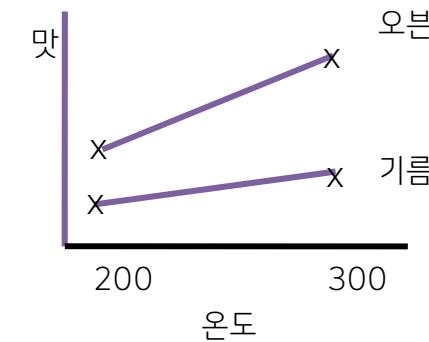
- 방법: X
- 온도: O
- 상호작용: X



- 방법: O
- 온도: O
- 상호작용: X



- 방법: X
- 온도: X
- 상호작용: O



- 방법: O
- 온도: X
- 상호작용: O

Two-Way ANOVA

LGE Internal Use Only

요인 A		온도		평균
		200도	300도	
방법	오븐	90 88 88 91 86 (87.5)	91 88 87 88 90 ... (94.19)	91.07
	기름	90 91 88 87 91 ... (94.0)	86 88 88 92 91 ... (86.0)	90.00
평균		90.86	90.23	

- ❖ 변동(총제곱합)의 분해
 - 총편차의 분해

총편차 = A처리효과 + B처리효과 + AB상호작용효과 + 오차

- ❖ 검정통계량

$$F_{A0} = \frac{SSA}{a-1} = \frac{MSA}{MSE} \sim F_{\alpha}(a-1, ab(r-1))$$

$$F_{B0} = \frac{SSB}{b-1} = \frac{MSB}{MSE} \sim F_{\alpha}(b-1, ab(r-1))$$

$$F_{AB0} = \frac{SSAB}{(a-1)(b-1)} = \frac{MSAB}{MSE} \sim F_{\alpha}((a-1)(b-1), ab(r-1))$$

Two-Way ANOVA

LGE Internal Use Only

요인	제곱합 (SS)	자유도(df)	평균제곱 (MS)	F
A 주효과	$SSA = ar \sum_i^b (\bar{Y}_i - \bar{Y})^2$	$a - 1$	$MSA = \frac{SSA}{a - 1}$	$\frac{MSA}{MSE}$
B 주효과	$SSB = br \sum_j^a (\bar{Y}_j - \bar{Y})^2$	$b - 1$	$MSB = \frac{SSB}{b - 1}$	$\frac{MSB}{MSE}$
상호 작용	$SSAB = r \sum_i^b \sum_j^a (\bar{Y}_{ij} - \bar{Y}_i - \bar{Y}_j + \bar{Y})^2$	$(a - 1)(b - 1)$	$MSAB = \frac{SSAB}{(a - 1)(b - 1)}$	$\frac{MSAB}{MSE}$
잔차	$SSE = \sum_i^b \sum_j^a \sum_k^r (\bar{Y}_{ijk} - \bar{Y}_{ij})^2$	$ab(r - 1)$	$MSE = \frac{SSE}{ab(r - 1)}$	
총계	$SST = \sum_j \sum_i (\bar{Y}_{ij} - \bar{Y})^2$	$abr - 1$		

Two-Way ANOVA

LGE Internal Use Only

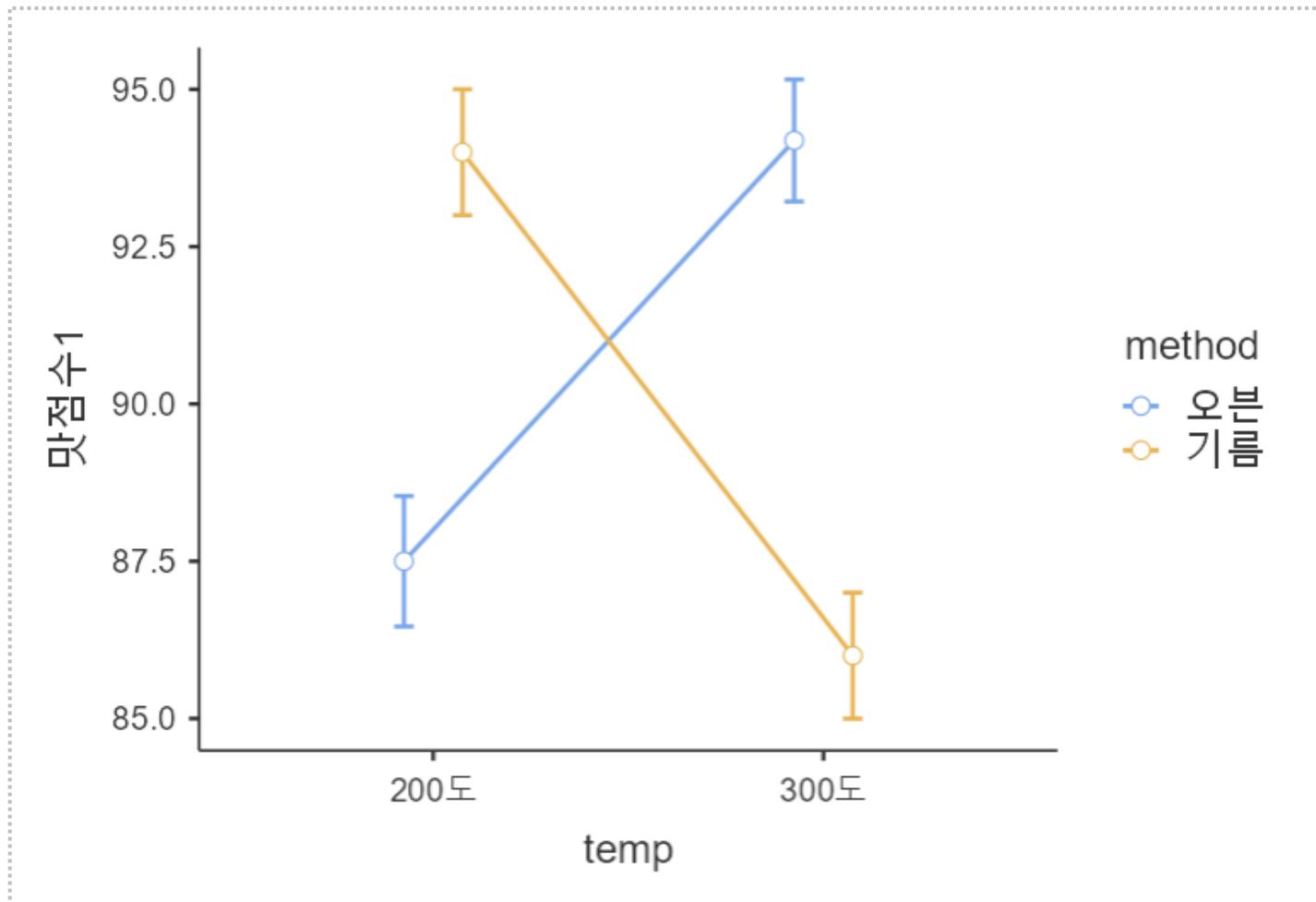
❖ 분산분석표

요인	제곱합 (SS)	자유도(df)	평균제곱 (MS)	F	P-value
방법주효과	10.65	1	10.65	2.84	0.097
온도주효과	6.45	1	6.45	1.72	0.195
상호작용	807.16	1	807.16	215.31	0.001
잔차	209.94	56	3.75		

- 상호작용효과 유의: 상호작용효과 해석
- 상호작용효과 유의하지 않은 경우: 상호작용효과를 오차항에 포함해서 다시 검정

Two-Way ANOVA

LGE Internal Use Only



09_1.Two-Way ANOVA

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following content:

- Section 09_1.Two-Way ANOVA**
 - <https://pingouin-stats.org/build/html/generated/pingouin.anova.html#pingouin.anova>
- Section 1.기본 package 설정**
 - [] # 그래프에서 한글 폰트 인식하기
!sudo apt-get install -y fonts-nanum
!sudo fc-cache -fv
!rm ~/.cache/matplotlib -rf
 - [] !pip install pingouin
*** 런타임 다시 시작
- Section 3초 [1] # 1.기본**
 - import numpy as np # numpy 패키지 가져오기
import matplotlib.pyplot as plt # 시각화 패키지 가져오기
import seaborn as sns # 시각화
 - # 2.데이터 가져오기
import pandas as pd # csv -> dataframe으로 전환
 - # 3.통계분석 package
import pingouin as pg
from scipy import stats
import statsmodels.api as sm
- Section 0초 [2] # 기본세팅**
 - # 테마 설정
sns.set_theme(style = "darkgrid")

At the bottom of the notebook, there is a status bar with the message "✓ 0초 오후 8:56에 완료됨".

2.데이터 불러오기

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following content:

- Section 2.1:** 2.1 데이터 프레임으로 저장
 - 원본데이터(csv)를 dataframe 형태로 가져오기(pandas)
- Code:** [3] twa_df = pd.read_csv('https://raw.githubusercontent.com/leecho-bigdata/statistics-python/main/09_1.TWA.csv', encoding="cp949")
twa_df.head()
- Data Preview:** A table showing the first 5 rows of the dataset.
- Next steps:** View recommended plots
- Section 2.2:** 2.2 범주형 변수 처리
 - 가변수 처리시 문자로 처리를 해야 변수명 구분이 쉬움
- Code:** [4] twa_df[['온도']].replace({1:'200도', 2:'300도'}, inplace=True)
twa_df[['방법']].replace({1:'오븐', 2:'기름'}, inplace=True)
twa_df['온도'] = twa_df['온도'].astype('category')
twa_df['방법'] = twa_df['방법'].astype('category')

twa_df.head()
- Data Preview:** A table showing the first 5 rows of the dataset after categorical conversion.

3. 기술통계

LGE Internal Use Only

```
✓ 3.기술통계

✓ [8] # 그룹별 기술통계
twa_df.groupby(["온도","방법"])["맛점수1"].describe().round(3)

      count    mean     std   min   25%   50%   75%   max
온도   방법
200도  기름    15.0  94.000  2.299  90.0  92.50  94.0  95.50  98.0
      오븐    14.0  87.500  2.029  84.0  86.25  87.5  89.00  91.0
300도  기름    15.0  86.000  1.195  84.0  85.00  86.0  87.00  88.0
      오븐    16.0  94.188  2.040  91.0  92.75  94.0  95.25  98.0

✓ [9] # 분석변수가 여러개 일 때
num_feature = ['맛점수1', '맛점수2', '맛점수3']
for num in num_feature:
    print("----", num, "----")
    results = twa_df.groupby(["온도","방법"])[num].describe().round(3)
    print(results, "#n")

---- 맛점수1 ----
      count    mean     std   min   25%   50%   75%   max
온도   방법
200도  기름    15.0  94.000  2.299  90.0  92.50  94.0  95.50  98.0
      오븐    14.0  87.500  2.029  84.0  86.25  87.5  89.00  91.0
300도  기름    15.0  86.000  1.195  84.0  85.00  86.0  87.00  88.0
      오븐    16.0  94.188  2.040  91.0  92.75  94.0  95.25  98.0

---- 맛점수2 ----
      count    mean     std   min   25%   50%   75%   max
온도   방법
200도  기름    15.0  86.000  1.195  84.0  85.00  86.0  87.00  88.0
      오븐    14.0  92.000  2.386  88.0  90.25  92.0  93.75  96.0
300도  기름    15.0  87.667  2.554  82.0  86.50  88.0  89.00  92.0
      오븐    16.0  94.188  2.040  91.0  92.75  94.0  95.25  98.0

---- 맛점수3 ----
      count    mean     std   min   25%   50%   75%   max
✓ 0초 오후 8:56에 완료됨
```

4.TW-ANOVA

▼ 4.TW-ANOVA

▼ 4.0 independent t-test로 한다면

```
✓ [10] # 차이가 있는 경우
x = twa_df['맛점수1'][twa_df['온도'] == '200도']
y = twa_df['맛점수1'][twa_df['온도'] == '300도']

✓ [11] pg.ttest(x, y, paired = False, alternative = "two-sided", correction = False).round(3)
```

	T	dof	alternative	p-val	CI95%	cohen-d	BF10	power
T-test	0.583	58	two-sided	0.562	[-1.55, 2.82]	0.151	0.303	0.088

```
✓ [12] # 차이가 있는 경우
x = twa_df['맛점수1'][twa_df['방법'] == '오븐']
y = twa_df['맛점수1'][twa_df['방법'] == '기름']

✓ [13] pg.ttest(x, y, paired = False, alternative = "two-sided", correction = False).round(3)
```

	T	dof	alternative	p-val	CI95%	cohen-d	BF10	power
T-test	0.983	58	two-sided	0.33	[-1.1, 3.24]	0.254	0.394	0.162

▼ 4.1 상호작용효과 있을때

```
✓ [14] # 상호작용효과를 검증할때
# ss_type = 3
pg.anova(dy = '맛점수1',
           between = ["온도", "방법"],
           ss_type = 3,
           data = twa_df).round(3)
```

✓ 0초 오후 8:56에 완료됨

4.TW-ANOVA

▼ 4.1 상호작용효과 있을때

```
0초 [14] # 상호작용효과를 검증할때
# ss_type = 3
pg.anova(dv = '맛점수1',
          between = ["온도", "방법"],
          ss_type = 3,
          data = twa_df).round(3)
```

Source	SS	DF	MS	F	p-unc	np2
0 온도	6.446	1.0	6.446	1.719	0.195	0.030
1 방법	10.655	1.0	10.655	2.842	0.097	0.048
2 온도 * 방법	807.158	1.0	807.158	215.306	0.000	0.794
3 Residual	209.938	56.0	3.749	NaN	NaN	NaN

▼ 0초 [15] # post-hoc test: 상호작용효과 있을때

```
pg.pairwise_tests(dv = '맛점수1',
                   between = ["온도", "방법"],
                   data = twa_df).round(3)
```

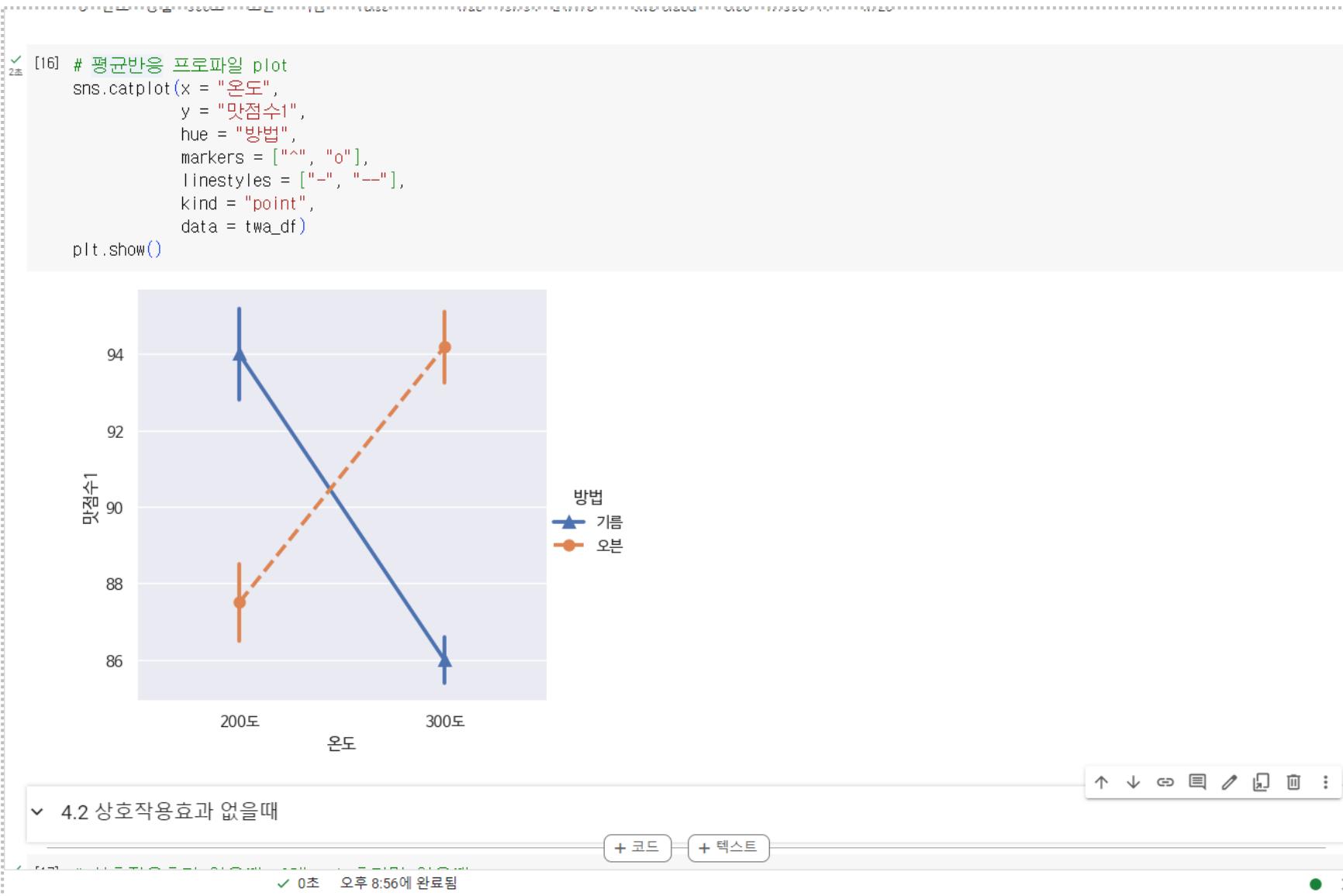
Contrast	온도	A	B	Paired	Parametric	T	dof	alternative	p-unc	BF10	hedges
0 온도	-	200도	300도	False	True	0.586	57.784	two-sided	0.56	0.303	0.149
1 방법	-	오븐	기름	False	True	0.983	58.000	two-sided	0.33	0.394	0.251
2 온도 * 방법	200도	오븐	기름	False	True	-8.085	26.923	two-sided	0.00	7.154e+05	-2.907
3 온도 * 방법	300도	오븐	기름	False	True	13.734	24.478	two-sided	0.00	1.155e+11	4.729

▼ 2초 [16] # 평균반응 프로파일 plot

```
sns.catplot(x = "온도",
            y = "맛점수1",
            hue = "방법",
            markers = ["^", "o"],
```

✓ 0초 오후 8:56에 완료됨

4.TW-ANOVA



4.TW-ANOVA

▼ 4.2 상호작용효과 없을때

```
[0초] [17] # 상호작용효과 없을때, 2개 main효과만 있을때
# ss_type = 2
pg.anova(dy = '맛점수2',
           between = ["온도", "방법"],
           ss_type = 2,
           data = twa_df).round(3)
```

	Source	SS	DF	MS	F	p-unc	np2
0	온도	55.548	1.0	55.548	12.555	0.001	0.183
1	방법	588.872	1.0	588.872	133.094	0.000	0.704
2	온도 * 방법	1.015	1.0	1.015	0.229	0.634	0.004
3	Residual	247.771	56.0	4.424	NaN	NaN	NaN

▼ [18] # 상호작용효과 제거모형

```
[0초] # pg모형은 지원안해서 statsmodels를 이용한 것과 비교
# 거의 유사함
import statsmodels.api as sm
from statsmodels.formula.api import ols

model = ols('맛점수2 ~ C(온도)+C(방법)', data = twa_df)
result = model.fit()
anova_table = sm.stats.anova_lm(result, typ=2)
anova_table
```

	sum_sq	df	F	PR(>F)
C(온도)	55.547513	1.0	12.726643	7.389417e-04
C(방법)	588.871577	1.0	134.917978	1.172735e-16
Residual	248.785820	57.0	NaN	NaN

Next steps: View recommended plots

✓ 0초 오후 8:56에 완료됨

4.TW-ANOVA

Residual 248.785820 57.0 NaN NaN

Next steps: [View recommended plots](#)

✓ [19] # post-hoc test: 상호작용효과 없을 때
`pg.pairwise_tests(dv = '맛점수2',
 between = ["온도", "방법"],
 interaction = False,
 data = twa_df).round(3)`

Contrast	A	B	Paired	Parametric	T	dof	alternative	p-unc	BF10	hedges
0	온도	200도	300도	False	True	-2.184	57.845	two-sided	0.033	1.873 -0.555
1	방법	오븐	기름	False	True	10.708	58.000	two-sided	0.000	1.76e+12 2.729

✓ [20] # 평균반응 프로파일 plot
`sns.catplot(x = "온도",
 y = "맛점수2",
 hue = "방법",
 markers = ["^", "o"],
 linestyles = ["-", "--"],
 kind = "point",
 data = twa_df)
plt.show()`

4.TW-ANOVA

▼ 4.3 상호작용효과 없을때, 1개 main효과만 있을때

```
0초 [21] # 상호작용효과 없을때, 1개 main효과만 있을때
# ss_type = 2
pg.anova(dy = '맛점수3',
           between = ["온도", "방법"],
           ss_type = 2,
           data = twa_df).round(3)
```

Source	SS	DF	MS	F	p-unc	np2
0 온도	807.484	1.0	807.484	215.393	0.000	0.794
1 방법	10.120	1.0	10.120	2.699	0.106	0.046
2 온도 * 방법	6.446	1.0	6.446	1.719	0.195	0.030
3 Residual	209.938	56.0	3.749	NaN	NaN	NaN

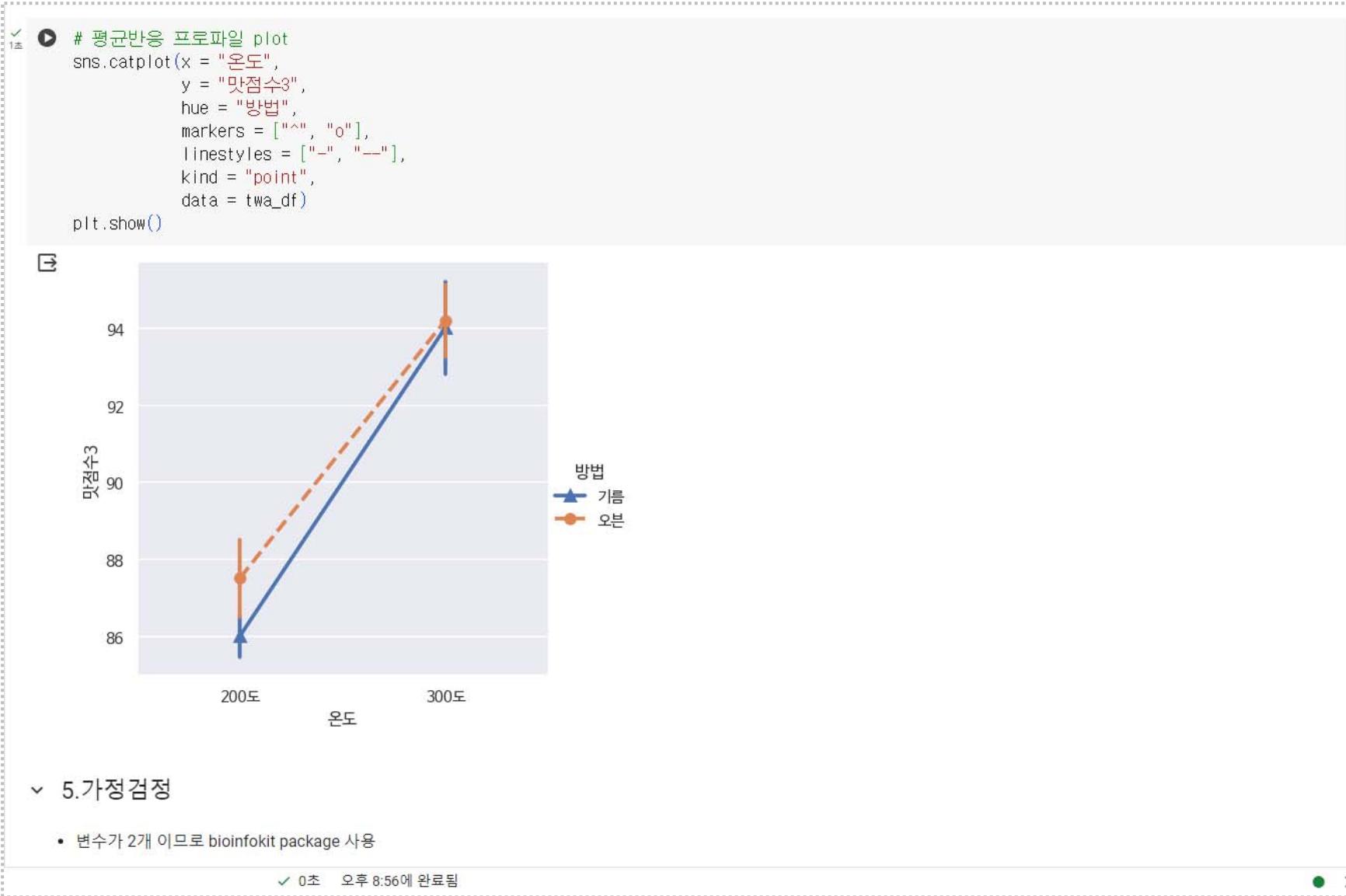

```
0초 [22] # post-hoc test: 상호작용효과 없을 때
pg.pairwise_tests(dy = '맛점수3',
                   between = ["온도", "방법"],
                   interaction = False,
                   data = twa_df).round(3)
```

Contrast	A	B	Paired	Parametric	T	dof	alternative	p-unc	BF10	hedges
0 온도	200도	300도	False	True	-14.527	57.344	two-sided	0.00	4.679e+17	-3.682
1 방법	오븐	기름	False	True	0.983	58.000	two-sided	0.33	0.394	0.251


```
1초 [23] # 평균반응 프로파일 plot
sns.catplot(x = "온도",
             y = "맛점수3",
             hue = "방법",
             markers = ["^", "o"],
             linestyles = ["-", "--"],
             kind = "point")
```

✓ 0초 오후 8:56에 완료됨

4.TW-ANOVA



5.등분산 검정

LGE Internal Use Only

5.등분산검정

- 변수가 2개 이므로 bioinfokit package 사용

✓ [24] !pip install bioinfokit

✓ [25]

```
from bioinfokit.analys import stat
res = stat()
res.levene(df=twa_df, res_var='맛점수1', xfac_var=['온도', '방법'])
res.levene_summary
```

Parameter	Value
0	Test statistics (W) 1.4620
1	Degrees of freedom (Df) 3.0000
2	p value 0.2347

6.정규성 검정

✓ [26]

```
# 잔차의 정규성을 검정해야 함
# pg 모형은 잔차가 없어서 statsmodels를 이용해야 함
import statsmodels.api as sm
from statsmodels.formula.api import ols
```

```
model = ols('맛점수1 ~ C(온도)+C(방법)+C(온도):C(방법)', data = twa_df)
result = model.fit()
pg.normality(result.resid).round(3)
```

	pval	normal
0	0.981	0.475

✓ [26]

✓ 0초 오후 8:56에 완료됨

6.정규성 검정

LGE Internal Use Only

The screenshot shows a Jupyter Notebook cell with the following content:

```
[26] # 잔차의 정규성을 검정해야 함
# pg 모형은 잔차가 없어서 statsmodels를 이용해야 함
import statsmodels.api as sm
from statsmodels.formula.api import ols

model = ols('맛점수1 ~ C(온도)+C(방법)+C(온도):C(방법)', data = twa_df)
result = model.fit()
pg.normality(result.resid).round(3)
```

Below the code, there is a table output:

	pval	normal
0	0.981	0.475

At the bottom of the cell, it says [26].

At the bottom of the notebook interface, there is a status bar with the text "✓ 0초 오후 8:56에 완료됨".

Two-Way ANOVA

LGE Internal Use Only

- ❖ 통닭을 튀기는 온도와 방법에 따라 맛이 차이가 있는지를 분석한 결과, 튀기는 온도와 방법에 따른 상호작용효과가 있었다($F = 215.31, p = 0.001$). 기름일 경우에는 온도가 200도 일 경우에 맛(94.19)이 제일 좋았으며, 오븐일 경우에는 300도로 튀겼을 경우에 맛(94.00)이 좋았다.

		온도		온도	F	P
		200도	300도			
방법	오븐	87.50	94.00	온도	2.84	0.097
	기름	94.19	86.00	방법 * 온도	215.32	0.001

연습문제

연습문제2

❖ 문제의 정의

- 치아성장에 영향을 주는 보충제를 연구하였다. 보충제와 용량은 치아성장에 상호작용 영향을 주었는가?
- supp: 1:OJ(오렌지쥬스), 2:VC(비타민C)
- dose: 500, 1000, 2000
- Tooth Growth.csv

len	supp	dose
4.2	VC	500
11.5	VC	500
7.3	VC	500
5.8	VC	500
6.4	VC	500
10.0	VC	500
11.2	VC	500
11.2	VC	500
5.2	VC	500
7.0	VC	500
16.5	VC	1000
16.5	VC	1000
15.2	VC	1000
17.3	VC	1000
22.5	VC	1000
17.3	VC	1000
13.6	VC	1000
14.5	VC	1000
18.8	VC	1000
15.5	VC	1000
23.6	VC	2000
18.5	VC	2000
33.9	VC	2000
25.5	VC	2000
26.4	VC	2000

IV. Two Way Repeated Measures(Mixed) ANOVA

Two-Way Repeated Measures ANOVA

LGE Internal Use Only

❖ 문제의 정의

- G병원에서는 이번에 새롭게 아로마테라피 치료를 개발하였다.
- 이 치료가 통증에 효과가 있는지를 검증하기 위해
- 새롭게 개발한 치료제로 향기요법을 처치 받는 실험군(2)과
- 일반 향기치료제로 가짜 향기요법을 처치 받는 대조군(1)을 나누고,
- 치료전과 후에 통증이 차이가 있는지를 검증하였다.
- 10_1.TWRMA.csv

Two-Way Repeated Measures ANOVA

LGE Internal Use Only

❖ 가설1

- 귀무가설(H_0): 실험전에 그룹간 통증은 차이가 없다.

$$H_0: \mu_{before_treatment} = \mu_{before_control}$$

- 연구가설(H_1): 실험전에 그룹간 통증은 차이가 있다.

$$H_1: \mu_{before_treatment} \neq \mu_{before_control}$$

❖ 가설2

- 귀무가설(H_0): 그룹과 시점간에는 상호작용이 없다.

$$H_0: \alpha\beta = 0$$

- 연구가설(H_1): 그룹과 시점간에는 상호작용이 있다.

$$H_1: \alpha\beta \neq 0$$

❖ 가설3

- 귀무가설(H_0): 실험후에 그룹간 통증은 차이가 없다.

$$H_0: \mu_{after_treatment} = \mu_{after_control}$$

- 연구가설(H_1): 실험전에 그룹간 통증은 차이가 있다.

$$H_1: \mu_{after_treatment} \neq \mu_{after_control}$$

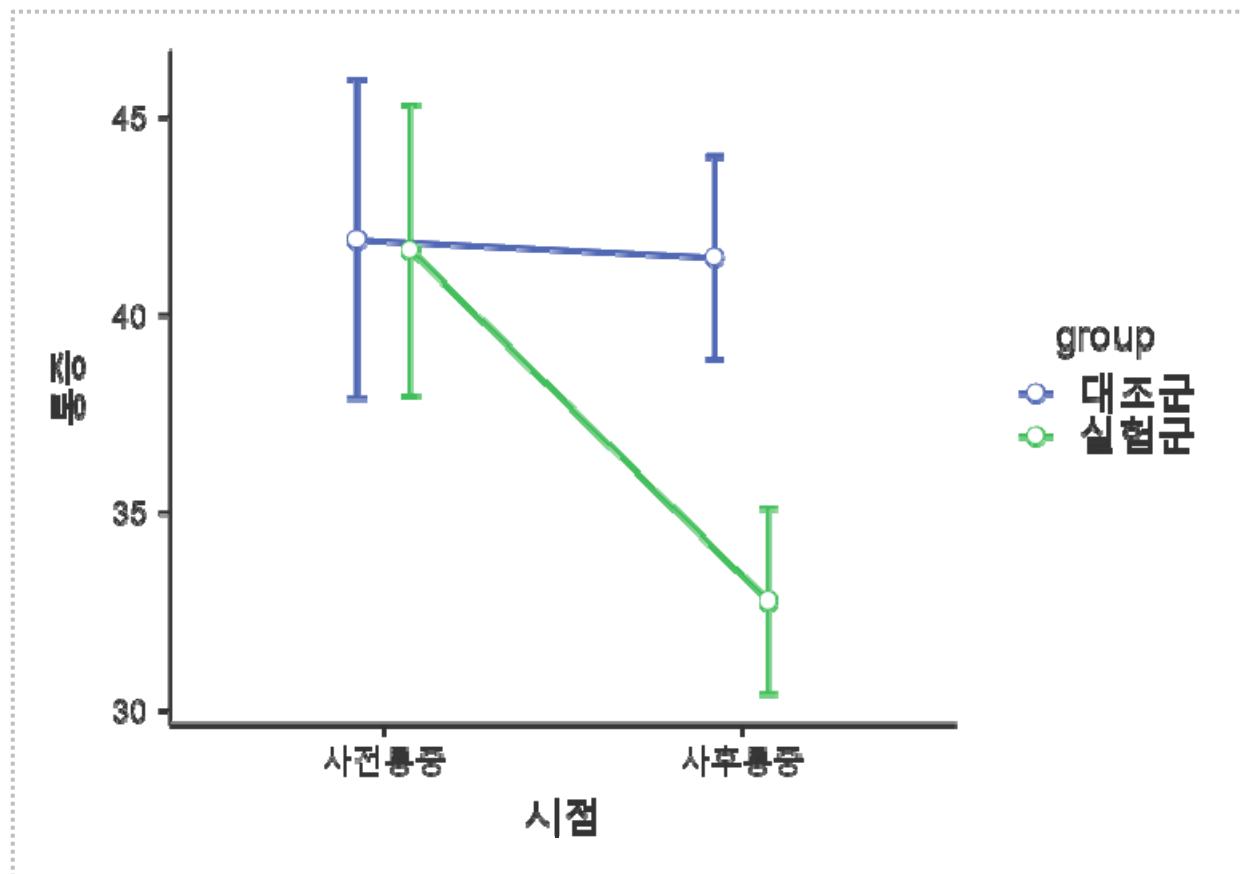
Two-Way Repeated Measures ANOVA

LGE Internal Use Only

❖ 이원 반복측정 분산분석

		요인A (시점)		평균
요인 B	개체	치료전	치료후	
대조군	1	57.16	55.15	41.69
	2	50.88	49.58	
	3	43.42	45.72	
	4	45.34	46.24	
	
실험군	1	40.55	30.56	37.21
	2	33.70	27.54	
	3	35.42	26.21	
	4	33.12	24.45	
	
평균		41.77	36.74	

Two-Way Repeated Measures ANOVA



10_1.Two-Way mixed ANOVA

```
10_1.Two-Way mixed ANOVA
https://pingouin-stats.org/build/html/generated/pingouin.mixed_anova.html#pingouin.mixed_anova

1.기본 package 설정

[ ] # 그래프에서 한글 폰트 인식하기
!sudo apt-get install -y fonts-nanum
!sudo fc-cache -fv
!rm ~/.cache/matplotlib -rf

[ ] !pip install pingouin
# *** 컨타임 다시 시작

[1] # 1.기본
import numpy as np # numpy 패키지 가져오기
import matplotlib.pyplot as plt # 시각화 패키지 가져오기
import seaborn as sns # 시각화

# 2.데이터 가져오기
import pandas as pd # csv -> dataframe으로 전환

# 3.통계분석 package
import pingouin as pg
from scipy import stats
import statsmodels.api as sm

[2] # 기본세팅
# 테마 설정
sns.set_theme(style = "darkgrid")
```

2.데이터 불러오기

LGE Internal Use Only

The screenshot shows a Jupyter Notebook interface with the following content:

- Section 2.1:** 2.1 데이터 프레임으로 저장
 - 원본데이터(csv)를 dataframe 형태로 가져오기(pandas)
- Code Cell [3]:** twma_df = pd.read_csv('https://raw.githubusercontent.com/leechoo-bigdata/statistics-python/main/10_1.TWMA.csv', encoding="cp949")
twma_df.head()
- Data Preview:** A table showing the first 5 rows of the DataFrame.
- Next steps:** View recommended plots
- Section 2.2:** 2.2 범주형 변수 처리
 - 가변수 처리시 문자로 처리를 해야 변수명 구분이 쉬움
- Code Cell [4]:** twma_df['실험그룹'].replace({1:'대조군', 2:'실험군'}, inplace=True)
twma_df['시점'].replace({1:'사전', 2:'사후'}, inplace=True)
twma_df['실험그룹'] = twma_df['실험그룹'].astype('category')
twma_df['시점'] = twma_df['시점'].astype('category')

twma_df.head()
- Data Preview:** A table showing the first 5 rows of the DataFrame after categorical conversion.

2.데이터 불러오기

LGE Internal Use Only

▼ 2.2 범주형 변수 처리

- 가변수 처리시 문자로 처리를 해야 변수명 구분이 쉬움

✓ [4] twma_df['실험그룹'].replace({1:'대조군', 2:'실험군'}, inplace=True)
twma_df['시점'].replace({1:'사전', 2:'사후'}, inplace=True)
twma_df['실험그룹'] = twma_df['실험그룹'].astype('category')
twma_df['시점'] = twma_df['시점'].astype('category')

```
twma_df.head()
```

	id	실험그룹	시점	통증1	통증2	통증3
0	1	대조군	사전	12.32	12.32	12.32
1	2	대조군	사전	57.16	57.16	57.16
2	3	대조군	사전	12.32	12.32	12.32
3	4	대조군	사전	16.62	16.62	16.62
4	5	대조군	사전	18.63	18.63	18.63

Next steps: View recommended plots

▼ 2.3 자료구조 살펴보기

✓ [5] twma_df.shape
(254, 6)

✓ [6] twma_df.info()

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 254 entries, 0 to 253  
Data columns (total 6 columns):  
 #   Column   Non-Null Count  Dtype     
---    
 0   id       254 non-null    int64  ✓ 1초  오후 9:02에 완료됨
```

3. 기술통계

LGE Internal Use Only

```
✓ 0초 [8] # 그룹별 기술통계
twma_df.groupby(["실험그룹", "시점"])["통증1"].describe().round(3)

      count    mean     std    min   25%   50%   75%   max
실험그룹 시점
대조군 사전    58.0  41.914  14.640  12.32  33.54  41.39  49.933  73.05
          사후    58.0  41.458  9.496  21.54  36.22  42.07  49.260  57.53
실험군 사전    69.0  41.639  16.141   5.08  31.22  42.45  51.150  72.57
          사후    69.0  32.782  10.085  10.80  25.42  32.62  38.820  54.58

✓ 0초 [9] # 분석변수가 여러개 일 때
num_feature = ['통증1', '통증2', '통증3']
for num in num_feature:
    print("----", num, "----")
    results = twma_df.groupby(["실험그룹", "시점"])[num].describe().round(3)
    print(results, "#n")

---- 통증1 ----
      count    mean     std    min   25%   50%   75%   max
실험그룹 시점
대조군 사전    58.0  41.914  14.640  12.32  33.54  41.39  49.933  73.05
          사후    58.0  41.458  9.496  21.54  36.22  42.07  49.260  57.53
실험군 사전    69.0  41.639  16.141   5.08  31.22  42.45  51.150  72.57
          사후    69.0  32.782  10.085  10.80  25.42  32.62  38.820  54.58

---- 통증2 ----
      count    mean     std    min   25%   50%   75%   max
실험그룹 시점
대조군 사전    58.0  41.914  14.640  12.32  33.54  41.39  49.933  73.05
          사후    58.0  40.458  9.496  20.54  35.22  41.07  48.260  56.53
실험군 사전    69.0  41.639  16.141   5.08  31.22  42.45  51.150  72.57
          사후    69.0  39.782  10.085  17.80  32.42  39.62  45.820  61.58

---- 통증3 ----
      count    mean     std    min   25%   50%   75%   max
✓ 1초 오후 9:02에 완료됨
```

4.ANOVA

▼ 4.TWM-ANOVA

▼ 4.1 상호작용효과 있을때

```
[10] # 상호작용효과 있을때,
# 구형성일때: correction = False
pg.mixed_anova(dv = '통증1',
                 between = "실험그룹",
                 within = "시점",
                 subject='id',
                 correction = False,
                 data = twma_df).round(3)
```

Source	SS	DF1	DF2	MS	F	p-unc	np2	eps
0 실험그룹	1262.254	1	125	1262.254	4.666	0.033	0.036	NaN
1 시점	1600.527	1	125	1600.527	24.479	0.000	0.164	1.0
2 Interaction	1112.226	1	125	1112.226	17.011	0.000	0.120	NaN

```
[11] # 시점이 2개일때는 구형성 없음
pg.sphericity(dv = '통증1',
               within = '시점',
               subject = "id",
               data = twma_df)

(True, nan, nan, 1, 1.0)
```

```
[12] # post-hoc test: 상호작용효과 있을때
pg.pairwise_tests(dv = '통증1',
                   between = "실험그룹",
                   within = "시점",
                   subject='id',
                   data = twma_df).round(3)
```

x#s... ✓ 1초 오후 9:02에 완료됨

4.ANOVA

```

0초 [12] # post-hoc test: 상호작용효과 있을때
pg.pairwise_tests(dy = '통증1',
                    between = "실험그룹",
                    within = "시점",
                    subject='id',
                    data = twma_df).round(3)

      Contrast 시점   A     B Paired Parametric    T    dof alternative p-unc    BF10 hedges
0       시점 - 사전 사후   True   True  4.660 126.000 two-sided 0.000 1811.627  0.377
1   실험그룹 - 대조군 실험군 False  2.205 124.470 two-sided 0.029  1.685  0.382
2 시점 * 실험그룹 사전 대조군 실험군 False  0.100 124.254 two-sided 0.920  0.191  0.018
3 시점 * 실험그룹 사후 대조군 실험군 False  4.985 123.368 two-sided 0.000  7393.895  0.878

0초 [13] # post-hoc test: 상호작용효과 있을때
# within_first = False: 실험그룹 * 시점
pg.pairwise_tests(dy = '통증1',
                    between = "실험그룹",
                    within = "시점",
                    subject = 'id',
                    within_first = False,
                    data = twma_df).round(3)

      Contrast 실험그룹   A     B Paired Parametric    T    dof alternative p-unc    BF10 hedges
0   실험그룹 - 대조군 실험군 False  2.205 124.47 two-sided 0.029  1.685  0.382
1       시점 - 사전 사후   True   True  4.660 126.00 two-sided 0.000 1811.627  0.377
2 실험그룹 * 시점 대조군 사전 사후   True  0.243  57.00 two-sided 0.809  0.148  0.037
3 실험그룹 * 시점 실험군 사전 사후   True  8.849  68.00 two-sided 0.000 1.277e+10  0.654

1초 [14] # 평균반응 프로파일 plot
sns.catplot(x = "시점",
             y = "통증1",
             hue = "실험그룹",
             markers = ["^", "o"])

```

✓ 1초 오후 9:02에 완료됨

4.ANOVA



4.ANOVA

▼ 4.2 상호작용효과 없을때, 그룹,시점 모두 의미없을때

```
[15] # 상호작용효과 없을때, 그룹,시점 모두 의미없을때
# 구형성일때: correction = False
pg.mixed_anova(dv = '통증2',
                 between = "실험그룹",
                 within = "시점",
                 subject='id',
                 correction = False,
                 data = twma_df).round(3)
```

Source	SS	DF1	DF2	MS	F	p-unc	np2	eps
0 실험그룹	14.237	1	125	14.237	0.053	0.819	0.000	NaN
1 시점	177.948	1	125	177.948	2.722	0.102	0.021	1.0
2 Interaction	2.544	1	125	2.544	0.039	0.844	0.000	NaN

```
[16] # 평균반응 프로파일 plot
sns.catplot(x = "시점",
             y = "통증2",
             hue = "실험그룹",
             markers = ["^", "o"],
             linestyles = ["-", "--"],
             kind = "point",
             data = twma_df)
plt.show()
```

4.ANOVA



4.ANOVA

▼ 4.3 상호작용효과 없을때, 시점만 의미 있을때

```
[17] # 상호작용효과 없을때, 시점만 의미 있을때
# 구형성일때: correction = False
pg.mixed_anova(dv = '통증3',
                 between = "실험그룹",
                 within = "시점",
                 subject='id',
                 correction = False,
                 data = twma_df).round(3)
```

Source	SS	DF1	DF2	MS	F	p-unc	np2	eps
0 실험그룹	0.038	1	125	0.038	0.000	0.991	0.000	NaN
1 시점	1083.485	1	125	1083.485	16.571	0.000	0.117	1.0
2 Interaction	5.637	1	125	5.637	0.086	0.770	0.001	NaN


```
[18] # 시점이 2개일때는 구형성 없음
pg.sphericity(dv = '통증3',
               within = '시점',
               subject = "id",
               data = twma_df)
```

(True, nan, nan, 1, 1.0)


```
[19] # post-hoc test: 상호작용효과 없을때
# interaction = False
pg.pairwise_tests(dv = '통증3',
                  between = "실험그룹",
                  within = "시점",
                  subject='id',
                  interaction = False,
                  data = twma_df).round(3)
```

Contract A B Paired Parametric T df Alternative_p-value DE10_bonfcor

✓ 1초 오후 9:02에 완료됨

4.ANOVA



Two-Way Repeated Measures ANOVA

LGE Internal Use Only

- ❖ 실험군과 대조군별로 새로운 향기요법이 효과가 있는지를 검증한 결과, 실험군과 대조군 사이에는 상호작용효과가 있었다($F = 17.01, p = 0.000$).
- ❖ 사전에 대조군과 실험군은 통증은 차이가 없었으나, 사후에는 실험군의 통증이 더 많이 줄어 들었다

		시점			F	P
		사전통증	사후통증			
집단	대조군	41.91	41.46	시점	20.90	0.000
	실험군	41.64	32.78	그룹	4.67	0.033
				시점 * 그룹	17.01	0.000

연습문제

연습문제3

LGE Internal Use Only

❖ 문제의 정의

- 다음은 호흡과 뇌파와의 관계를 연구한 자료이다. 한사람에게 호흡을 3번(흡호=1:1, 7:3, 3:7)변화하게 하여 뇌파를 측정하였다.
- 성별: 1:남자, 2:여자
- 1. 성별과 호흡의 변화에 따라 Ch1, Ch2는 차이가 있는가?
- 2. 구형성 가정이 만족하는가? 만족하지 않다면 GH, HF를 사용하세요.
- 08_2.EEG.omv파일을 복사해서 사용하세요.

성별	ch1_11	ch1_73	ch1_37	ch2_11
2	0.027	0.007	0.029	0
2	0.055	0.032	0.070	0
2	0.050	0.044	0.044	0
2	0.009	0.046	0.069	0
2	0.040	0.044	0.029	0
2	0.028	0.026	0.034	0
2	0.021	0.074	0.092	0
2	0.044	0.049	0.057	0
2	0.026	0.061	0.019	0
2	0.034	0.090	0.065	0
2	0.042	0.046	0.064	0
2	0.012	0.016	0.043	0
2	0.062	0.054	0.037	0
2	0.041	0.038	0.029	0
2	0.026	0.025	0.073	0
2	0.027	0.048	0.023	0
1	0.045	0.067	0.020	0
2	0.048	0.048	0.060	0
1	0.029	0.061	0.053	0
2	0.084	0.032	0.055	0
2	0.034	0.027	0.054	0
2	0.054	0.041	0.033	0
2	0.021	0.064	0.027	0
2	0.035	0.042	0.057	0
2	0.048	0.035	0.051	0