

ADHD 아동의 기분에 따른 게임 Performance 영향도 분석

1. Attention_4 게임

1.1 기분에 따른 정답률에 차이가 있는지 분석

-등분산검정

```
df=pd.DataFrame(a1111,columns=['기분','A4정답률'])

df['A4정답률'] = pd.to_numeric(df['A4정답률'], errors='coerce')

df_1=df[df.기분==1]['A4정답률'].tolist()
df_2=df[df.기분==2]['A4정답률'].tolist()
df_3=df[df.기분==3]['A4정답률'].tolist()
df_4=df[df.기분==4]['A4정답률'].tolist()
df_5=df[df.기분==5]['A4정답률'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df_1,df_2,df_3,df_4,df_5))
```

LeveneResult(statistic=0.6095751118515526, pvalue=0.6559668984701987)
유의확률이 0.656으로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA 분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	766.572693	191.643173	2.144194	0.074748
Residual	385.0	34410.415905	89.377704	NaN	NaN

유의확률이 0.075로 유의수준 0.05 하에서 유의하지 않음

-> 기분에 따른 정답률에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

1.2 기분에 따른 난이도 평균에 차이가 있는지 분석

-등분산검정

```
df1=pd.DataFrame(a1111,columns=['기분','A4난이도_평균'])

df1['A4난이도_평균'] = pd.to_numeric(df1['A4난이도_평균'], errors='coerce')

df1_1=df1[df1.기분==1]['A4난이도_평균'].tolist()
df1_2=df1[df1.기분==2]['A4난이도_평균'].tolist()
df1_3=df1[df1.기분==3]['A4난이도_평균'].tolist()
df1_4=df1[df1.기분==4]['A4난이도_평균'].tolist()
df1_5=df1[df1.기분==5]['A4난이도_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df1_1,df1_2,df1_3,df1_4,df1_5))
```

LeveneResult(statistic=1.238357021447205, pvalue=0.2940753621230201)
유의확률이 0.294로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA 분석

```

              df      sum_sq  mean_sq      F      PR(>F)
C(기분)      4.0    191.965026  47.991257  4.271052  0.002153
Residual    385.0  4326.014934  11.236402      NaN      NaN
유의확률이 0.002로 유의수준 0.05 하에서 유의함

```

->기분에 따른 난이도 평균에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```

r1=f1.allpairtest(scipy.stats.ttest_ind, method='bonf')
r1[0]
#기분 1과4, 기분1와5 간의 평균차이가 유의미함

```

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	0.6098	0.5426	1.0	False
1.0	3.0	1.8271	0.069	0.6905	False
1.0	4.0	3.3647	0.001	0.0099	True
1.0	5.0	3.1156	0.0022	0.022	True
2.0	3.0	1.1534	0.2501	1.0	False
2.0	4.0	2.7572	0.0067	0.0668	False
2.0	5.0	2.4644	0.015	0.1495	False
3.0	4.0	1.9086	0.0584	0.5843	False
3.0	5.0	1.5072	0.1339	1.0	False
4.0	5.0	-0.5705	0.5703	1.0	False

-> 기분1(아주 좋음)과 기분4(나쁨) 사이에 난이도 평균에 차이가 있음

-> 기분1(아주 좋음)과 기분5(아주 나쁨) 사이에 난이도 평균에 차이가 있음

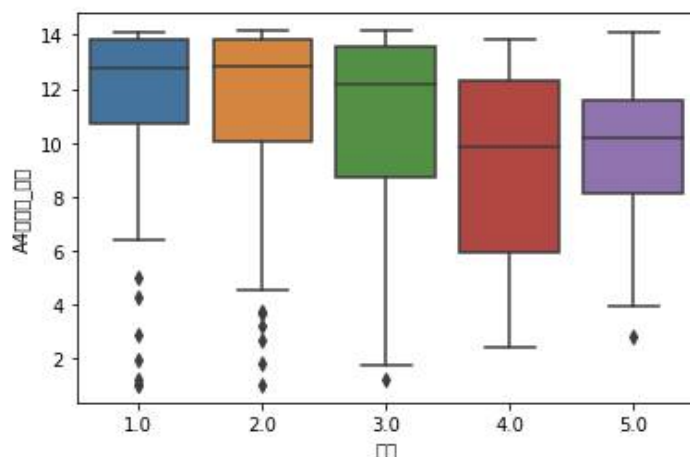
-시각화 (상자그림)

```

import seaborn as sns
sns.boxplot(x='기분', y='A4난이도_평균', data=df1)

```

<AxesSubplot: xlabel='기분', ylabel='A4난이도_평균'>



상자그림 확인 시 대체로 기분이 좋을 때 난이도 평균이 증가하는 경향을 보임

->기분이 좋을 때 게임 수행 능력이 좋은 것으로 생각

1.3 기분에 따른 난이도 표준편차에 차이가 있는지 분석

- 등분산 검정

```
df2=pd.DataFrame(a1111,columns=['기분','A4난이도_표준편차'])
df2['A4난이도_표준편차'] = pd.to_numeric(df2['A4난이도_표준편차'], errors='coerce')
df2_1=df2[df2.기분==1]['A4난이도_표준편차'].tolist()
df2_2=df2[df2.기분==2]['A4난이도_표준편차'].tolist()
df2_3=df2[df2.기분==3]['A4난이도_표준편차'].tolist()
df2_4=df2[df2.기분==4]['A4난이도_표준편차'].tolist()
df2_5=df2[df2.기분==5]['A4난이도_표준편차'].tolist()
```

#등분산성

```
from scipy.stats import levene
print(stats.levene(df2_1,df2_2,df2_3,df2_4,df2_5))
```

LeveneResult(statistic=1.197641693205069, pvalue=0.3113540795480005)

유의확률이 0.3114로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA 분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	24.395121	6.09878	2.435829	0.046813
Residual	385.0	963.955221	2.50378	NaN	NaN

유의확률이 0.0468로 유의수준 0.05하에서 유의함

->기분에 따른 난이도 표준편차에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```
r2=f1.allpairtest(scipy.stats.ttest_ind, method='bonf')
r2[0]
```

#기분 1과4, 기분1와5 간의 평균차이가 유의미함

Test Multiple Comparison ttest_ind FWER=0.05

method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	0.6098	0.5426	1.0	False
1.0	3.0	1.8271	0.069	0.6905	False
1.0	4.0	3.3647	0.001	0.0099	True
1.0	5.0	3.1156	0.0022	0.022	True
2.0	3.0	1.1534	0.2501	1.0	False
2.0	4.0	2.7572	0.0067	0.0668	False
2.0	5.0	2.4644	0.015	0.1495	False
3.0	4.0	1.9086	0.0584	0.5843	False
3.0	5.0	1.5072	0.1339	1.0	False
4.0	5.0	-0.5705	0.5703	1.0	False

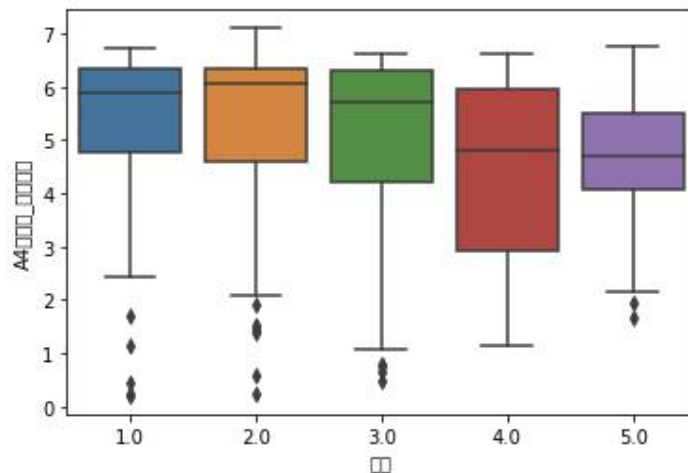
->기분1(아주 좋음)과 기분4(나쁨) 사이에 난이도 표준편차가 차이가 있음

->기분1(아주 좋음)과 기분5(아주 나쁨) 사이에 난이도 표준편차가 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분',y='A4난이도_표준편차',data=df2)
```

```
<AxesSubplot: xlabel='기분', ylabel='A4난이도_표준편차'>
```



상자그림 확인 시 대체로 기분이 좋을 때 난이도 표준편차가 증가하는 경향을 보임
->기분이 좋을 때 높은 게임 수행 난이도를 보여준다고 생각

1.4 기분에 따른 난이도 최대에 차이가 있는지 분석

-등분산검정

```
df3=pd.DataFrame(a1111,columns=['기분','A4난이도_최대'])

df3['A4난이도_최대'] = pd.to_numeric(df3['A4난이도_최대'], errors='coerce')

df3_1=df3[df3.기분==1]['A4난이도_최대'].tolist()
df3_2=df3[df3.기분==2]['A4난이도_최대'].tolist()
df3_3=df3[df3.기분==3]['A4난이도_최대'].tolist()
df3_4=df3[df3.기분==4]['A4난이도_최대'].tolist()
df3_5=df3[df3.기분==5]['A4난이도_최대'].tolist()

#등분산성
from scipy.stats import levene
print(stats.levene(df3_1,df3_2,df3_3,df3_4,df3_5))
```

LeveneResult(statistic=1.7730677755562565, pvalue=0.13348034127934413)

유의확률이 0.1335으로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA 분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	176.10050	44.025125	2.209393	0.067379
Residual	385.0	7671.64309	19.926346	NaN	NaN

유의확률이 0.0674로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 난이도 최대에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

1.5 기분에 따른 결정시간 평균에 차이가 있는지 분석

-등분산검정

```
df4=pd.DataFrame(a1111,columns=['기분','A4결정시간_평균'])
df4['A4결정시간_평균'] = pd.to_numeric(df4['A4결정시간_평균'], errors='coerce')

df4_1=df4[df4.기분==1]['A4결정시간_평균'].tolist()
df4_2=df4[df4.기분==2]['A4결정시간_평균'].tolist()
df4_3=df4[df4.기분==3]['A4결정시간_평균'].tolist()
df4_4=df4[df4.기분==4]['A4결정시간_평균'].tolist()
df4_5=df4[df4.기분==5]['A4결정시간_평균'].tolist()

#등분산성
from scipy.stats import levene
print(stats.levene(df4_1,df4_2,df4_3,df4_4,df4_5))
```

LeveneResult(statistic=0.9133072954673731, pvalue=0.4561124924587757)

유의확률이 0.4561으로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1.208528	0.302132	3.721323	0.005503
Residual	385.0	31.257929	0.081189	NaN	NaN

유의확률이 0.0055로 유의수준 0.05하에서 유의함

->기분에 따른 결정시간 평균에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```
r4=f4.allpairtest(scipy.stats.ttest_ind, method='bonf')
r4[0]
#기분 1과4, 기분3와4 간의 평균차이가 유의미함
```

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-1.6221	0.1063	1.0	False
1.0	3.0	-1.3303	0.1848	1.0	False
1.0	4.0	-4.3797	0.0	0.0002	True
1.0	5.0	-1.8401	0.0677	0.6775	False
2.0	3.0	0.4682	0.6401	1.0	False
2.0	4.0	-2.2975	0.0232	0.232	False
2.0	5.0	-0.4053	0.6859	1.0	False
3.0	4.0	-3.24	0.0015	0.0151	True
3.0	5.0	-0.8654	0.3882	1.0	False
4.0	5.0	1.862	0.0671	0.6713	False

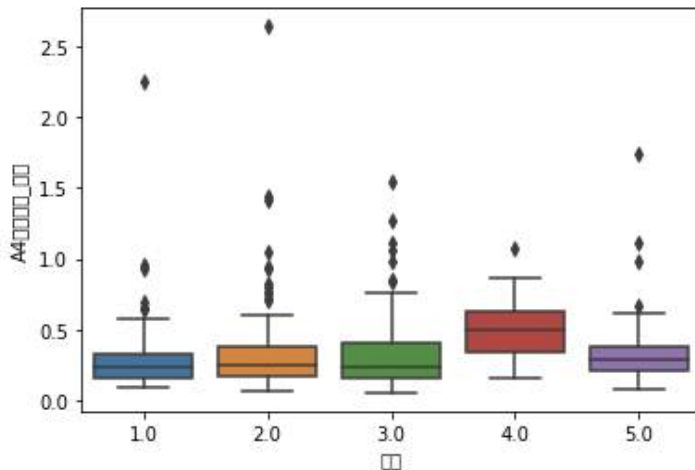
->기분1(아주 좋음)과 기분4(나쁨) 사이에 결정시간 평균에 차이가 있음

->기분3(보통)과 기분4(나쁨) 사이에 결정시간 평균에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분',y='A4결정시간_평균',data=df4)
```

```
<AxesSubplot: xlabel='기분', ylabel='A4결정시간_평균'>
```



상자그림 확인 시 특히 기분 4(나쁨)일 때 결정시간 평균이 증가하는 경향을 보임
->기분이 4(나쁨)일 때 결정에 오랜 시간을 사용한다고 생각

1.6 기분에 따른 결정시간 표준편차에 차이가 있는지 분석

-등분산검정

```
df5=pd.DataFrame(al111,columns=['기분','A4결정시간_표준편차'])
df5['A4결정시간_표준편차'] = pd.to_numeric(df5['A4결정시간_표준편차'], errors='coerce')

df5_1=df5[df5.기분==1]['A4결정시간_표준편차'].tolist()
df5_2=df5[df5.기분==2]['A4결정시간_표준편차'].tolist()
df5_3=df5[df5.기분==3]['A4결정시간_표준편차'].tolist()
df5_4=df5[df5.기분==4]['A4결정시간_표준편차'].tolist()
df5_5=df5[df5.기분==5]['A4결정시간_표준편차'].tolist()

#등분산성
from scipy.stats import levene
print(stats.levene(df5_1,df5_2,df5_3,df5_4,df5_5))
```

```
LeveneResult(statistic=1.4476677792897996, pvalue=0.2176052523835338)
```

유의확률이 0.2176으로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1.855094	0.463774	7.18226	0.000014
Residual	385.0	24.860258	0.064572	NaN	NaN

유의확률이 0.000014로 유의수준 0.05하에서 유의함

->기분에 따른 결정시간 표준편차에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```
r5=f5.allpairtest(scipy.stats.ttest_ind, method='bonf')
r5[0]
```

#기분 1과4, 기분2와4, 기분3과4 간의 평균차이가 유의미함

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-1.8567	0.0647	0.6474	False
1.0	3.0	-2.2059	0.0284	0.2843	False
1.0	4.0	-6.0166	0.0	0.0	True
1.0	5.0	-2.5467	0.0119	0.1189	False
2.0	3.0	-0.4876	0.6263	1.0	False
2.0	4.0	-4.2276	0.0	0.0004	True
2.0	5.0	-1.0611	0.2905	1.0	False
3.0	4.0	-3.3259	0.0011	0.0114	True
3.0	5.0	-0.5894	0.5565	1.0	False
4.0	5.0	2.6629	0.0098	0.0976	False

->기분1(아주 좋음)과 기분4(나쁨) 사이에 결정시간 표준편차에 차이가 있음

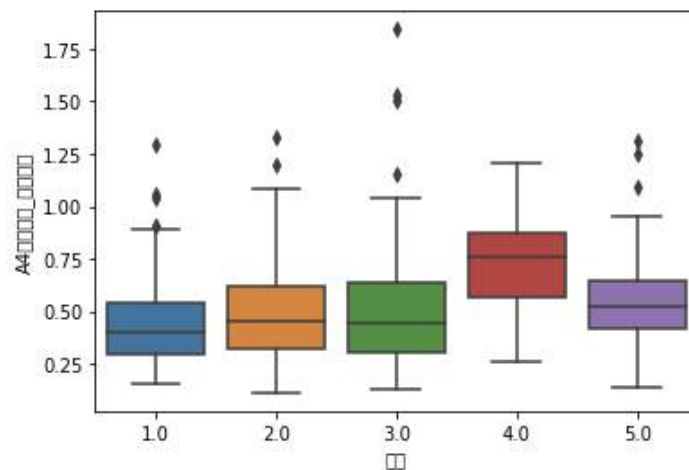
->기분2(좋음)와 기분4(나쁨) 사이에 결정시간 표준편차에 차이가 있음

->기분3(보통)과 기분4(나쁨) 사이에 결정시간 표준편차에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='A4결정시간_표준편차', data=df5)
```

<AxesSubplot: xlabel='기분', ylabel='A4결정시간_표준편차'>



상자그림 확인 시 특히 기분 4(나쁨)일 때 결정시간 표준편차가 증가하는 경향을 보임

->기분이 4(나쁨)일 때 특히 결정에 오랜 시간을 사용한다고 생각

1.7 기분에 따른 터치 횟수에 차이가 있는지 분석

-등분산검정

```
df6=pd.DataFrame(a1111,columns=['기분','A4터치_횟수'])
df6['A4터치_횟수']=pd.to_numeric(df6['A4터치_횟수'],errors='coerce')

df6_1=df6[df6.기분==1]['A4터치_횟수'].tolist()
df6_2=df6[df6.기분==2]['A4터치_횟수'].tolist()
df6_3=df6[df6.기분==3]['A4터치_횟수'].tolist()
df6_4=df6[df6.기분==4]['A4터치_횟수'].tolist()
df6_5=df6[df6.기분==5]['A4터치_횟수'].tolist()
```

#등분산성

```
from scipy.stats import levene
print(stats.levene(df6_1,df6_2,df6_3,df6_4,df6_5))
```

LeveneResult(statistic=2.331236245307095, pvalue=0.05542597070414572)
유의확률이 0.0554으로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

```
df      sum_sq      mean_sq      F      PR(>F)
C(기분)    4.0  2.353429e+05  58835.715798  2.847893  0.023839
Residual 385.0  7.953862e+06  20659.382347      NaN      NaN
```

유의확률이 0.023839로 유의수준 0.05하에서 유의함

->기분에 따른 터치 횟수에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```
r6=f6.allpairtest(scipy.stats.ttest_ind, method='bonf')
r6[0]
#기분 1과4간의 평균차이가 유의미함
```

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

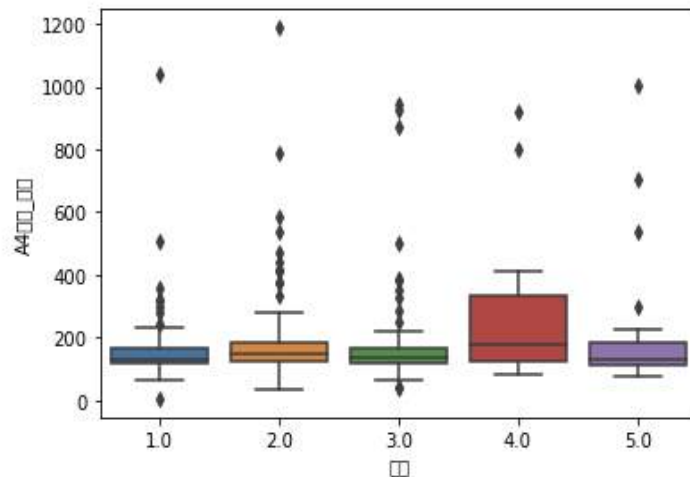
group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-1.8613	0.0641	0.6408	False
1.0	3.0	-0.7041	0.4821	1.0	False
1.0	4.0	-3.5623	0.0005	0.005	True
1.0	5.0	-1.3636	0.1747	1.0	False
2.0	3.0	1.0324	0.3031	1.0	False
2.0	4.0	-1.8222	0.0707	0.7074	False
2.0	5.0	0.0065	0.9948	1.0	False
3.0	4.0	-2.5608	0.0115	0.1154	False
3.0	5.0	-0.7189	0.4733	1.0	False
4.0	5.0	1.353	0.1807	1.0	False

->기분1(아주 좋음)과 기분4(나쁨) 사이에 터치횟수에 차이가 있음

-시각화 (상자 그림)

```
sns.boxplot(x='기분', y='A4터치_횟수', data=df6)
```

<AxesSubplot: xlabel='기분', ylabel='A4터치_횟수'>



상자그림 확인 시 특히 기분 4(나쁨)일 때 터치 횟수가 증가하는 경향을 보임
->기분이 4(나쁨)일 때 불필요한 터치를 많이 한다고 생각

2. Attention_5 게임

2.1 기분에 따른 정답률에 차이가 있는지 분석

-등분산검정

```
df7=pd.DataFrame(a1111,columns=['기분', 'A5정답율'])  
df7['A5정답율'] = pd.to_numeric(df7['A5정답율'], errors='coerce')
```

```
df7_1=df7[df7.기분==1]['A5정답율'].tolist()  
df7_2=df7[df7.기분==2]['A5정답율'].tolist()  
df7_3=df7[df7.기분==3]['A5정답율'].tolist()  
df7_4=df7[df7.기분==4]['A5정답율'].tolist()  
df7_5=df7[df7.기분==5]['A5정답율'].tolist()
```

등분산성

```
from scipy.stats import levene  
print(stats.levene(df7_1,df7_2,df7_3,df7_4,df7_5))
```

LeveneResult(statistic=1.084300319845635, pvalue=0.36391321565220147)

유의확률이 0.3639으로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	72.642666	18.160667	0.086534	0.98659
Residual	385.0	80798.955813	209.867418	NaN	NaN

유의확률이 0.9866으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 정답율에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

2.2 기분에 따른 난이도 평균에 차이가 있는지 분석

-등분산검정

```
df8=pd.DataFrame(a1111,columns=['기분', 'A5난이도_평균'])
df8['A5난이도_평균'] = pd.to_numeric(df8['A5난이도_평균'], errors='coerce')

df8_1=df8[df8.기분==1]['A5난이도_평균'].tolist()
df8_2=df8[df8.기분==2]['A5난이도_평균'].tolist()
df8_3=df8[df8.기분==3]['A5난이도_평균'].tolist()
df8_4=df8[df8.기분==4]['A5난이도_평균'].tolist()
df8_5=df8[df8.기분==5]['A5난이도_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df8_1,df8_2,df8_3,df8_4,df8_5))
```

LeveneResult(statistic=0.6820339187496525, pvalue=0.6047488854675087)

유의확률이 0.6047으로 유의수준 0.05 하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	40.007407	10.001852	0.928935	0.447035
Residual	385.0	4145.299069	10.767011	NaN	NaN

유의확률이 0.447으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 난이도 평균에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

2.3 기분에 따른 난이도 표준편차에 차이가 있는지 분석

-등분산검정

```
df9=pd.DataFrame(a1111,columns=['기분','A5난이도_표준편차'])
df9['A5난이도_표준편차'] = pd.to_numeric(df9['A5난이도_표준편차'], errors='coerce')

df9_1=df9[df9.기분==1]['A5난이도_표준편차'].tolist()
df9_2=df9[df9.기분==2]['A5난이도_표준편차'].tolist()
df9_3=df9[df9.기분==3]['A5난이도_표준편차'].tolist()
df9_4=df9[df9.기분==4]['A5난이도_표준편차'].tolist()
df9_5=df9[df9.기분==5]['A5난이도_표준편차'].tolist()
```

```
# 등분산성
from scipy.stats import levene
print(stats.levene(df9_1,df9_2,df9_3,df9_4,df9_5))
```

LeveneResult(statistic=0.2527799502991334, pvalue=0.9079079305382353)

유의확률이 0.9079로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	9.035269	2.258817	0.805519	0.522192
Residual	385.0	1079.607748	2.804176	NaN	NaN

유의확률이 0.5222으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 난이도 표준편차에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

2.4 기분에 따른 난이도 최대에 차이가 있는지 분석

-등분산검정

```
df10=pd.DataFrame(a1111,columns=['기분','A5난이도_최대'])
df10['A5난이도_최대'] = pd.to_numeric(df10['A5난이도_최대'], errors='coerce')

df10_1=df10[df10.기분==1]['A5난이도_최대'].tolist()
df10_2=df10[df10.기분==2]['A5난이도_최대'].tolist()
df10_3=df10[df10.기분==3]['A5난이도_최대'].tolist()
df10_4=df10[df10.기분==4]['A5난이도_최대'].tolist()
df10_5=df10[df10.기분==5]['A5난이도_최대'].tolist()
```

```
# 등분산성
from scipy.stats import levene
print(stats.levene(df10_1,df10_2,df10_3,df10_4,df10_5))
```

LeveneResult(statistic=0.16263478126239278, pvalue=0.9571517929333058)

유의확률이 0.9572로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	35.319661	8.829915	0.356356	0.839579
Residual	385.0	9539.657263	24.778331	NaN	NaN

유의확률이 0.8396으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 난이도 최대에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

2.5 기분에 따른 결정시간 평균에 차이가 있는지 분석

-등분산검정

```
df11=pd.DataFrame(al111,columns=['기분','A5결정시간_평균'])
df11['A5결정시간_평균'] = pd.to_numeric(df11['A5결정시간_평균'], errors='coerce')

df11_1=df11[df11.기분==1]['A5결정시간_평균'].tolist()
df11_2=df11[df11.기분==2]['A5결정시간_평균'].tolist()
df11_3=df11[df11.기분==3]['A5결정시간_평균'].tolist()
df11_4=df11[df11.기분==4]['A5결정시간_평균'].tolist()
df11_5=df11[df11.기분==5]['A5결정시간_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df11_1,df11_2,df11_3,df11_4,df11_5))
```

LeveneResult(statistic=0.6455405047655085, pvalue=0.6303321844075764)

유의확률이 0.6303으로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1.657331	0.414333	0.692978	0.597171
Residual	385.0	230.192059	0.597901	NaN	NaN

유의확률이 0.5972으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 결정시간 평균에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

2.6 기분에 따른 결정시간 표준편차에 차이가 있는지 분석

-등분산검정

```
df12=pd.DataFrame(al111,columns=['기분','A5결정시간_표준편차'])
df12['A5결정시간_표준편차'] = pd.to_numeric(df12['A5결정시간_표준편차'], errors='coerce')

df12_1=df12[df12.기분==1]['A5결정시간_표준편차'].tolist()
df12_2=df12[df12.기분==2]['A5결정시간_표준편차'].tolist()
df12_3=df12[df12.기분==3]['A5결정시간_표준편차'].tolist()
df12_4=df12[df12.기분==4]['A5결정시간_표준편차'].tolist()
df12_5=df12[df12.기분==5]['A5결정시간_표준편차'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df12_1,df12_2,df12_3,df12_4,df12_5))
```

LeveneResult(statistic=0.8368593752233898, pvalue=0.5023668683278031)

유의확률이 0.5024으로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1.037219	0.259305	0.725074	0.575222
Residual	385.0	137.685781	0.357625	NaN	NaN

유의확률이 0.5752로 유의수준 0.05하에서 유의하지않음

->기분에 따른 결정시간 표준편차에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

2.7 기분에 따른 터치 횟수에 차이가 있는지 분석

-등분산검정

```
df13=pd.DataFrame(al111,columns=['기분','A5터치_횟수'])
df13['A5터치_횟수'] = pd.to_numeric(df13['A5터치_횟수'], errors='coerce')

df13_1=df13[df13.기분==1]['A5터치_횟수'].tolist()
df13_2=df13[df13.기분==2]['A5터치_횟수'].tolist()
df13_3=df13[df13.기분==3]['A5터치_횟수'].tolist()
df13_4=df13[df13.기분==4]['A5터치_횟수'].tolist()
df13_5=df13[df13.기분==5]['A5터치_횟수'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df13_1,df13_2,df13_3,df13_4,df13_5))
```

LeveneResult(statistic=0.840284020089251, pvalue=0.5002303255759355)

유의확률이 0.5002으로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	4.098337e+04	10245.843247	1.259703	0.285344
Residual	385.0	3.131413e+06	8133.539964	NaN	NaN

유의확률이 0.2853으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 터치횟수에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

3. Memory_5 게임

3.1 기분에 따른 정답률에 차이가 있는지 분석

-등분산검정

```
df14=pd.DataFrame(al111,columns=['기분','M5정답률'])
df14['M5정답률'] = pd.to_numeric(df14['M5정답률'], errors='coerce')

df14_1=df14[df14.기분==1]['M5정답률'].tolist()
df14_2=df14[df14.기분==2]['M5정답률'].tolist()
df14_3=df14[df14.기분==3]['M5정답률'].tolist()
df14_4=df14[df14.기분==4]['M5정답률'].tolist()
df14_5=df14[df14.기분==5]['M5정답률'].tolist()
```

등분산성

```
from scipy.stats import levene
print(stats.levene(df14_1,df14_2,df14_3,df14_4,df14_5))
```

LeveneResult(statistic=0.7229902166284039, pvalue=0.5766338066750472)

유의확률이 0.5766으로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	60.005763	15.001441	0.289048	0.885053
Residual	385.0	19981.312263	51.899512	NaN	NaN

유의확률이 0.8850으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 정답률에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

3.2 기분에 따른 난이도 평균에 차이가 있는지 분석

-등분산검정

```
df15=pd.DataFrame(al111,columns=['기분','M5난이도_평균'])
df15['M5난이도_평균'] = pd.to_numeric(df15['M5난이도_평균'], errors='coerce')

df15_1=df15[df15.기분==1]['M5난이도_평균'].tolist()
df15_2=df15[df15.기분==2]['M5난이도_평균'].tolist()
df15_3=df15[df15.기분==3]['M5난이도_평균'].tolist()
df15_4=df15[df15.기분==4]['M5난이도_평균'].tolist()
df15_5=df15[df15.기분==5]['M5난이도_평균'].tolist()
```

등분산성

```
from scipy.stats import levene
print(stats.levene(df15_1,df15_2,df15_3,df15_4,df15_5))
```

LeveneResult(statistic=0.7520781360507722, pvalue=0.5570912580421514)

유의확률이 0.5571로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	3.946171	0.986543	1.653395	0.160162
Residual	385.0	229.720663	0.596677	NaN	NaN

유의확률이 0.1602으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 난이도 평균에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

3.3 기분에 따른 난이도 표준편차에 차이가 있는지 분석

-등분산검정

```
df16=pd.DataFrame(a1111,columns=['기분', 'M5난이도_표준편차'])
df16['M5난이도_표준편차'] = pd.to_numeric(df16['M5난이도_표준편차'], errors='coerce')

df16_1=df16[df16.기분==1]['M5난이도_표준편차'].tolist()
df16_2=df16[df16.기분==2]['M5난이도_표준편차'].tolist()
df16_3=df16[df16.기분==3]['M5난이도_표준편차'].tolist()
df16_4=df16[df16.기분==4]['M5난이도_표준편차'].tolist()
df16_5=df16[df16.기분==5]['M5난이도_표준편차'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df16_1,df16_2,df16_3,df16_4,df16_5))
```

LeveneResult(statistic=1.2304883729414253, pvalue=0.297350409700196)

유의확률이 0.2974로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1.180202	0.295051	1.325994	0.259621
Residual	385.0	85.667424	0.222513	NaN	NaN

유의확률이 0.2596으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 난이도 표준편차에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

3.4 기분에 따른 난이도 최대에 차이가 있는지 분석

-등분산검정

```
df17=pd.DataFrame(a1111,columns=['기분', 'M5난이도_최대'])
df17['M5난이도_최대'] = pd.to_numeric(df17['M5난이도_최대'], errors='coerce')

df17_1=df17[df17.기분==1]['M5난이도_최대'].tolist()
df17_2=df17[df17.기분==2]['M5난이도_최대'].tolist()
df17_3=df17[df17.기분==3]['M5난이도_최대'].tolist()
df17_4=df17[df17.기분==4]['M5난이도_최대'].tolist()
df17_5=df17[df17.기분==5]['M5난이도_최대'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df17_1,df17_2,df17_3,df17_4,df17_5))
```

LeveneResult(statistic=1.8399310243944813, pvalue=0.1204257140919954)

유의확률이 0.1204로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	14.651579	3.662895	1.647102	0.161692
Residual	385.0	856.179190	2.223842	NaN	NaN

유의확률이 0.1617으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 난이도 최대에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

3.5 기분에 따른 남은 제한시간 평균에 차이가 있는지 분석

-등분산검정

```
df18=pd.DataFrame(al111,columns=['기분','W5남은_제한시간_평균'])
df18['W5남은_제한시간_평균'] = pd.to_numeric(df18['W5남은_제한시간_평균'], errors='coerce')

df18_1=df18[df18.기분==1]['W5남은_제한시간_평균'].tolist()
df18_2=df18[df18.기분==2]['W5남은_제한시간_평균'].tolist()
df18_3=df18[df18.기분==3]['W5남은_제한시간_평균'].tolist()
df18_4=df18[df18.기분==4]['W5남은_제한시간_평균'].tolist()
df18_5=df18[df18.기분==5]['W5남은_제한시간_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df18_1,df18_2,df18_3,df18_4,df18_5))
```

LeveneResult(statistic=2.5494738374948493, pvalue=0.03891872320688181)

유의확률이 0.0389로 유의수준0.05하에서 유의함

-> 등분산성 만족X

-Welch's test 분석

```
import pandas as pd
from scipy import stats

# 데이터프레임에서 필요한 열 선택
response = df18['W5남은_제한시간_평균']
independent_var = df18['기분']
# 독립변수의 범주 확인
categories = independent_var.unique()
# 그룹별 데이터 생성
groups = [response[independent_var == category] for category in categories]
# Welch's 분산분석 수행
f_value, p_value = stats.f_oneway(*groups)

# 결과 출력
print("Welch's 분산분석 결과:")
print("F-value:", f_value)
print("p-value:", p_value)
```

Welch's 분산분석 결과:

F-value: 2.6113231744345415

p-value: 0.03517995789726259

유의확률이 0.0352으로 유의수준 0.05하에서 유의함

->기분에 따른 남은 제한시간 평균에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (터키)

```
print(gl8.summary())
```

Multiple Comparison of Means - Tukey HSD, FWER=0.05

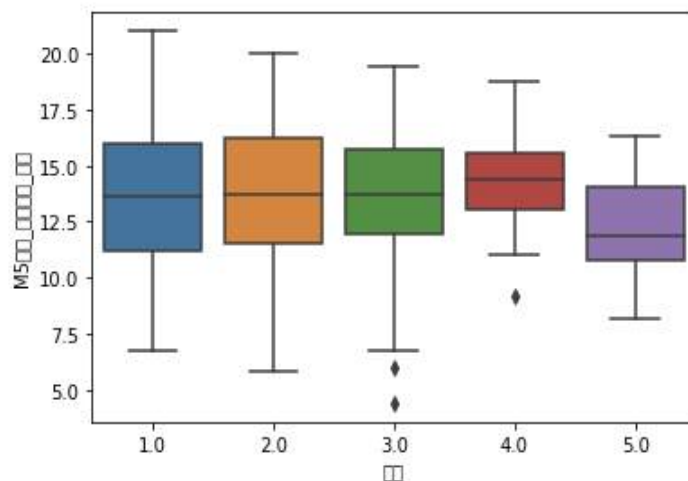
group1	group2	meandiff	p-adj	lower	upper	reject
1.0	2.0	0.0	1.0	-1.0818	1.0818	False
1.0	3.0	-0.0	1.0	-1.0659	1.0659	False
1.0	4.0	-0.0	1.0	-1.6488	1.6488	False
1.0	5.0	0.0	1.0	-1.4854	1.4854	False
2.0	3.0	-0.0	1.0	-1.0936	1.0936	False
2.0	4.0	-0.0	1.0	-1.6669	1.6669	False
2.0	5.0	0.0	1.0	-1.5054	1.5054	False
3.0	4.0	-0.0	1.0	-1.6566	1.6566	False
3.0	5.0	0.0	1.0	-1.494	1.494	False
4.0	5.0	0.0	1.0	-1.9531	1.9531	False

사후검정 해본 결과 남은 제한시간 평균에 차이가 없음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='M5남은_제한시간_평균', data=df18)
```

<AxesSubplot: xlabel='기분', ylabel='M5남은_제한시간_평균'>



상자그림 확인 시 남은 제한 시간 평균이 큰 차이는 없으나

기분이 안 좋을 때 게임 수행 시간이 촉박한 것으로 생각

->기분이 좋지 않을수록 집중력이 떨어질 것이라 유추

3.6 기분에 따른 남은 표준편차에 차이가 있는지 분석

-등분산검정

```
df19=pd.DataFrame(a1111,columns=['기분','M5남은_제한시간_표준편차'])
df19['M5남은_제한시간_표준편차'] = pd.to_numeric(df19['M5남은_제한시간_표준편차'], errors='coerce')

df19_1=df19[df19.기분==1]['M5남은_제한시간_표준편차'].tolist()
df19_2=df19[df19.기분==2]['M5남은_제한시간_표준편차'].tolist()
df19_3=df19[df19.기분==3]['M5남은_제한시간_표준편차'].tolist()
df19_4=df19[df19.기분==4]['M5남은_제한시간_표준편차'].tolist()
df19_5=df19[df19.기분==5]['M5남은_제한시간_표준편차'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df19_1,df19_2,df19_3,df19_4,df19_5))
```

LeveneResult(statistic=0.4396504583247655, pvalue=0.7799474879944174)
유의확률이 0.7799로 유의수준 0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	3.814046	0.953511	0.527692	0.715451
Residual	385.0	695.674442	1.806947	NaN	NaN

유의확률이 0.7155으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 남은 제한시간 표준편차에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

3.7 기분에 따른 남은 제한시간 최대에 차이가 있는지 분석

-등분산검정

```
df20=pd.DataFrame(a1111,columns=['기분','M5남은_제한시간_최대'])
df20['M5남은_제한시간_최대'] = pd.to_numeric(df20['M5남은_제한시간_최대'], errors='coerce')

df20_1=df20[df20.기분==1]['M5남은_제한시간_최대'].tolist()
df20_2=df20[df20.기분==2]['M5남은_제한시간_최대'].tolist()
df20_3=df20[df20.기분==3]['M5남은_제한시간_최대'].tolist()
df20_4=df20[df20.기분==4]['M5남은_제한시간_최대'].tolist()
df20_5=df20[df20.기분==5]['M5남은_제한시간_최대'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df20_1,df20_2,df20_3,df20_4,df20_5))
```

LeveneResult(statistic=1.3816029397411935, pvalue=0.23961169703494653)
유의확률이 0.2396으로 유의수준 0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	20.763709	5.190927	1.607474	0.171638
Residual	385.0	1243.259368	3.229245	NaN	NaN

유의확률이 0.1716으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 남은 제한시간 최대에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

3.7 기분에 따른 터치 횟수에 차이가 있는지 분석

-등분산검정

```
df21=pd.DataFrame(a1111,columns=['기분','M5터치_횟수'])
df21['M5터치_횟수'] = pd.to_numeric(df21['M5터치_횟수'], errors='coerce')

df21_1=df21[df21.기분==1]['M5터치_횟수'].tolist()
df21_2=df21[df21.기분==2]['M5터치_횟수'].tolist()
df21_3=df21[df21.기분==3]['M5터치_횟수'].tolist()
df21_4=df21[df21.기분==4]['M5터치_횟수'].tolist()
df21_5=df21[df21.기분==5]['M5터치_횟수'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df21_1,df21_2,df21_3,df21_4,df21_5))
```

LeveneResult(statistic=2.1751313998101245, pvalue=0.07116015067563412)
유의확률이 0.0711으로 유의수준 0.05하에서 유의하지않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	4.416252e+04	11040.629393	1.642998	0.162697
Residual	385.0	2.587126e+06	6719.806641	NaN	NaN

유의확률이 0.1627으로 유의수준 0.05하에서 유의하지않음

->기분에 따른 터치횟수에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

4. Visual_4 게임

4.1 기분에 따른 전체 정답률에 차이가 있는지 분석

-등분산성

```
df22=pd.DataFrame(a1111,columns=['기분','V4전체_정답률'])
df22['V4전체_정답률'] = pd.to_numeric(df22['V4전체_정답률'], errors='coerce')

df22_1=df22[df22.기분==1]['V4전체_정답률'].tolist()
df22_2=df22[df22.기분==2]['V4전체_정답률'].tolist()
df22_3=df22[df22.기분==3]['V4전체_정답률'].tolist()
df22_4=df22[df22.기분==4]['V4전체_정답률'].tolist()
df22_5=df22[df22.기분==5]['V4전체_정답률'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df22_1,df22_2,df22_3,df22_4,df22_5))
```

LeveneResult(statistic=2.9129319652396046, pvalue=0.021405273056434894)

유의확률이 0.0214로 유의수준0.05하에서 유의함

-> 등분산성 만족X

-Welch's test 분석

```
response = df22['V4전체_정답률']
independent_var = df22['기분']
categories = independent_var.unique()
groups = [response[independent_var == category] for category in categories]
# Welch's 분산분석 수행
f_value, p_value = stats.f_oneway(*groups)
# 결과 출력
print("Welch's 분산분석 결과:")
print("F-value:", f_value)
print("p-value:", p_value)
```

Welch's 분산분석 결과:

F-value: 3.0651629197668933

p-value: 0.016617926267130868

유의확률이 0.01661으로 유의수준 0.05하에서 유의함

->기분에 따른 전체 정답률에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (터키)

```
print(g22.summary())
```

Multiple Comparison of Means - Tukey HSD, FWER=0.05

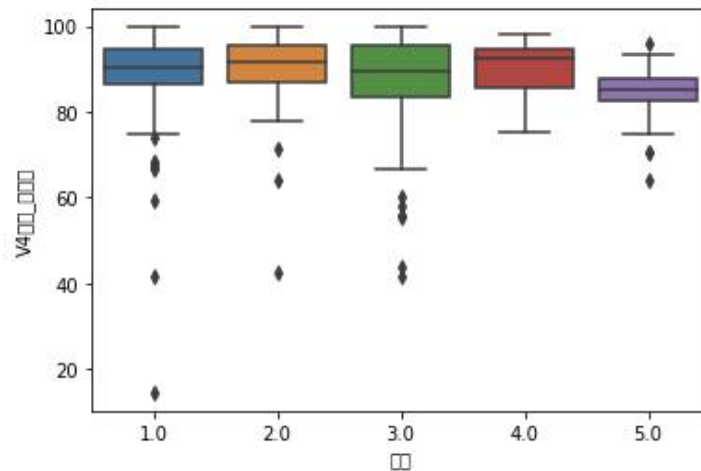
group1	group2	meandiff	p-adj	lower	upper	reject
1.0	2.0	0.0	1.0	-3.703	3.703	False
1.0	3.0	-0.0	1.0	-3.6485	3.6485	False
1.0	4.0	-0.0	1.0	-5.6438	5.6438	False
1.0	5.0	0.0	1.0	-5.0842	5.0842	False
2.0	3.0	-0.0	1.0	-3.7434	3.7434	False
2.0	4.0	-0.0	1.0	-5.7057	5.7057	False
2.0	5.0	0.0	1.0	-5.1528	5.1528	False
3.0	4.0	-0.0	1.0	-5.6705	5.6705	False
3.0	5.0	0.0	1.0	-5.1138	5.1138	False
4.0	5.0	0.0	1.0	-6.6852	6.6852	False

사후검정 해본 결과 전체 정답률에 차이가 없음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4전체_정답율', data=df22)
```

<AxesSubplot: xlabel='기분', ylabel='V4전체_정답율'>



상자그림 확인 시 전체 정답률이 큰 차이는 없으나 기분5(아주 나쁨)일 때 전체 정답률이 특히 감소하는 것을 보아 기분이 안 좋을 때 게임 수행에 대한 정확성이 떨어지는 것으로 생각
->기분이 좋지 않을수록 집중력이 떨어질 것이라 유추

4.2 기분에 따른 왼쪽 정답률에 차이가 있는지 분석

-등분산성

```
df23=pd.DataFrame(a1111,columns=['기분','V4왼쪽_정답율'])
df23['V4왼쪽_정답율'] = pd.to_numeric(df23['V4왼쪽_정답율'], errors='coerce')

df23_1=df23[df23.기분==1]['V4왼쪽_정답율'].tolist()
df23_2=df23[df23.기분==2]['V4왼쪽_정답율'].tolist()
df23_3=df23[df23.기분==3]['V4왼쪽_정답율'].tolist()
df23_4=df23[df23.기분==4]['V4왼쪽_정답율'].tolist()
df23_5=df23[df23.기분==5]['V4왼쪽_정답율'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df23_1,df23_2,df23_3,df23_4,df23_5))
```

LeveneResult(statistic=1.900082960881048, pvalue=0.10970758570656573)

유의확률이 0.1097으로 유의수준0.05하에서 유의하지않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1156.537258	289.134315	2.866848	0.023103
Residual	385.0	38828.961279	100.854445	NaN	NaN

유의확률이 0.0231으로 유의수준 0.05하에서 유의함

->기분에 따른 왼쪽 정답률에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```
r23=f23.allpairtest(scipy.stats.ttest_ind, method='bonf')
r23[0]
#기분 2와5 , 기분 4와5간의 평균차이가 유의미함
```

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-1.3248	0.1867	1.0	False
1.0	3.0	0.384	0.7014	1.0	False
1.0	4.0	-1.4044	0.1624	1.0	False
1.0	5.0	1.7024	0.0908	0.9076	False
2.0	3.0	1.8405	0.0671	0.6712	False
2.0	4.0	-0.8712	0.3853	1.0	False
2.0	5.0	3.7221	0.0003	0.0029	True
3.0	4.0	-1.7926	0.0753	0.7527	False
3.0	5.0	1.5142	0.1322	1.0	False
4.0	5.0	4.3475	0.0	0.0005	True

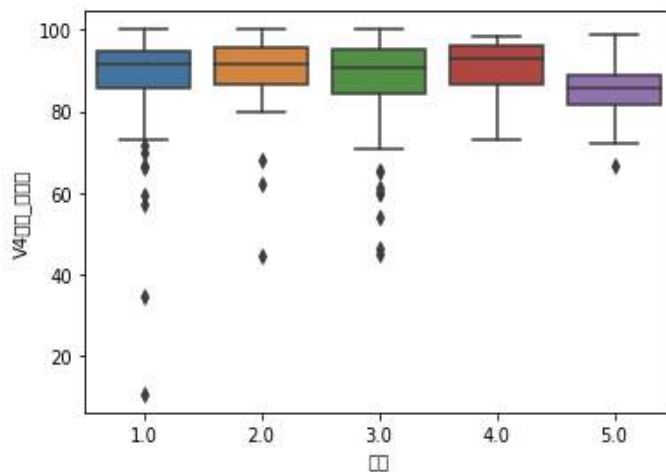
->기분2(좋은)과 기분5(아주 나쁨) 사이에 왼쪽 정답률에 차이가 있음

->기분4(나쁨)와 기분5(아주 나쁨)사이에 왼쪽 정답률에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4왼쪽_정답율', data=df23)
```

<AxesSubplot: xlabel='기분', ylabel='V4왼쪽_정답율'>



상자그림 확인 시 특히 기분5(아주 나쁨)일 때 왼쪽 정답율이 감소하는 경향을 보임

->기분이 5(아주 나쁨)일 때 왼쪽 정답률에 대한 게임 수행 능력이 떨어진다고 생각

4.3 기분에 따른 오른쪽 정답률에 차이가 있는지 분석

-등분산검정

```
df24=pd.DataFrame(a1111,columns=['기분','V4오른쪽_정답률'])
df24['V4오른쪽_정답률'] = pd.to_numeric(df24['V4오른쪽_정답률'], errors='coerce')

df24_1=df24[df24.기분==1]['V4오른쪽_정답률'].tolist()
df24_2=df24[df24.기분==2]['V4오른쪽_정답률'].tolist()
df24_3=df24[df24.기분==3]['V4오른쪽_정답률'].tolist()
df24_4=df24[df24.기분==4]['V4오른쪽_정답률'].tolist()
df24_5=df24[df24.기분==5]['V4오른쪽_정답률'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df24_1,df24_2,df24_3,df24_4,df24_5))
```

LeveneResult(statistic=3.2548581217185912, pvalue=0.012098973341174546)

유의확률이 0.0121로 유의수준 0.05하에서 유의함

->등분산성 만족X

-Welch's test 분석

```
response = df24['V4오른쪽_정답률']
independent_var = df24['기분']
categories = independent_var.unique()
groups = [response[independent_var == category] for category in categories]
# Welch's 분산분석 수행
f_value, p_value = stats.f_oneway(*groups)
# 결과 출력
print("Welch's 분산분석 결과:")
print("F-value:", f_value)
print("p-value:", p_value)
```

Welch's 분산분석 결과:

F-value: 3.1745537318803416

p-value: 0.01384206313875654

유의확률이 0.0138으로 유의수준 0.05하에서 유의함

->기분에 따른 오른쪽 정답률에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (터키)

```
print(g24.summary())
```

Multiple Comparison of Means - Tukey HSD, FWER=0.05

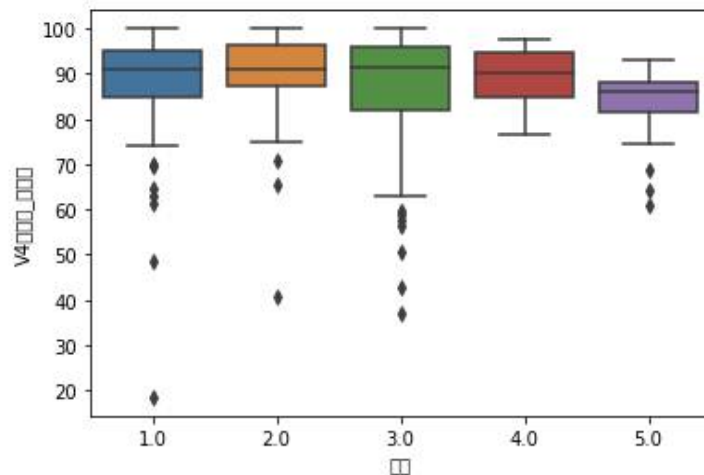
group1	group2	meandiff	p-adj	lower	upper	reject
1.0	2.0	0.0	1.0	-3.867	3.867	False
1.0	3.0	-0.0	1.0	-3.8101	3.8101	False
1.0	4.0	-0.0	1.0	-5.8939	5.8939	False
1.0	5.0	0.0	1.0	-5.3095	5.3095	False
2.0	3.0	-0.0	1.0	-3.9092	3.9092	False
2.0	4.0	-0.0	1.0	-5.9584	5.9584	False
2.0	5.0	0.0	1.0	-5.381	5.381	False
3.0	4.0	-0.0	1.0	-5.9217	5.9217	False
3.0	5.0	0.0	1.0	-5.3403	5.3403	False
4.0	5.0	0.0	1.0	-6.9814	6.9814	False

사후검정 해본 결과 오른쪽 정답률에 차이가 없음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4오른쪽_정답율', data=df24)
```

<AxesSubplot: xlabel='기분', ylabel='V4오른쪽_정답율'>



상자그림 확인 시 오른쪽 정답률이 큰 차이는 없으나 기분5(아주 나쁨)일 때 오른쪽 정답률이 특히 감소하는 것을 보아 기분이 안 좋을 때 게임 수행에 대한 정확성이 떨어지는 것으로 생각

->기분이 좋지 않을수록 집중력이 떨어질 것이라 유추

4.4 기분에 따른 난이도 평균에 차이가 있는지 분석

-등분산성

```
df25=pd.DataFrame(a1111,columns=['기분','V4난이도_평균'])
df25['V4난이도_평균'] = pd.to_numeric(df25['V4난이도_평균'], errors='coerce')

df25_1=df25[df25.기분==1]['V4난이도_평균'].tolist()
df25_2=df25[df25.기분==2]['V4난이도_평균'].tolist()
df25_3=df25[df25.기분==3]['V4난이도_평균'].tolist()
df25_4=df25[df25.기분==4]['V4난이도_평균'].tolist()
df25_5=df25[df25.기분==5]['V4난이도_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df25_1,df25_2,df25_3,df25_4,df25_5))
```

LeveneResult(statistic=4.447774387855553, pvalue=0.0015891352805344078)

유의확률이 0.0016으로 유의수준 0.05하에서 유의함

->등분산성 만족X

-Welch's test 분석

```
response = df25['V4난이도_평균']
independent_var = df25['기분']
categories = independent_var.unique()
groups = [response[independent_var == category] for category in categories]
# Welch's 분산분석 수행
f_value, p_value = stats.f_oneway(*groups)
# 결과 출력
print("Welch's 분산분석 결과:")
print("F-value:", f_value)
print("p-value:", p_value)
```

Welch's 분산분석 결과:

F-value: 4.703149079015305

p-value: 0.0010237139865255742

유의확률이 0.00102으로 유의수준 0.05하에서 유의함

->기분에 따른 난이도 평균에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (터키)

```
print(g25.summary())
```

Multiple Comparison of Means - Tukey HSD, FWER=0.05

=====

group1	group2	meandiff	p-adj	lower	upper	reject
--------	--------	----------	-------	-------	-------	--------

=====

1.0	2.0	0.0	1.0	-1.6856	1.6856	False
1.0	3.0	0.0	1.0	-1.6608	1.6608	False
1.0	4.0	-0.0	1.0	-2.569	2.569	False
1.0	5.0	0.0	1.0	-2.3143	2.3143	False
2.0	3.0	-0.0	1.0	-1.704	1.704	False
2.0	4.0	-0.0	1.0	-2.5972	2.5972	False
2.0	5.0	0.0	1.0	-2.3455	2.3455	False
3.0	4.0	-0.0	1.0	-2.5812	2.5812	False
3.0	5.0	0.0	1.0	-2.3278	2.3278	False
4.0	5.0	0.0	1.0	-3.0431	3.0431	False

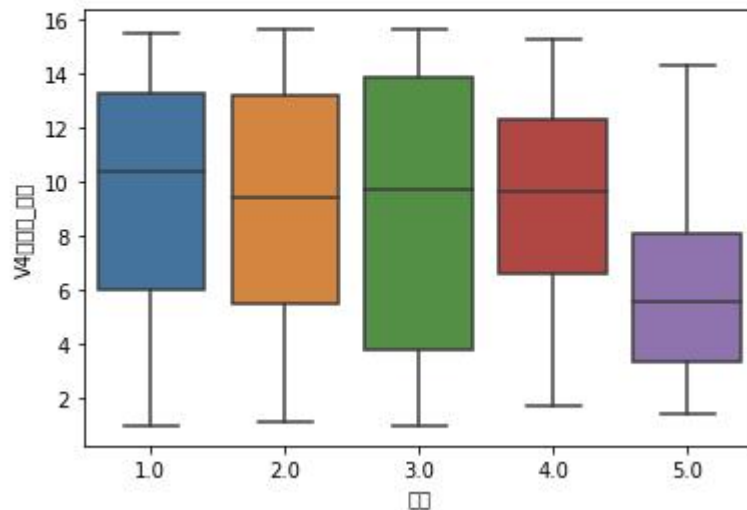
=====

사후검정 해본 결과 난이도 평균에 차이가 없음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4난이도_평균', data=df25)
```

```
<AxesSubplot: xlabel='기분', ylabel='V4난이도_평균'>
```



상자그림 확인 시 난이도 평균이 큰 차이는 없으나 기분5(아주 나쁨)일 때 난이도 평균이 특히 감소하는 것을 보아 기분이 안 좋을 때 게임 수행 능력이 떨어진다고 생각

4.5 기분에 따른 난이도 표준편차에 차이가 있는지 분석

-등분산검정

```
df26=pd.DataFrame(a1111,columns=['기분','V4난이도_표준편차'])
df26['V4난이도_표준편차'] = pd.to_numeric(df26['V4난이도_표준편차'], errors='coerce')

df26_1=df26[df26.기분==1]['V4난이도_표준편차'].tolist()
df26_2=df26[df26.기분==2]['V4난이도_표준편차'].tolist()
df26_3=df26[df26.기분==3]['V4난이도_표준편차'].tolist()
df26_4=df26[df26.기분==4]['V4난이도_표준편차'].tolist()
df26_5=df26[df26.기분==5]['V4난이도_표준편차'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df26_1,df26_2,df26_3,df26_4,df26_5))
```

```
LeveneResult(statistic=3.440267033498115, pvalue=0.00885635258900404)
```

유의확률이 0.0089으로 유의수준 0.05하에서 유의함

->등분산성 만족X

-Welch's test 분석

```
response = df26['V4난이도_표준편차']
independent_var = df26['기분']
categories = independent_var.unique()
groups = [response[independent_var == category] for category in categories]
# Welch's 분산분석 수행
f_value, p_value = stats.f_oneway(*groups)
# 결과 출력
print("Welch's 분산분석 결과:")
print("F-value:", f_value)
print("p-value:", p_value)
```

Welch's 분산분석 결과:

F-value: 4.2984174747695825

p-value: 0.002053895261040629

유의확률이 0.0021으로 유의수준 0.05하에서 유의함

->기분에 따른 난이도 표준편차에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정(터키)

```
print(g26.summary())
```

Multiple Comparison of Means - Tukey HSD, FWER=0.05

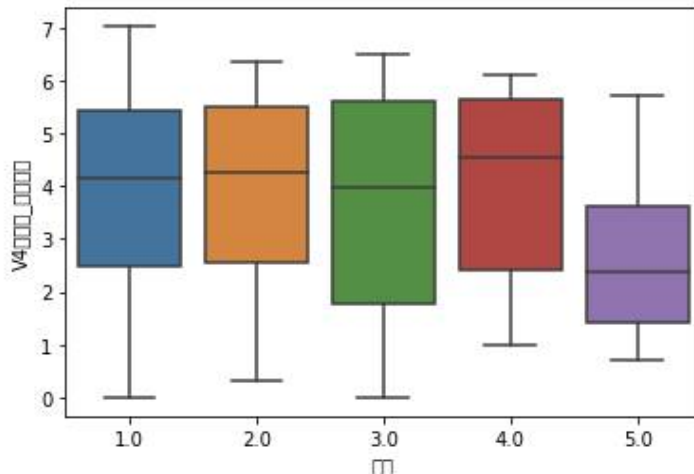
group1	group2	meandiff	p-adj	lower	upper	reject
1.0	2.0	0.0	1.0	-0.6795	0.6795	False
1.0	3.0	-0.0	1.0	-0.6695	0.6695	False
1.0	4.0	-0.0	1.0	-1.0357	1.0357	False
1.0	5.0	0.0	1.0	-0.933	0.933	False
2.0	3.0	-0.0	1.0	-0.6869	0.6869	False
2.0	4.0	-0.0	1.0	-1.047	1.047	False
2.0	5.0	0.0	1.0	-0.9456	0.9456	False
3.0	4.0	-0.0	1.0	-1.0406	1.0406	False
3.0	5.0	0.0	1.0	-0.9384	0.9384	False
4.0	5.0	0.0	1.0	-1.2268	1.2268	False

사후검정 해본 결과 난이도 표준편차에 차이가 없음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4난이도_표준편차', data=df26)
```

```
<AxesSubplot: xlabel='기분', ylabel='V4난이도_표준편차'>
```



상자그림 확인 시 난이도 표준편차가 큰 차이는 없으나 기분5(아주 나쁨)일 때 난이도 표준편차가 특히 감소하는 것을 보아 기분이 안 좋을 때 게임 수행 능력이 떨어진다고 생각

4.6 기분에 따른 난이도 최대에 차이가 있는지 분석

-등분산검정

```
df27=pd.DataFrame(a1111,columns=['기분','V4난이도_최대'])
df27['V4난이도_최대'] = pd.to_numeric(df27['V4난이도_최대'], errors='coerce')
```

```
df27_1=df27[df27.기분==1]['V4난이도_최대'].tolist()
df27_2=df27[df27.기분==2]['V4난이도_최대'].tolist()
df27_3=df27[df27.기분==3]['V4난이도_최대'].tolist()
df27_4=df27[df27.기분==4]['V4난이도_최대'].tolist()
df27_5=df27[df27.기분==5]['V4난이도_최대'].tolist()
```

등분산성

```
from scipy.stats import levene
print(stats.levene(df27_1,df27_2,df27_3,df27_4,df27_5))
```

LeveneResult(statistic=2.3361060021662765, pvalue=0.05499322598128185)

유의확률이 0.055로 유의수준 0.05하에서 유의함

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	568.108433	142.027108	4.121033	0.002784
Residual	385.0	13268.622336	34.463954	NaN	NaN

유의확률이 0.0028으로 유의수준 0.05하에서 유의함

->기분에 따른 난이도 최대에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정(본페로니)

```
r27=f27.allpairtest(scipy.stats.ttest_ind, method='bonf')
r27[0]
```

#기분 1와5 , 기분2와5 , 기분 4와5간의 평균차이가 유의미함

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	0.0001	0.9999	1.0	False
1.0	3.0	1.2118	0.2269	1.0	False
1.0	4.0	-0.4162	0.6779	1.0	False
1.0	5.0	3.8044	0.0002	0.0021	True
2.0	3.0	1.2065	0.229	1.0	False
2.0	4.0	-0.4319	0.6665	1.0	False
2.0	5.0	3.9462	0.0001	0.0013	True
3.0	4.0	-1.117	0.266	1.0	False
3.0	5.0	2.5222	0.0127	0.1275	False
4.0	5.0	3.3741	0.0013	0.0125	True

->기분1(아주 좋음)과 기분5(아주 나쁨) 사이에 난이도 최대에 차이가 있음

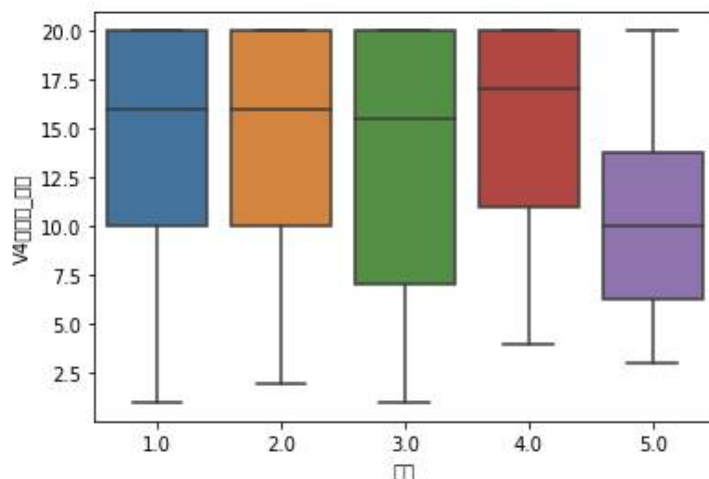
->기분2(좋음)와 기분5(아주 나쁨) 사이에 난이도 최대에 차이가 있음

->기분4(아주 나쁨)과 기분5(아주 나쁨) 사이에 난이도 최대에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4난이도_최대', data=df27)
```

<AxesSubplot: xlabel='기분', ylabel='V4난이도_최대'>



상자그림 확인 시 특히 기분 5(나쁨)일 때 난이도 최대가 감소하는 경향을 보임

->기분이 5(나쁨)일 때 게임에서 게임 수행 능력이 떨어질 것이라고 생각

4.7 기분에 따른 마지막 터치 시간 평균에 차이가 있는지 분석

-등분산검정

```
df28=pd.DataFrame(a1111,columns=['기분','V4마지막_터치시간_평균'])
df28['V4마지막_터치시간_평균'] = pd.to_numeric(df28['V4마지막_터치시간_평균'], errors='coerce')

df28_1=df28[df28.기분==1]['V4마지막_터치시간_평균'].tolist()
df28_2=df28[df28.기분==2]['V4마지막_터치시간_평균'].tolist()
df28_3=df28[df28.기분==3]['V4마지막_터치시간_평균'].tolist()
df28_4=df28[df28.기분==4]['V4마지막_터치시간_평균'].tolist()
df28_5=df28[df28.기분==5]['V4마지막_터치시간_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df28_1,df28_2,df28_3,df28_4,df28_5))
```

LeveneResult(statistic=2.297505003203062, pvalue=0.058514371403286365)

유의확률이 0.05851로 유의수준 0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

```
df      sum_sq  mean_sq      F    PR(>F)
C(기분)    4.0    4.279068  1.069767  5.427041  0.000293
Residual 385.0   75.890386  0.197118    NaN      NaN
```

유의확률이 0.00029으로 유의수준 0.05하에서 유의함

->기분에 따른 마지막 터치시간 평균에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```
r28=f28.allpairtest(scipy.stats.ttest_ind, method='bonf')
r28[0]
#기분 1과5, 2와5, 3과5, 4와5 간의 평균차이가 유의미함
```

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-0.13	0.8967	1.0	False
1.0	3.0	-1.2143	0.226	1.0	False
1.0	4.0	0.0959	0.9238	1.0	False
1.0	5.0	-4.4612	0.0	0.0002	True
2.0	3.0	-1.0973	0.2738	1.0	False
2.0	4.0	0.1889	0.8504	1.0	False
2.0	5.0	-4.5713	0.0	0.0001	True
3.0	4.0	0.8454	0.3994	1.0	False
3.0	5.0	-3.1348	0.0021	0.0209	True
4.0	5.0	-3.6859	0.0005	0.0047	True

->기분1(아주 좋음)과 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음

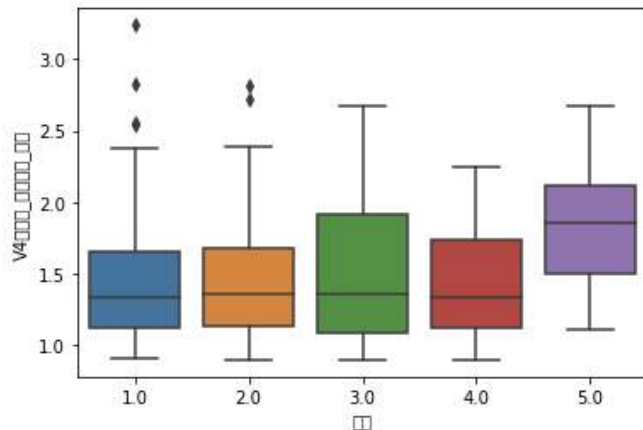
->기분2(좋음)와 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음

->기분3(보통)과 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음
 ->기분4(나쁨)과 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4마지막_터치시간_평균', data=df28)
```

```
<AxesSubplot: xlabel='기분', ylabel='V4마지막_터치시간_평균'>
```



상자그림 확인 시 특히 기분 5(나쁨)일 때 마지막 터치시간 평균이 증가하는 경향을 보임
 ->기분이 5(나쁨)일 때 게임에서 마지막 불필요한 터치가 길다? 라고 생각

4.8 기분에 따른 마지막 터치시간 표준편차에 차이가 있는지 분석

-등분산검정

```
df29=pd.DataFrame(a111,columns=['기분','V4마지막_터치시간_표준편차'])
df29['V4마지막_터치시간_표준편차'] = pd.to_numeric(df29['V4마지막_터치시간_표준편차'], errors='coerce')

df29_1=df29[df29.기분==1]['V4마지막_터치시간_표준편차'].tolist()
df29_2=df29[df29.기분==2]['V4마지막_터치시간_표준편차'].tolist()
df29_3=df29[df29.기분==3]['V4마지막_터치시간_표준편차'].tolist()
df29_4=df29[df29.기분==4]['V4마지막_터치시간_표준편차'].tolist()
df29_5=df29[df29.기분==5]['V4마지막_터치시간_표준편차'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df29_1, df29_2, df29_3, df29_4, df29_5))
```

```
LeveneResult(statistic=2.297505003203062, pvalue=0.058514371403286365)
```

유의확률이 0.0585로 유의수준0.05하에서 유의하지 않음

-> 등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
3(기분)	4.0	4.279068	1.069767	5.427041	0.000293
Residual	385.0	75.890386	0.197118	NaN	NaN

유의확률이 0.000293으로 유의수준 0.05하에서 유의함

->기분에 따른 마지막 터치시간 표준편차에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정(본페로니)

```
r29=f29.allpairtest(scipy.stats.ttest_ind, method='bonf')
r29[0]
```

#기분 1과5, 2와5, 3과5, 4와5 간의 평균차이가 유의미함

Test Multiple Comparison ttest_ind FWER=0.05

method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-0.13	0.8967	1.0	False
1.0	3.0	-1.2143	0.226	1.0	False
1.0	4.0	0.0959	0.9238	1.0	False
1.0	5.0	-4.4612	0.0	0.0002	True
2.0	3.0	-1.0973	0.2738	1.0	False
2.0	4.0	0.1889	0.8504	1.0	False
2.0	5.0	-4.5713	0.0	0.0001	True
3.0	4.0	0.8454	0.3994	1.0	False
3.0	5.0	-3.1348	0.0021	0.0209	True
4.0	5.0	-3.6859	0.0005	0.0047	True

->기분1(아주 좋음)과 기분5(아주 나쁨) 사이에 마지막 터치시간 표준편차에 차이가 있음

->기분2(좋음)와 기분5(아주 나쁨) 사이에 마지막 터치시간 표준편차에 차이가 있음

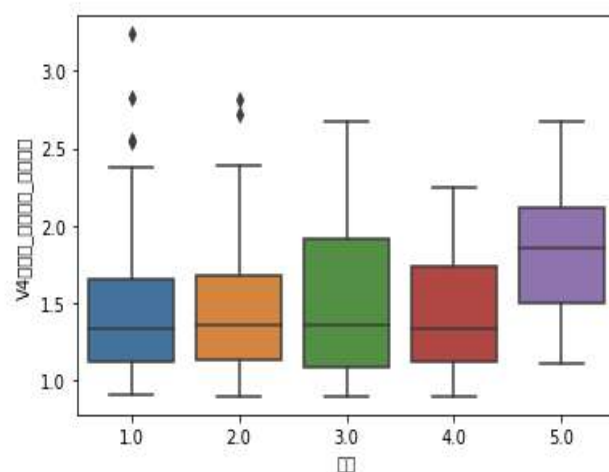
->기분3(보통)과 기분5(아주 나쁨) 사이에 마지막 터치시간 표준편차에 차이가 있음

->기분4(나쁨)과 기분5(아주 나쁨) 사이에 마지막 터치시간 표준편차에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4마지막_터치시간_표준편차', data=df29)
```

<AxesSubplot: xlabel='기분', ylabel='V4마지막_터치시간_표준편차'>



상자그림 확인 시 특히 기분 5(나쁨)일 때 마지막 터치시간 표준편차가 증가하는 경향을 보임

->기분이 5(나쁨)일 때 게임에서 마지막 불필요한 터치가 길어진다? 라고 생각

4.9 기분에 따른 왼쪽 마지막 터치 시간 평균에 차이가 있는지 분석

-등분산검정

```
df30=pd.DataFrame(a111,columns=['기분','V4왼쪽_마지막_터치시간_평균'])
df30['V4왼쪽_마지막_터치시간_평균'] = pd.to_numeric(df30['V4왼쪽_마지막_터치시간_평균'], errors='coerce')

df30_1=df30[df30.기분==1]['V4왼쪽_마지막_터치시간_평균'].tolist()
df30_2=df30[df30.기분==2]['V4왼쪽_마지막_터치시간_평균'].tolist()
df30_3=df30[df30.기분==3]['V4왼쪽_마지막_터치시간_평균'].tolist()
df30_4=df30[df30.기분==4]['V4왼쪽_마지막_터치시간_평균'].tolist()
df30_5=df30[df30.기분==5]['V4왼쪽_마지막_터치시간_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df30_1,df30_2,df30_3,df30_4,df30_5))
```

LeveneResult(statistic=1.525764335951187, pvalue=0.19391735222892725)

유의확률이 0.1939로 유의수준 0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	2.516961	0.629240	4.859902	0.000781
Residual	385.0	49.848223	0.129476	NaN	NaN

유의확률이 0.00078으로 유의수준 0.05하에서 유의함

->기분에 따른 왼쪽 마지막 터치시간 평균에 차이가 있다-> 차이가 있으므로 사후검정 시행
O

-사후검정(본페로니)

```
r30=f30.allpairtest(scipy.stats.ttest_ind, method='bonf')
r30[0]
```

#기분 1과5, 2와5, 3과5 간의 평균차이가 유의미함

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	0.034	0.9729	1.0	False
1.0	3.0	-1.1497	0.2515	1.0	False
1.0	4.0	-0.5451	0.5866	1.0	False
1.0	5.0	-4.2256	0.0	0.0004	True
2.0	3.0	-1.167	0.2446	1.0	False
2.0	4.0	-0.5777	0.5645	1.0	False
2.0	5.0	-4.3122	0.0	0.0003	True
3.0	4.0	0.2127	0.8319	1.0	False
3.0	5.0	-2.9746	0.0034	0.0344	True
4.0	5.0	-2.5354	0.0136	0.1365	False

->기분1(아주 좋음)과 기분5(아주 나쁨) 사이에 왼쪽 마지막 터치시간 평균에 차이가 있음

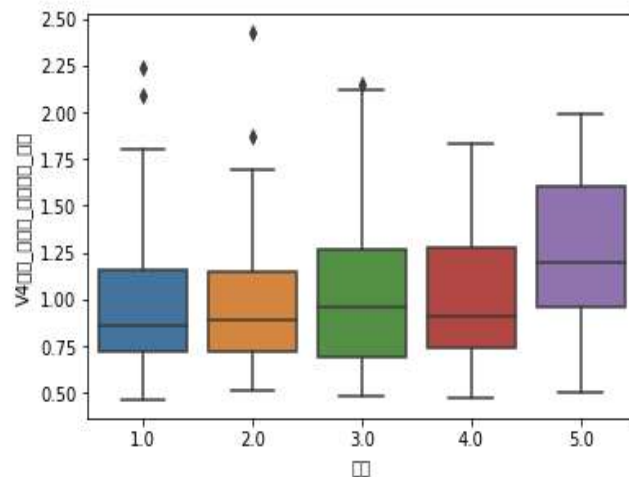
->기분2(좋음)와 기분5(아주 나쁨) 사이에 왼쪽 마지막 터치시간 평균에 차이가 있음

->기분3(보통)과 기분5(아주 나쁨) 사이에 왼쪽 마지막 터치시간 평균에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4왼쪽_마지막_터치시간_평균', data=df30)
```

```
<AxesSubplot: xlabel='기분', ylabel='V4왼쪽_마지막_터치시간_평균'>
```



상자그림 확인 시 특히 기분 5(나쁨)일 때 왼쪽 마지막 터치시간 평균이 증가하는 경향을 보임

->기분이 5(나쁨)일 때 게임에서 마지막 불필요한 왼쪽 터치가 길어진다? 라고 생각

4.10 기분에 따른 왼쪽 마지막 터치 시간 표준편차에 차이가 있는지 분석

-등분산검정

```
df31=pd.DataFrame(al111,columns=['기분','V4왼쪽_마지막_터치시간_표준편차'])
df31['V4왼쪽_마지막_터치시간_표준편차']=pd.to_numeric(df31['V4왼쪽_마지막_터치시간_표준편차'], errors='coerce')

df31_1=df31[df31.기분==1]['V4왼쪽_마지막_터치시간_표준편차'].tolist()
df31_2=df31[df31.기분==2]['V4왼쪽_마지막_터치시간_표준편차'].tolist()
df31_3=df31[df31.기분==3]['V4왼쪽_마지막_터치시간_표준편차'].tolist()
df31_4=df31[df31.기분==4]['V4왼쪽_마지막_터치시간_표준편차'].tolist()
df31_5=df31[df31.기분==5]['V4왼쪽_마지막_터치시간_표준편차'].tolist()
# 등분산성
from scipy.stats import levene
print(stats.levene(df31_1,df31_2,df31_3,df31_4,df31_5))
```

LeveneResult(statistic=1.3832273915456912, pvalue=0.23904798175097203)
유의확률이 0.239로 유의수준0.05하에서 유의하지않음

->등분산성 만족

-ANOVA분석

```

              df      sum_sq  mean_sq      F      PR(>F)
3(기분)      4.0      2.039152  0.509788  3.55261  0.007325
Residual    385.0    55.246249  0.143497      NaN      NaN
유의확률이 0.0073으로 유의수준 0.05하에서 유의함

```

->기분에 따른 왼쪽 마지막 터치시간 표준편차에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정(본페로니)

```

r31=f31.allpairtest(scipy.stats.ttest_ind, method='bonf')
r31[0]
#기분 2와 5간의 평균차이가 유의미함

```

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	0.2339	0.8153	1.0	False
1.0	3.0	-0.5055	0.6137	1.0	False
1.0	4.0	0.4317	0.6666	1.0	False
1.0	5.0	-2.7912	0.0059	0.0594	False
2.0	3.0	-0.8211	0.4125	1.0	False
2.0	4.0	0.3459	0.7299	1.0	False
2.0	5.0	-3.2448	0.0015	0.0147	True
3.0	4.0	0.8975	0.371	1.0	False
3.0	5.0	-2.7598	0.0065	0.0654	False
4.0	5.0	-2.3937	0.0196	0.1957	False

->기분2(좋음)와 기분5(아주 나쁨) 사이에 왼쪽 마지막 터치시간 표준편차에 차이가 있음

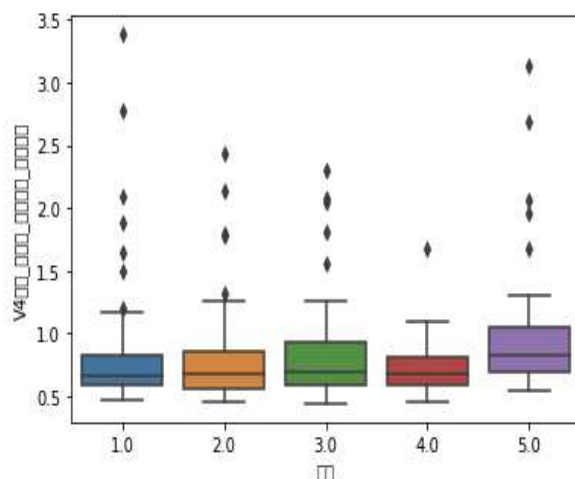
-시각화 (상자그림)

```

sns.boxplot(x='기분', y='V4왼쪽_마지막_터치시간_표준편차', data=df31)

```

<AxesSubplot: xlabel='기분', ylabel='V4왼쪽_마지막_터치시간_표준편차'>



상자그림 확인 시 특히 기분 5(나쁨)일 때 왼쪽 마지막 터치시간 표준편차가 증가하는 경향을 보임

->기분이 5(나쁨)일 때 게임에서 마지막 불필요한 왼쪽 터치가 길어진다? 라고 생각

4.11 기분에 따른 오른쪽 마지막 터치 시간 평균에 차이가 있는지 분석

-등분산검정

```
df32=pd.DataFrame(a1111,columns=['기분','V4오른쪽_마지막_터치시간_평균'])
df32['V4오른쪽_마지막_터치시간_평균'] = pd.to_numeric(df32['V4오른쪽_마지막_터치시간_평균'], errors='coerce')

df32_1=df32[df32.기분==1]['V4오른쪽_마지막_터치시간_평균'].tolist()
df32_2=df32[df32.기분==2]['V4오른쪽_마지막_터치시간_평균'].tolist()
df32_3=df32[df32.기분==3]['V4오른쪽_마지막_터치시간_평균'].tolist()
df32_4=df32[df32.기분==4]['V4오른쪽_마지막_터치시간_평균'].tolist()
df32_5=df32[df32.기분==5]['V4오른쪽_마지막_터치시간_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df32_1,df32_2,df32_3,df32_4,df32_5))
```

LeveneResult(statistic=2.171440113303925, pvalue=0.07157944140305326)

유의확률이 0.0716으로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

```
df      sum_sq  mean_sq      F    PR(>F)
C(기분)    4.0    2.936648  0.734162  5.463759  0.000275
Residual 385.0  51.732217  0.134369      NaN      NaN
유의확률이 0.0003으로 유의수준 0.05하에서 유의함
```

->기분에 따른 오른쪽 마지막 터치시간 평균에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정(본페로니)

```
r32=f32.allpairtest(scipy.stats.ttest_ind, method='bonf')
r32[0]
#기분 1과5, 2와5, 3과5, 4와5 간의 평균차이가 유의미함
```

Test Multiple Comparison ttest_ind FWER=0.05
method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-0.3564	0.7219	1.0	False
1.0	3.0	-1.3616	0.1747	1.0	False
1.0	4.0	0.1069	0.915	1.0	False
1.0	5.0	-4.2197	0.0	0.0004	True
2.0	3.0	-1.0853	0.2791	1.0	False
2.0	4.0	0.3842	0.7015	1.0	False
2.0	5.0	-4.4068	0.0	0.0002	True
3.0	4.0	0.9861	0.3258	1.0	False
3.0	5.0	-2.983	0.0034	0.0335	True
4.0	5.0	-3.5443	0.0007	0.0074	True

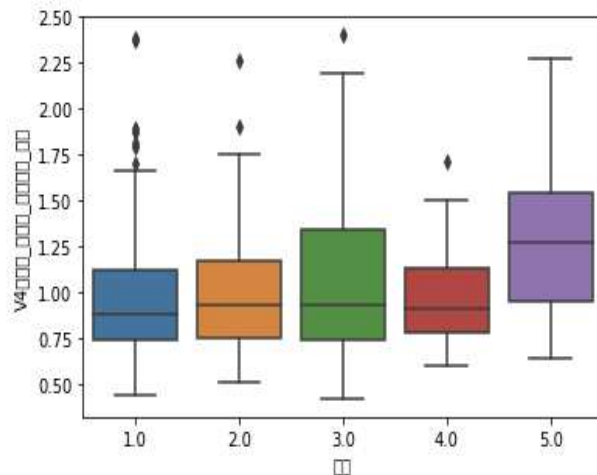
->기분1(아주 좋음)과 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음

- >기분2(좋음)와 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음
- >기분3(보통)과 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음
- >기분4(나쁨)과 기분5(아주 나쁨) 사이에 마지막 터치시간 평균에 차이가 있음

-시각화 (상자그림)

```
sns.boxplot(x='기분', y='V4오른쪽_마지막_터치시간_평균', data=df32)
```

```
<AxesSubplot: xlabel='기분', ylabel='V4오른쪽_마지막_터치시간_평균'>
```



상자그림 확인 시 특히 기분 5(나쁨)일 때 오른쪽 마지막 터치시간 평균이 증가하는 경향을 보임

->기분이 5(나쁨)일 때 게임에서 마지막 불필요한 오른쪽 터치가 길어진다고 생각

4.12 기분에 따른 오른쪽 마지막 터치 시간 표준편차에 차이가 있는지 분석

-등분산검정

```
df33=pd.DataFrame(a1111,columns=['기분','V4오른쪽_마지막_터치시간_표준편차'])
df33['V4오른쪽_마지막_터치시간_표준편차'] = pd.to_numeric(df33['V4오른쪽_마지막_터치시간_표준편차'], errors='coerce')
```

```
df33_1=df33[df33.기분==1]['V4오른쪽_마지막_터치시간_표준편차'].tolist()
df33_2=df33[df33.기분==2]['V4오른쪽_마지막_터치시간_표준편차'].tolist()
df33_3=df33[df33.기분==3]['V4오른쪽_마지막_터치시간_표준편차'].tolist()
df33_4=df33[df33.기분==4]['V4오른쪽_마지막_터치시간_표준편차'].tolist()
df33_5=df33[df33.기분==5]['V4오른쪽_마지막_터치시간_표준편차'].tolist()
```

등분산성

```
from scipy.stats import levene
print(stats.levene(df33_1,df33_2,df33_3,df33_4,df33_5))
```

```
LeveneResult(statistic=0.9465185391202132, pvalue=0.4369767492916845)
```

유의확률이 0.437로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

```

df      sum_sq  mean_sq      F      PR(>F)
C(기분)    4.0    1.529336  0.382334  2.507078  0.041701
Residual 385.0    58.713195  0.152502      NaN      NaN
유의확률이 0.0417으로 유의수준 0.05하에서 유의함

```

->기분에 따른 오른쪽 마지막 터치시간 표준편차에 차이가 있다-> 차이가 있으므로 사후검정 시행 O

-사후검정 (본페로니)

```

r33=f33.allpairtest(scipy.stats.ttest_ind, method='bonf')
r33[0]

```

Test Multiple Comparison ttest_ind FWER=0.05

method=bonf alphacSidak=0.01, alphacBonf=0.005

group1	group2	stat	pval	pval_corr	reject
1.0	2.0	-0.4443	0.6573	1.0	False
1.0	3.0	-0.6162	0.5384	1.0	False
1.0	4.0	0.3552	0.723	1.0	False
1.0	5.0	-2.6544	0.0088	0.0881	False
2.0	3.0	-0.2253	0.8219	1.0	False
2.0	4.0	0.7902	0.4309	1.0	False
2.0	5.0	-2.716	0.0075	0.0745	False
3.0	4.0	0.785	0.4338	1.0	False
3.0	5.0	-2.2344	0.027	0.27	False
4.0	5.0	-2.4477	0.0171	0.1709	False

->오른쪽 마지막 터치시간 표준편차에 차이가 있는 그룹이 없음-> 유의한 차이가 없다

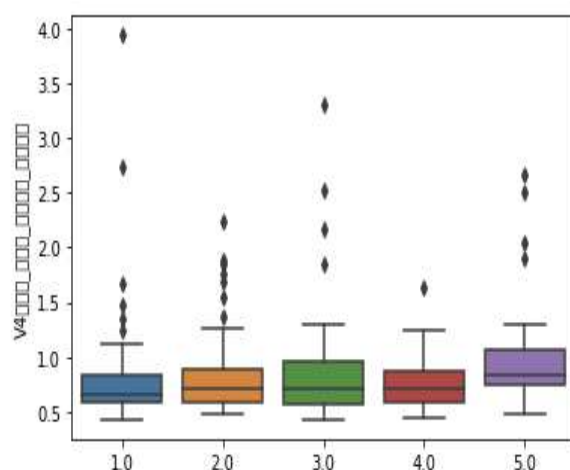
-시각화 (상자그림)

```

sns.boxplot(x='기분',y='V4오른쪽_마지막_터치시간_표준편차',data=df33)

```

<AxesSubplot: xlabel='기분', ylabel='V4오른쪽_마지막_터치시간_표준편차'>



상자그림 확인 시 큰 차이가 없지만 기분5(아주 나쁨)에서 약간의 증가하는 경향이 있음
-> 기분이 나쁠수록 게임 수행 시 오른쪽 화면에 마지막 터치시간이 길어진다고 유추

4.13 기분에 따른 터치 횟수에 차이가 있는지 분석

-등분산검정

```
df34=pd.DataFrame(a1111,columns=['기분','V4터치_횟수'])
df34['V4터치_횟수'] = pd.to_numeric(df34['V4터치_횟수'], errors='coerce')

df34_1=df34[df34.기분==1]['V4터치_횟수'].tolist()
df34_2=df34[df34.기분==2]['V4터치_횟수'].tolist()
df34_3=df34[df34.기분==3]['V4터치_횟수'].tolist()
df34_4=df34[df34.기분==4]['V4터치_횟수'].tolist()
df34_5=df34[df34.기분==5]['V4터치_횟수'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df34_1,df34_2,df34_3,df34_4,df34_5))
```

LeveneResult(statistic=2.3381286465817674, pvalue=0.054814440877718744)
유의확률이 0.0548으로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	5.196639e+03	1299.159759	0.183579	0.946877
Residual	385.0	2.724582e+06	7076.836495	NaN	NaN

유의확률이 0.9469으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 터치횟수에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

4.14 기분에 따른 왼쪽 터치 횟수에 차이가 있는지 분석

-등분산검정

```
df35=pd.DataFrame(a1111,columns=['기분','V4왼쪽_터치_횟수'])
df35['V4왼쪽_터치_횟수'] = pd.to_numeric(df35['V4왼쪽_터치_횟수'], errors='coerce')

df35_1=df35[df35.기분==1]['V4왼쪽_터치_횟수'].tolist()
df35_2=df35[df35.기분==2]['V4왼쪽_터치_횟수'].tolist()
df35_3=df35[df35.기분==3]['V4왼쪽_터치_횟수'].tolist()
df35_4=df35[df35.기분==4]['V4왼쪽_터치_횟수'].tolist()
df35_5=df35[df35.기분==5]['V4왼쪽_터치_횟수'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df35_1,df35_2,df35_3,df35_4,df35_5))
```

LeveneResult(statistic=1.919482967273336, pvalue=0.10644569592304963)
유의확률이 0.1064로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1064.755257	266.188814	0.145254	0.965034
Residual	385.0	705541.575512	1832.575521	NaN	NaN

유의확률이 0.965으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 왼쪽 터치횟수에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

4.15 기분에 따른 오른쪽 터치 횟수에 차이가 있는지 분석

-등분산검정

```
df36=pd.DataFrame(a1111,columns=['기분','V4오른쪽_터치_횟수'])
df36['V4오른쪽_터치_횟수'] = pd.to_numeric(df36['V4오른쪽_터치_횟수'], errors='coerce')

df36_1=df36[df36.기분==1]['V4오른쪽_터치_횟수'].tolist()
df36_2=df36[df36.기분==2]['V4오른쪽_터치_횟수'].tolist()
df36_3=df36[df36.기분==3]['V4오른쪽_터치_횟수'].tolist()
df36_4=df36[df36.기분==4]['V4오른쪽_터치_횟수'].tolist()
df36_5=df36[df36.기분==5]['V4오른쪽_터치_횟수'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df36_1,df36_2,df36_3,df36_4,df36_5))
```

LeveneResult(statistic=1.978211959178386, pvalue=0.09711722519106494)

유의확률이 0.0971로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1925.724751	481.431188	0.256569	0.905587
Residual	385.0	722422.418838	1876.421867	NaN	NaN

유의확률이 0.9056으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 오른쪽 터치횟수에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

5. Visual_5 게임

5.1 기분에 따른 정답률에 차이가 있는지 분석

-등분산검정

```
df37=pd.DataFrame(a1111,columns=['기분','V5정답률'])
df37['V5정답률'] = pd.to_numeric(df37['V5정답률'], errors='coerce')

df37_1=df37[df37.기분==1]['V5정답률'].tolist()
df37_2=df37[df37.기분==2]['V5정답률'].tolist()
df37_3=df37[df37.기분==3]['V5정답률'].tolist()
df37_4=df37[df37.기분==4]['V5정답률'].tolist()
df37_5=df37[df37.기분==5]['V5정답률'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df37_1,df37_2,df37_3,df37_4,df37_5))
```

LeveneResult(statistic=1.816004439396628, pvalue=0.12495440711318512)

유의확률이 0.125로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	397.321720	99.330430	1.351616	0.250228
Residual	385.0	28293.688812	73.490101	NaN	NaN

유의확률이 0.2502으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 정답률에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

5.2 기분에 따른 난이도 평균에 차이가 있는지 분석

-등분산검정

```
df38=pd.DataFrame(a1111,columns=['기분','V5난이도_평균'])
df38['V5난이도_평균'] = pd.to_numeric(df38['V5난이도_평균'], errors='coerce')

df38_1=df38[df38.기분==1]['V5난이도_평균'].tolist()
df38_2=df38[df38.기분==2]['V5난이도_평균'].tolist()
df38_3=df38[df38.기분==3]['V5난이도_평균'].tolist()
df38_4=df38[df38.기분==4]['V5난이도_평균'].tolist()
df38_5=df38[df38.기분==5]['V5난이도_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df38_1,df38_2,df38_3,df38_4,df38_5))
```

LeveneResult(statistic=0.6147704204806956, pvalue=0.6522408823371351)

유의확률이 0.652로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	34.234143	8.558536	0.892607	0.468336
Residual	385.0	3691.474465	9.588245	NaN	NaN

유의확률이 0.4683으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 난이도 평균에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

5.3 기분에 따른 난이도 표준편차에 차이가 있는지 분석

-등분산검정

```
df39=pd.DataFrame(a1111,columns=['기분','V5난이도_표준편차'])
df39['V5난이도_표준편차'] = pd.to_numeric(df39['V5난이도_표준편차'], errors='coerce')

df39_1=df39[df39.기분==1]['V5난이도_표준편차'].tolist()
df39_2=df39[df39.기분==2]['V5난이도_표준편차'].tolist()
df39_3=df39[df39.기분==3]['V5난이도_표준편차'].tolist()
df39_4=df39[df39.기분==4]['V5난이도_표준편차'].tolist()
df39_5=df39[df39.기분==5]['V5난이도_표준편차'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df39_1,df39_2,df39_3,df39_4,df39_5))
```

LeveneResult(statistic=0.619787443270211, pvalue=0.6486497268480615)

유의확률이 0.6486로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	8.842675	2.210669	0.918405	0.453137
Residual	385.0	926.723100	2.407073	NaN	NaN

유의확률이 0.4531으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 난이도 표준편차에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

5.4 기분에 따른 난이도 최대에 차이가 있는지 분석

-등분산검정

```
df40=pd.DataFrame(a1111,columns=['기분','V5난이도_최대'])
df40['V5난이도_최대'] = pd.to_numeric(df40['V5난이도_최대'], errors='coerce')

df40_1=df40[df40.기분==1]['V5난이도_최대'].tolist()
df40_2=df40[df40.기분==2]['V5난이도_최대'].tolist()
df40_3=df40[df40.기분==3]['V5난이도_최대'].tolist()
df40_4=df40[df40.기분==4]['V5난이도_최대'].tolist()
df40_5=df40[df40.기분==5]['V5난이도_최대'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df40_1,df40_2,df40_3,df40_4,df40_5))
```

LeveneResult(statistic=0.5385883058623308, pvalue=0.7074740827459338)

유의확률이 0.7075로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	61.673744	15.418436	0.764876	0.548612

유의확률이 0.5486으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 난이도 최대에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

5.5 기분에 따른 결정시간 평균에 차이가 있는지 분석

-등분산검정

```
df41=pd.DataFrame(a1111,columns=['기분','V5결정시간_평균'])
df41['V5결정시간_평균'] = pd.to_numeric(df41['V5결정시간_평균'], errors='coerce')

df41_1=df41[df41.기분==1]['V5결정시간_평균'].tolist()
df41_2=df41[df41.기분==2]['V5결정시간_평균'].tolist()
df41_3=df41[df41.기분==3]['V5결정시간_평균'].tolist()
df41_4=df41[df41.기분==4]['V5결정시간_평균'].tolist()
df41_5=df41[df41.기분==5]['V5결정시간_평균'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df41_1,df41_2,df41_3,df41_4,df41_5))
```

LeveneResult(statistic=0.7654080663621343, pvalue=0.5482613388247481)

유의확률이 0.5483로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	0.687687	0.171922	0.726406	0.57432
Residual	385.0	91.119737	0.236675	NaN	NaN

유의확률이 0.5743으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 결정시간 평균에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

5.6 기분에 따른 결정시간 표준편차에 차이가 있는지 분석

-등분산검정

```
df42=pd.DataFrame(a1111,columns=['기분','V5결정시간_표준편차'])

df42['V5결정시간_표준편차'] = pd.to_numeric(df42['V5결정시간_표준편차'], errors='coerce')

df42_1=df42[df42.기분==1]['V5결정시간_표준편차'].tolist()
df42_2=df42[df42.기분==2]['V5결정시간_표준편차'].tolist()
df42_3=df42[df42.기분==3]['V5결정시간_표준편차'].tolist()
df42_4=df42[df42.기분==4]['V5결정시간_표준편차'].tolist()
df42_5=df42[df42.기분==5]['V5결정시간_표준편차'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df42_1,df42_2,df42_3,df42_4,df42_5))
```

LeveneResult(statistic=0.34538788827534295, pvalue=0.8472039941546973)

유의확률이 0.8472로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	0.446429	0.111607	0.692739	0.597336
Residual	385.0	62.027443	0.161110	NaN	NaN

유의확률이 0.5973으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 결정시간 표준편차에 차이가 없다-> 차이가 없으므로 사후검정 시행 X

5.7 기분에 따른 터치 횟수에 차이가 있는지 분석

-등분산검정

```
df43=pd.DataFrame(a1111,columns=['기분','V5터치_횟수'])
df43['V5터치_횟수'] = pd.to_numeric(df43['V5터치_횟수'], errors='coerce')

df43_1=df43[df43.기분==1]['V5터치_횟수'].tolist()
df43_2=df43[df43.기분==2]['V5터치_횟수'].tolist()
df43_3=df43[df43.기분==3]['V5터치_횟수'].tolist()
df43_4=df43[df43.기분==4]['V5터치_횟수'].tolist()
df43_5=df43[df43.기분==5]['V5터치_횟수'].tolist()

# 등분산성
from scipy.stats import levene
print(stats.levene(df43_1,df43_2,df43_3,df43_4,df43_5))
```

LeveneResult(statistic=0.7016752616220077, pvalue=0.5911821593623054)

유의확률이 0.5912로 유의수준0.05하에서 유의하지 않음

->등분산성 만족

-ANOVA분석

	df	sum_sq	mean_sq	F	PR(>F)
C(기분)	4.0	1.359030e+04	3397.574097	0.375925	0.825815
Residual	385.0	3.479594e+06	9037.905917	NaN	NaN

유의확률이 0.8258으로 유의수준 0.05하에서 유의하지 않음

->기분에 따른 터치횟수에 차이가 없다-> 차이가 없으므로 사후검정 시행 X