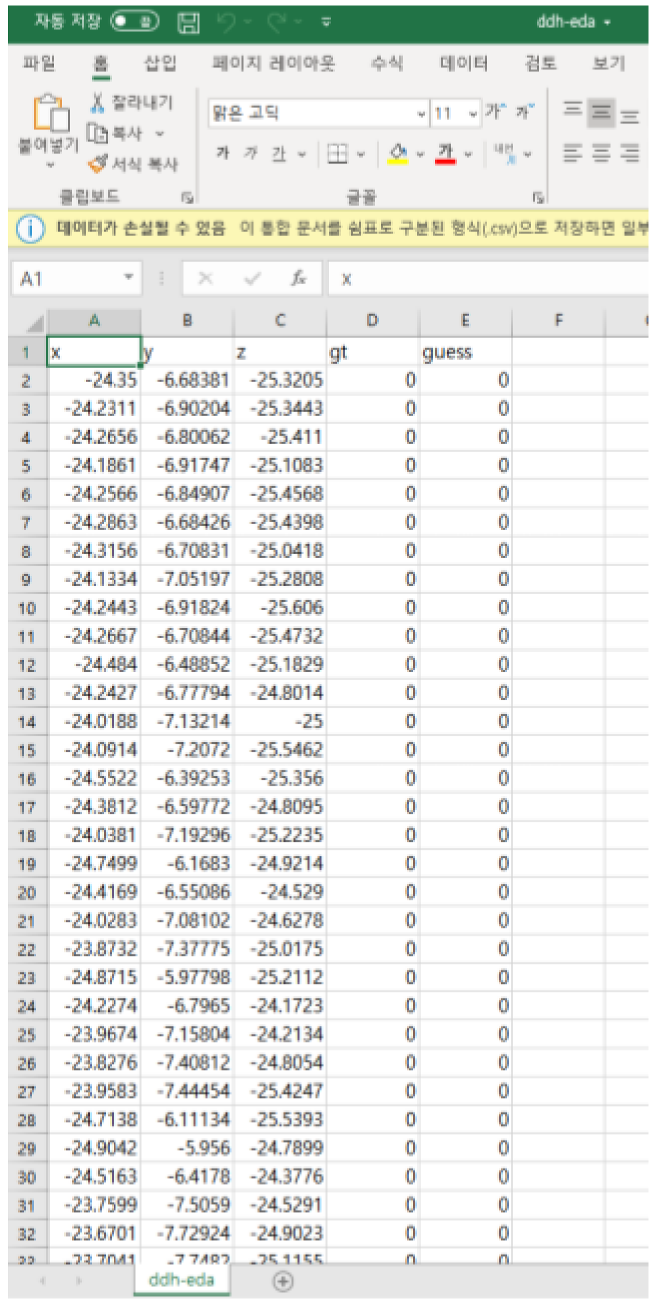
ddh & daeyou EDA

끝이나지 않는 탐색전

2021.01.27

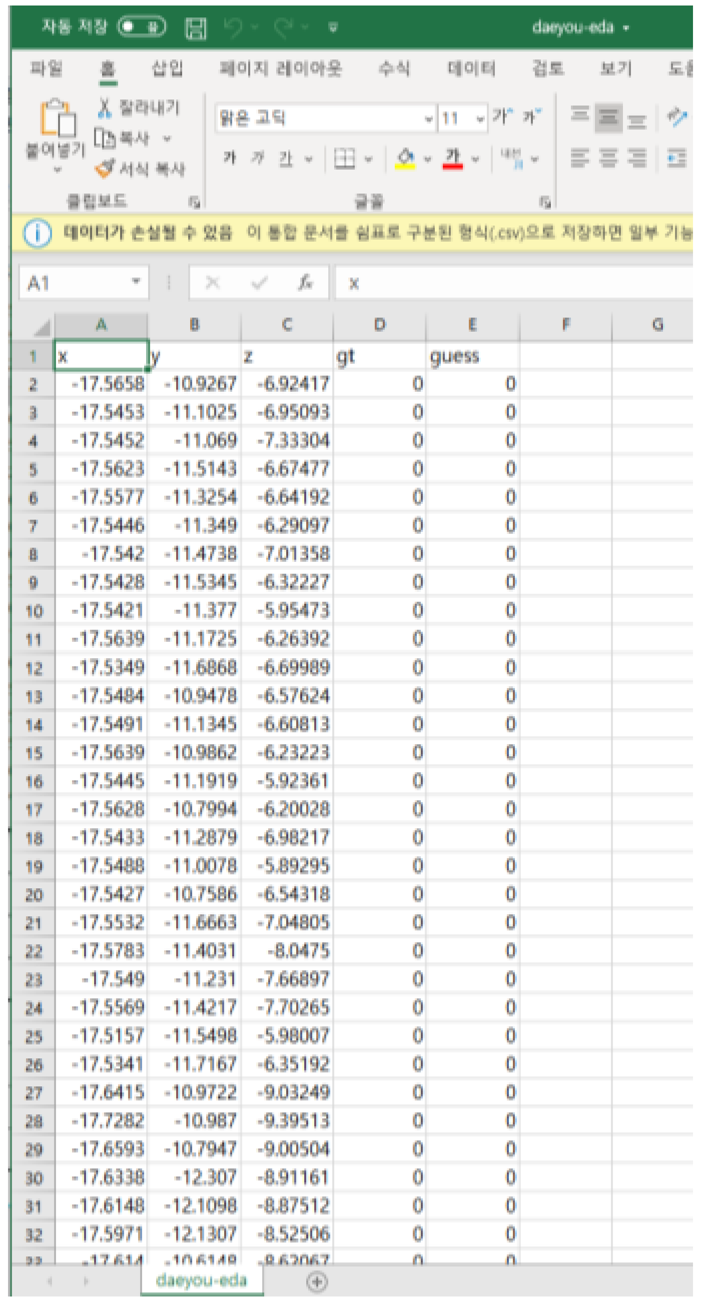
목차

과제내용



데이터가 손실될 수 있음 이 통합 문서를 쉼표로 구분된 형식(.csv)으로 저장하면 일부

	A	B	C	D	E
1	x	y	z	gt	guess
2	-24.35	-6.68381	-25.3205	0	0
3	-24.2311	-6.90204	-25.3443	0	0
4	-24.2656	-6.80062	-25.411	0	0
5	-24.1861	-6.91747	-25.1083	0	0
6	-24.2566	-6.84907	-25.4568	0	0
7	-24.2863	-6.68426	-25.4398	0	0
8	-24.3156	-6.70831	-25.0418	0	0
9	-24.1334	-7.05197	-25.2808	0	0
10	-24.2443	-6.91824	-25.606	0	0
11	-24.2667	-6.70844	-25.4732	0	0
12	-24.484	-6.48852	-25.1829	0	0
13	-24.2427	-6.77794	-24.8014	0	0
14	-24.0188	-7.13214	-25	0	0
15	-24.0914	-7.2072	-25.5462	0	0
16	-24.5522	-6.39253	-25.356	0	0
17	-24.3812	-6.59772	-24.8095	0	0
18	-24.0381	-7.19296	-25.2235	0	0
19	-24.7499	-6.1683	-24.9214	0	0
20	-24.4169	-6.55086	-24.529	0	0
21	-24.0283	-7.08102	-24.6278	0	0
22	-23.8732	-7.37775	-25.0175	0	0
23	-24.8715	-5.97798	-25.2112	0	0
24	-24.2274	-6.7965	-24.1723	0	0
25	-23.9674	-7.15804	-24.2134	0	0
26	-23.8276	-7.40812	-24.8054	0	0
27	-23.9583	-7.44454	-25.4247	0	0
28	-24.7138	-6.11134	-25.5393	0	0
29	-24.9042	-5.956	-24.7899	0	0
30	-24.5163	-6.4178	-24.3776	0	0
31	-23.7599	-7.5059	-24.5291	0	0
32	-23.6701	-7.72924	-24.9023	0	0
33	-23.7041	-7.7482	-25.1155	0	0



데이터가 손실될 수 있음 이 통합 문서를 쉼표로 구분된 형식(.csv)으로 저장하면 일부 기능

	A	B	C	D	E
1	x	y	z	gt	guess
2	-17.5658	-10.9267	-6.92417	0	0
3	-17.5453	-11.1025	-6.95093	0	0
4	-17.5452	-11.069	-7.33304	0	0
5	-17.5623	-11.5143	-6.67477	0	0
6	-17.5577	-11.3254	-6.64192	0	0
7	-17.5446	-11.349	-6.29097	0	0
8	-17.542	-11.4738	-7.01358	0	0
9	-17.5428	-11.5345	-6.32227	0	0
10	-17.5421	-11.377	-5.95473	0	0
11	-17.5639	-11.1725	-6.26392	0	0
12	-17.5349	-11.6868	-6.69989	0	0
13	-17.5484	-10.9478	-6.57624	0	0
14	-17.5491	-11.1345	-6.60813	0	0
15	-17.5639	-10.9862	-6.23223	0	0
16	-17.5445	-11.1919	-5.92361	0	0
17	-17.5628	-10.7994	-6.20028	0	0
18	-17.5433	-11.2879	-6.98217	0	0
19	-17.5488	-11.0078	-5.89295	0	0
20	-17.5427	-10.7586	-6.54318	0	0
21	-17.5532	-11.6663	-7.04805	0	0
22	-17.5783	-11.4031	-8.0475	0	0
23	-17.549	-11.231	-7.66897	0	0
24	-17.5569	-11.4217	-7.70265	0	0
25	-17.5157	-11.5498	-5.98007	0	0
26	-17.5341	-11.7167	-6.35192	0	0
27	-17.6415	-10.9722	-9.03249	0	0
28	-17.7282	-10.987	-9.39513	0	0
29	-17.6593	-10.7947	-9.00504	0	0
30	-17.6338	-12.307	-8.91161	0	0
31	-17.6148	-12.1098	-8.87512	0	0
32	-17.5971	-12.1307	-8.52506	0	0
33	-17.614	-10.6149	-8.62067	0	0

gt : 실질 치아 포인터 값

guess : 인공지능이 인식한 치아 포인터 값

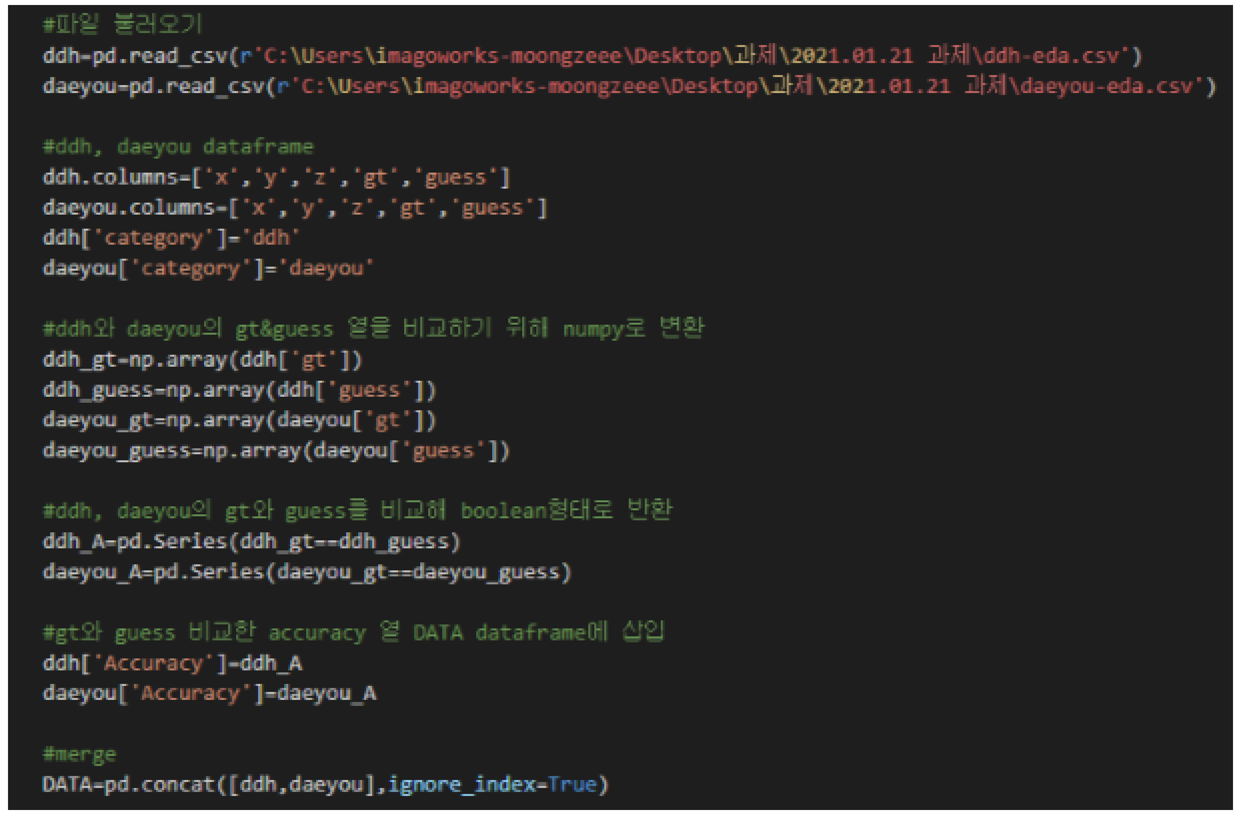
x,y,z : 치아 포인터별 좌표

탐색이라고 하고 헤엄

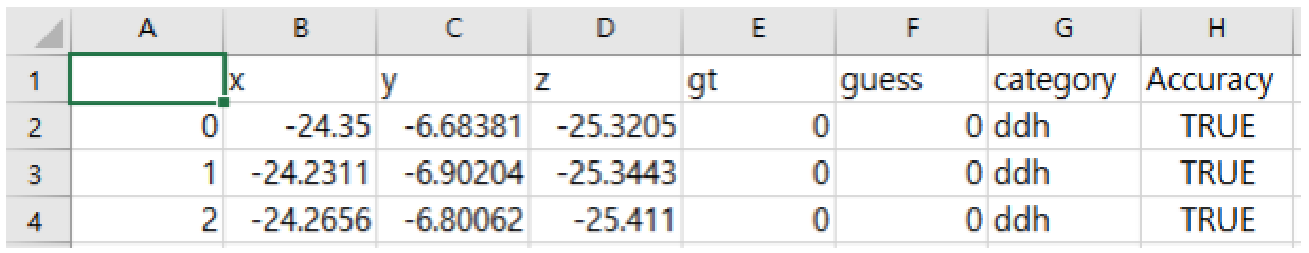
- **데이터 전처리**
  - 데이터 merge
  - 데이터 타입 분류
- **visualization**
  - Boxen Plot
  - 2D-Scatter Plot
  - KDE Plot
- **χ^2 - test**
  - 범주형 data 상관관계분석
- **t-test**
  - 범주형 data -연속형 data 차이분석

PROJECT 1

data merge



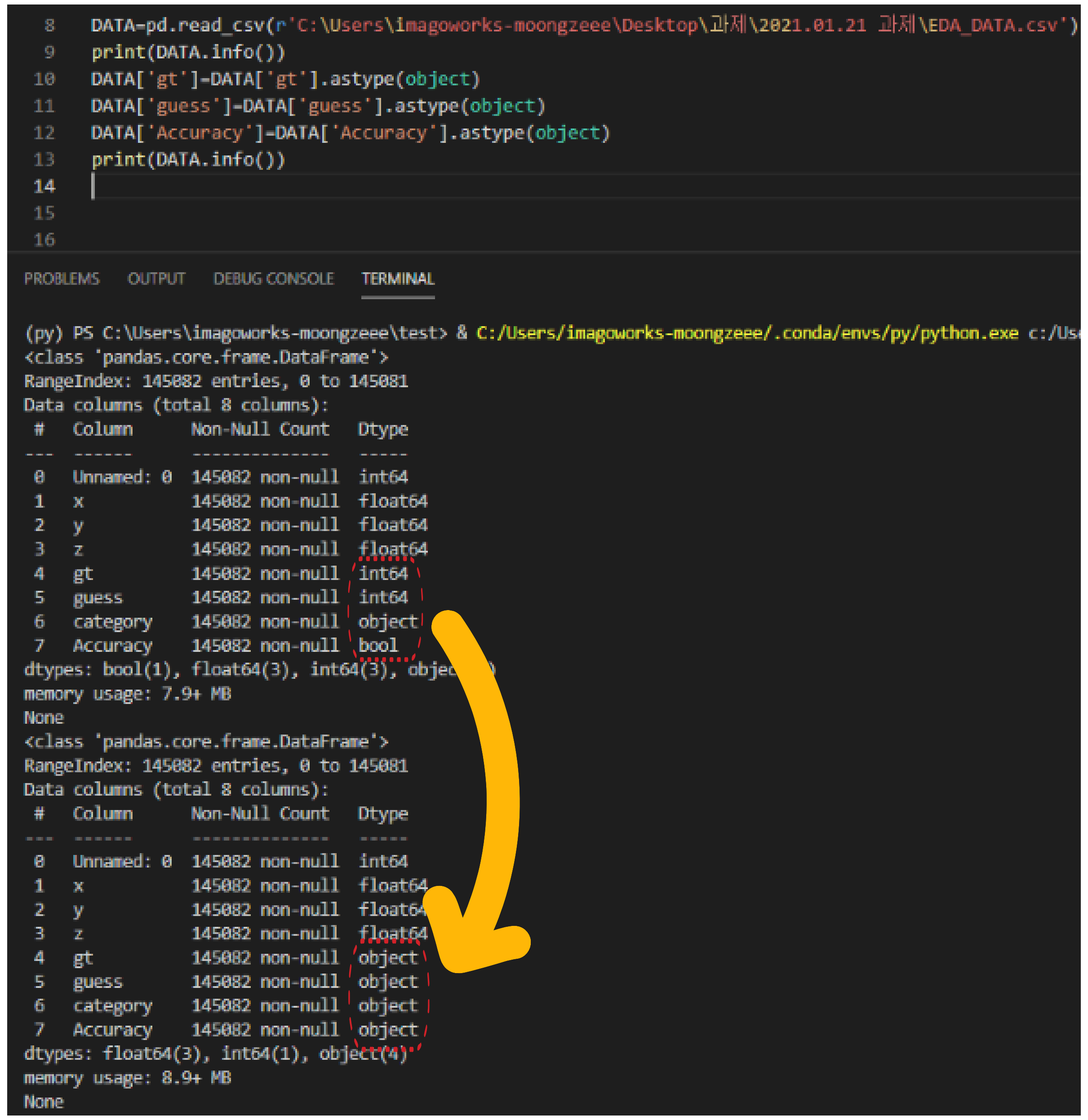
```
#파일 불러오기
ddh=pd.read_csv(r'C:\Users\imagoworks-moongzeee\Desktop\과제\2021.01.21 과제\ddh-eda.csv')
daeyou=pd.read_csv(r'C:\Users\imagoworks-moongzeee\Desktop\과제\2021.01.21 과제\daeyou-eda.csv')

#ddh, daeyou dataframe
ddh.columns=['x','y','z','gt','guess']
daeyou.columns=['x','y','z','gt','guess']
ddh['category']='ddh'
daeyou['category']='daeyou'

#ddh와 daeyou의 gt&guess 열을 비교하기 위해 numpy로 변환
ddh_gt=np.array(ddh['gt'])
ddh_guess=np.array(ddh['guess'])
daeyou_gt=np.array(daeyou['gt'])
daeyou_guess=np.array(daeyou['guess'])

#ddh, daeyou의 gt와 guess를 비교해 boolean형태로 반환
ddh_A=pd.Series(ddh_gt==ddh_guess)
daeyou_A=pd.Series(daeyou_gt==daeyou_guess)

#gt와 guess 비교한 accuracy 열 DATA dataframe에 삽입
ddh['Accuracy']=ddh_A
daeyou['Accuracy']=daeyou_A

#merge
DATA=pd.concat([ddh,daeyou],ignore_index=True)
```



	A	B	C	D	E	F	G	H
1		x	y	z	gt	guess	category	Accuracy
2	0	-24.35	-6.68381	-25.3205	0	0	ddh	TRUE
3	1	-24.2311	-6.90204	-25.3443	0	0	ddh	TRUE
4	2	-24.2656	-6.80062	-25.411	0	0	ddh	TRUE

PROJECT 1

data type change



```
DATA=pd.read_csv(r'C:\Users\imagoworks-moongzeee\Desktop\과제\2021.01.21 과제\EDA_DATA.csv')
print(DATA.info())
DATA['gt']=DATA['gt'].astype(object)
DATA['guess']=DATA['guess'].astype(object)
DATA['Accuracy']=DATA['Accuracy'].astype(object)
print(DATA.info())
```

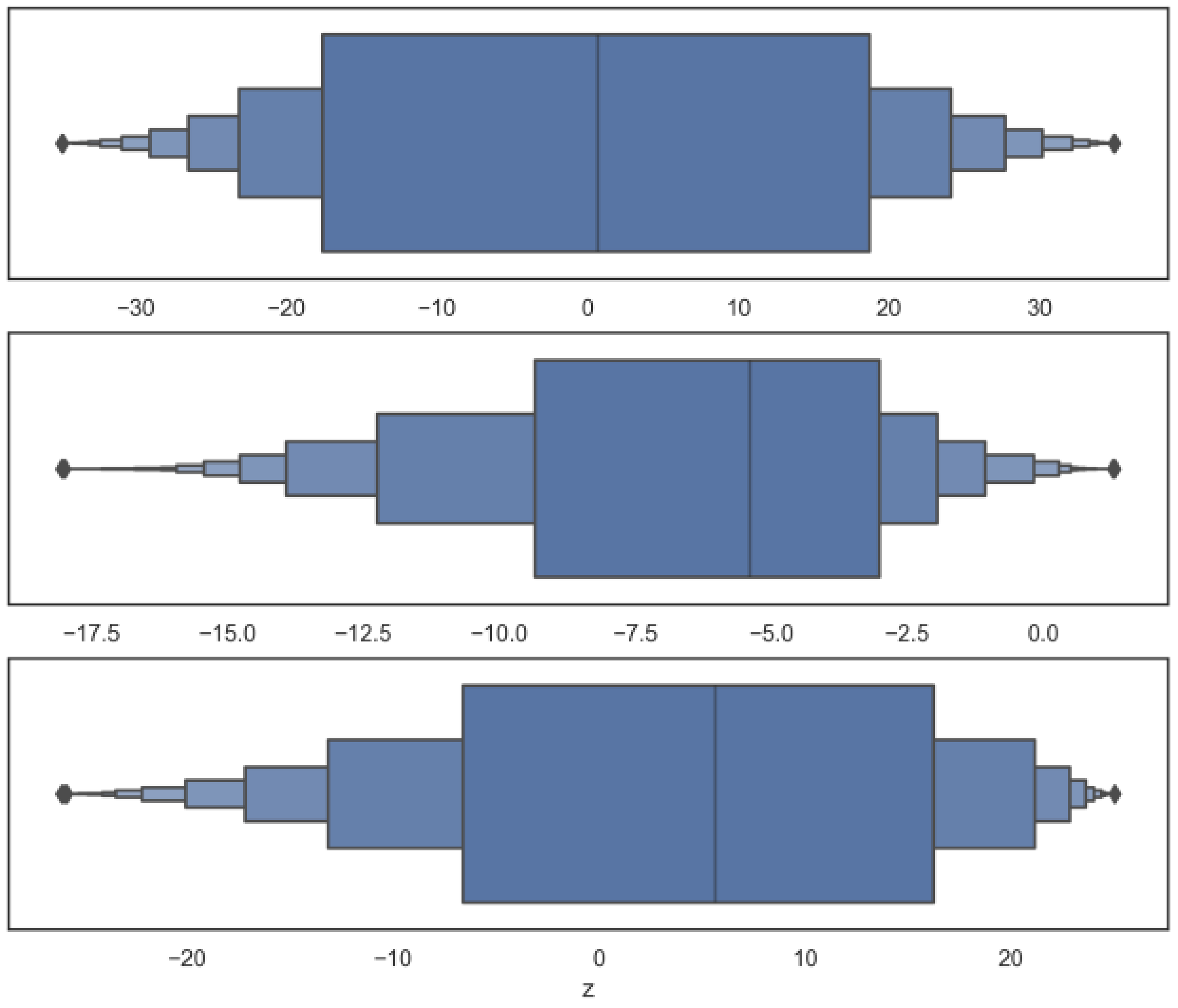
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
(py) PS C:\Users\imagoworks-moongzeee\test> & C:/Users/imagoworks-moongzeee/.conda/envs/py/python.exe c:/Us
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145082 entries, 0 to 145081
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   Unnamed: 0   145082 non-null  int64
 1   x            145082 non-null  float64
 2   y            145082 non-null  float64
 3   z            145082 non-null  float64
 4   gt           145082 non-null  int64
 5   guess        145082 non-null  int64
 6   category     145082 non-null  object
 7   Accuracy     145082 non-null  bool
dtypes: bool(1), float64(3), int64(3), objec
memory usage: 7.9+ MB
None
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145082 entries, 0 to 145081
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   Unnamed: 0   145082 non-null  int64
 1   x            145082 non-null  float64
 2   y            145082 non-null  float64
 3   z            145082 non-null  float64
 4   gt           145082 non-null  object
 5   guess        145082 non-null  object
 6   category     145082 non-null  object
 7   Accuracy     145082 non-null  object
dtypes: float64(3), int64(1), object(4)
memory usage: 8.9+ MB
None
```

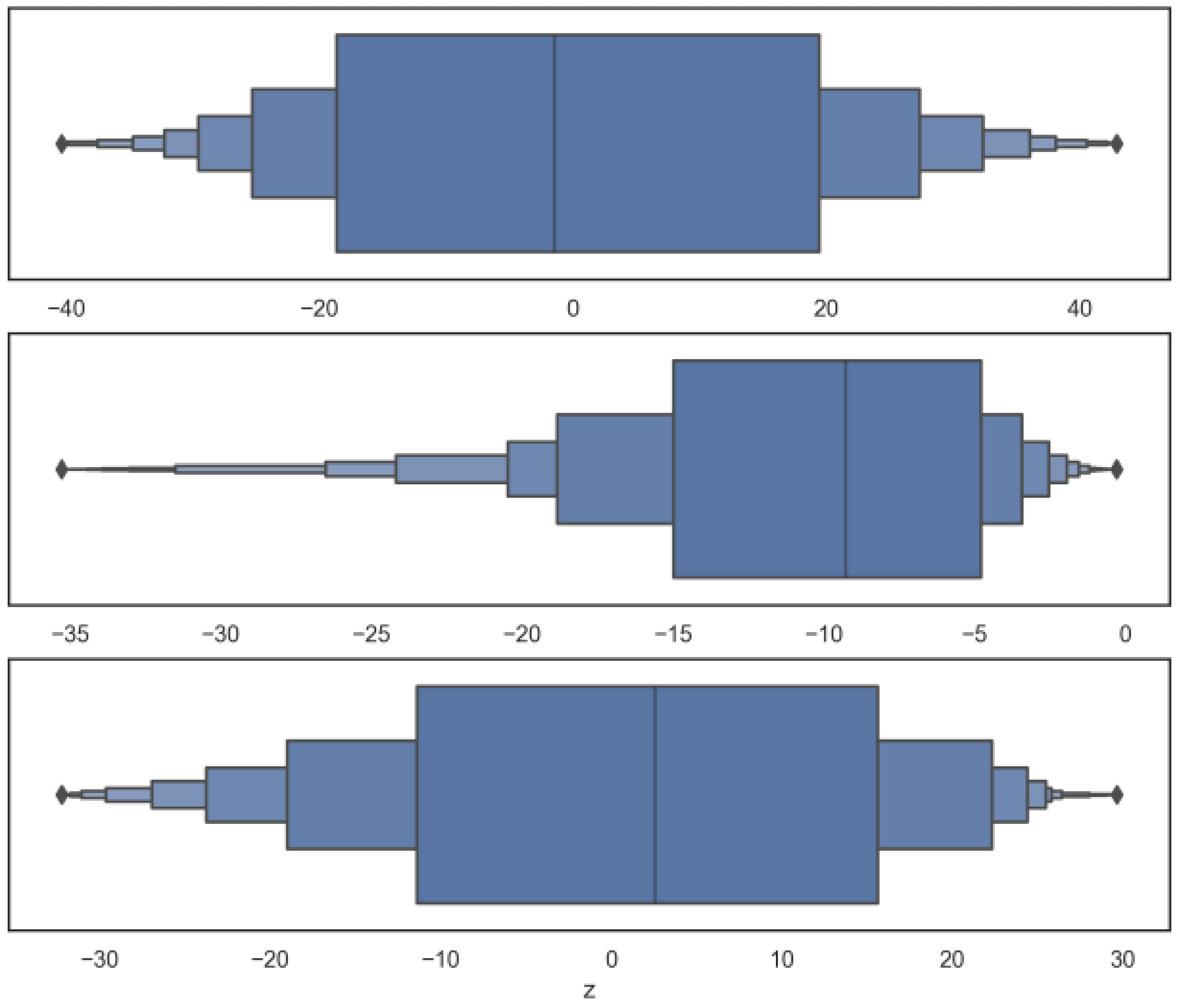
PROJECT 2

Boxen Plot

ddh



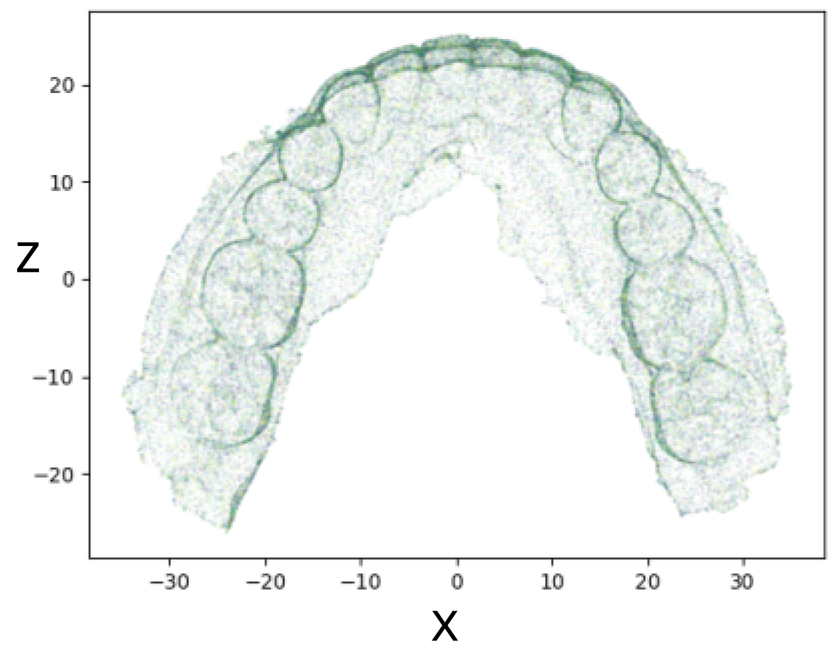
daeyou

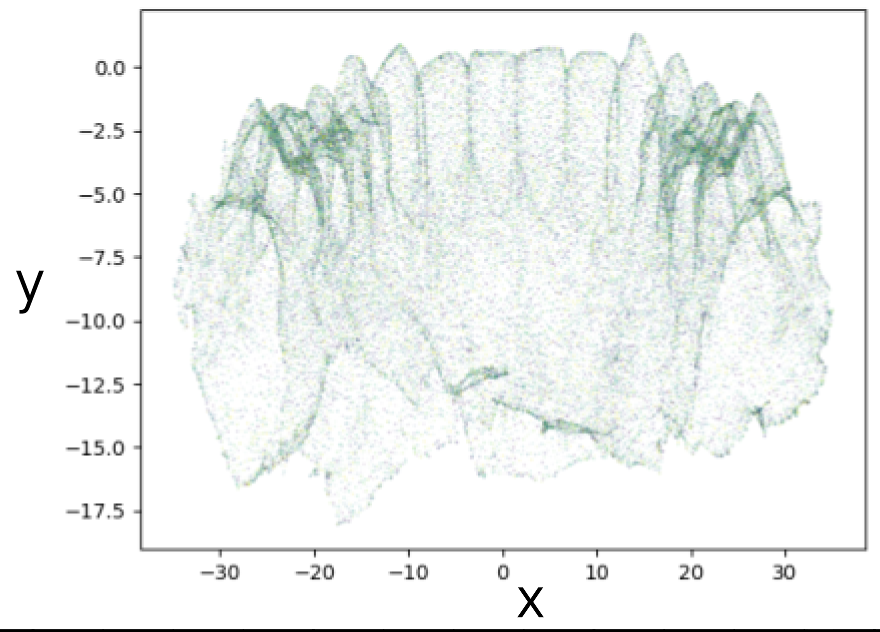


PROJECT 2

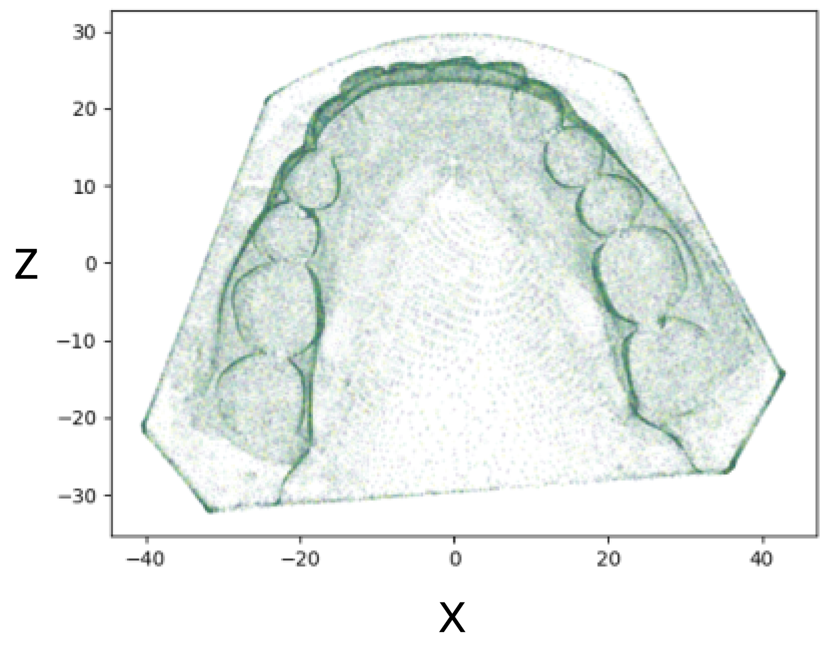
2D-Scatter Plot

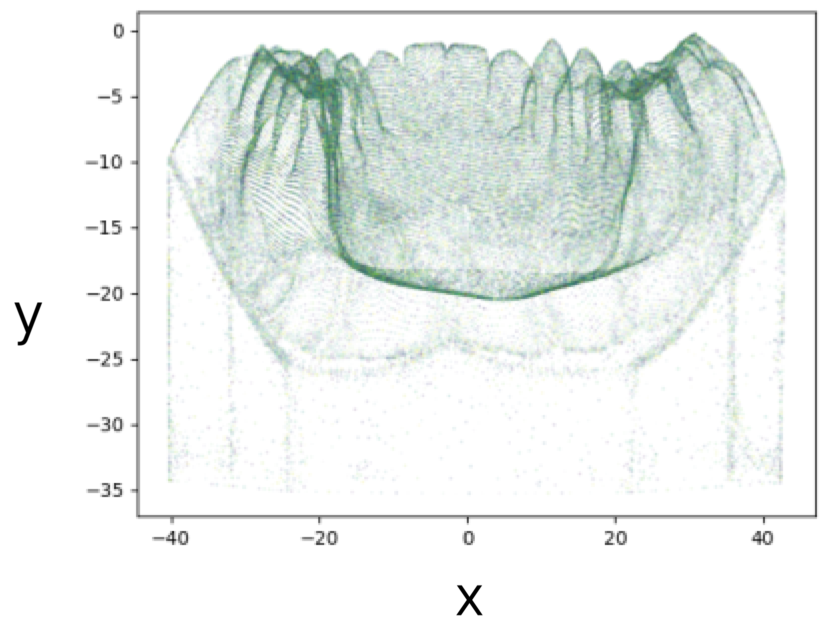
ddh





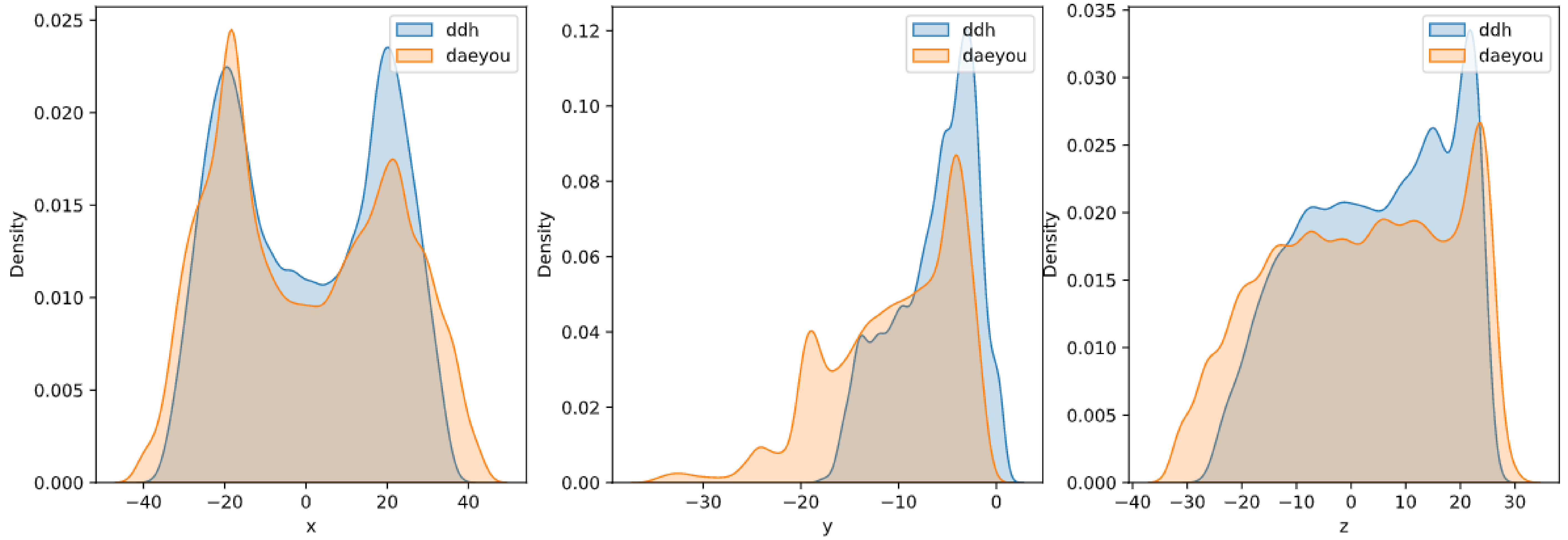
daeyou





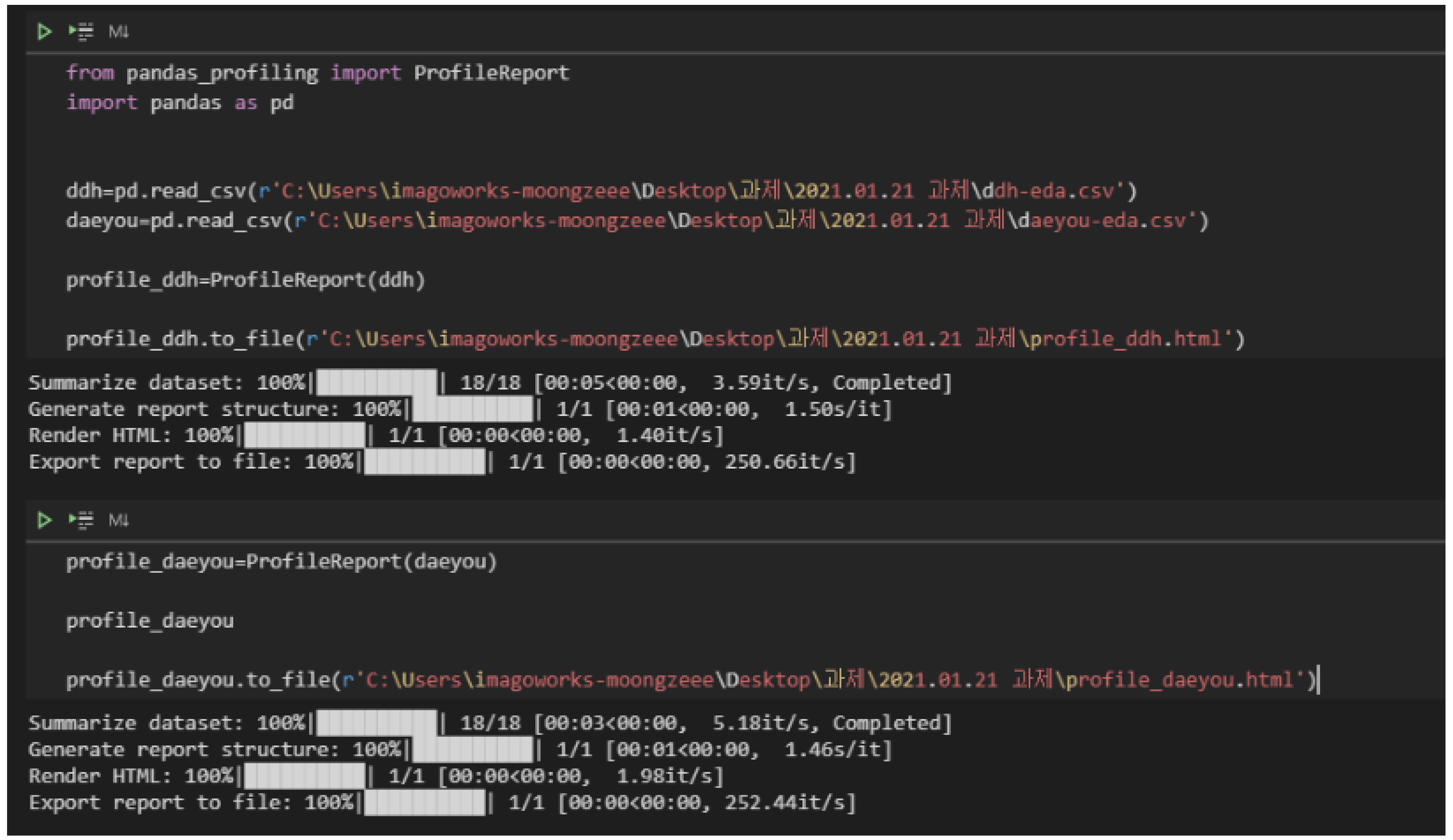
PROJECT 2

KDE Plot



기술통계분석

pandas profilereport module



```
from pandas_profiling import ProfileReport
import pandas as pd


ddh=pd.read_csv(r'C:\Users\imagoworks-moongzeee\Desktop\과제\2021.01.21 과제\ddh-eda.csv')
daeyou=pd.read_csv(r'C:\Users\imagoworks-moongzeee\Desktop\과제\2021.01.21 과제\daeyou-eda.csv')

profile_ddh=ProfileReport(ddh)

profile_ddh.to_file(r'C:\Users\imagoworks-moongzeee\Desktop\과제\2021.01.21 과제\profile_ddh.html')
```

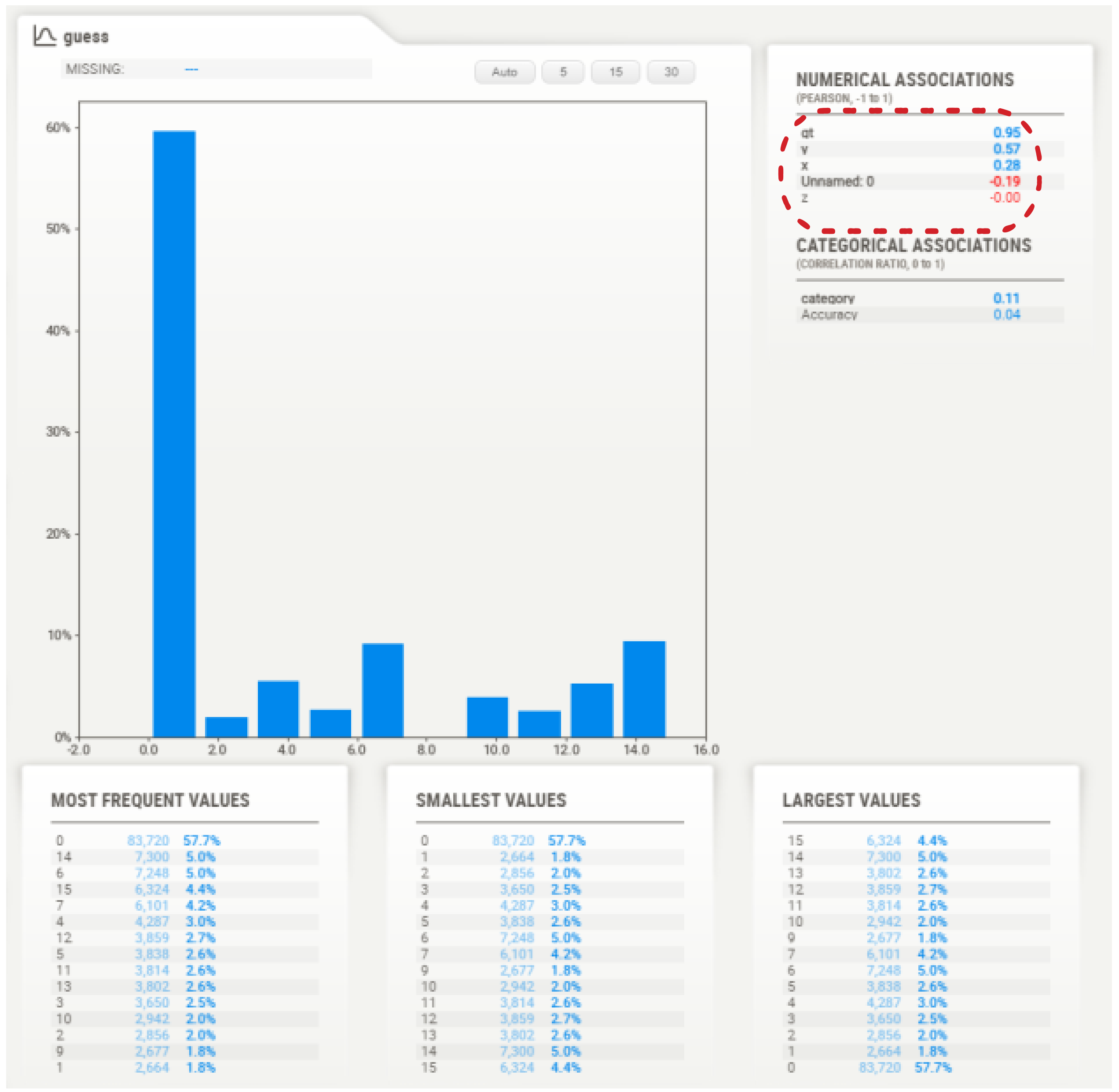
```
Summarize dataset: 100%|██████████| 18/18 [00:05<00:00,  3.59it/s, Completed]
Generate report structure: 100%|██████████| 1/1 [00:01<00:00,  1.50s/it]
Render HTML: 100%|██████████| 1/1 [00:00<00:00,  1.40it/s]
Export report to file: 100%|██████████| 1/1 [00:00<00:00, 250.66it/s]
```

```
profile_daeyou=ProfileReport(daeyou)

profile_daeyou

profile_daeyou.to_file(r'C:\Users\imagoworks-moongzeee\Desktop\과제\2021.01.21 과제\profile_daeyou.html')
```

```
Summarize dataset: 100%|██████████| 18/18 [00:03<00:00,  5.18it/s, Completed]
Generate report structure: 100%|██████████| 1/1 [00:01<00:00,  1.46s/it]
Render HTML: 100%|██████████| 1/1 [00:00<00:00,  1.98it/s]
Export report to file: 100%|██████████| 1/1 [00:00<00:00, 252.44it/s]
```

보고서가 html 파일로 생성됨

report



NUMERICAL ASSOCIATIONS
(PEARSON, -1 to 1)

qt	0.95
v	0.57
x	0.28
Unnamed: 0	-0.19
z	-0.00

CATEGORICAL ASSOCIATIONS
(CORRELATION RATIO, 0 to 1)

category	0.11
Accuracy	0.04

MOST FREQUENT VALUES

0	83,720	57.7%
14	7,300	5.0%
6	7,248	5.0%
15	6,324	4.4%
7	6,101	4.2%
4	4,287	3.0%
12	3,859	2.7%
5	3,838	2.6%
11	3,814	2.6%
13	3,802	2.6%
3	3,650	2.5%
10	2,942	2.0%
2	2,856	2.0%
9	2,677	1.8%
1	2,664	1.8%

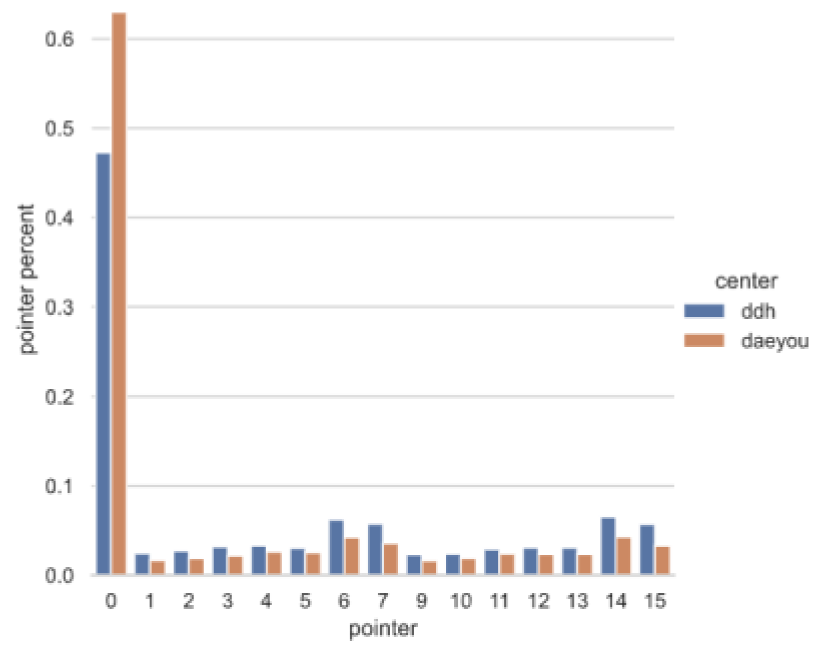
SMALLEST VALUES

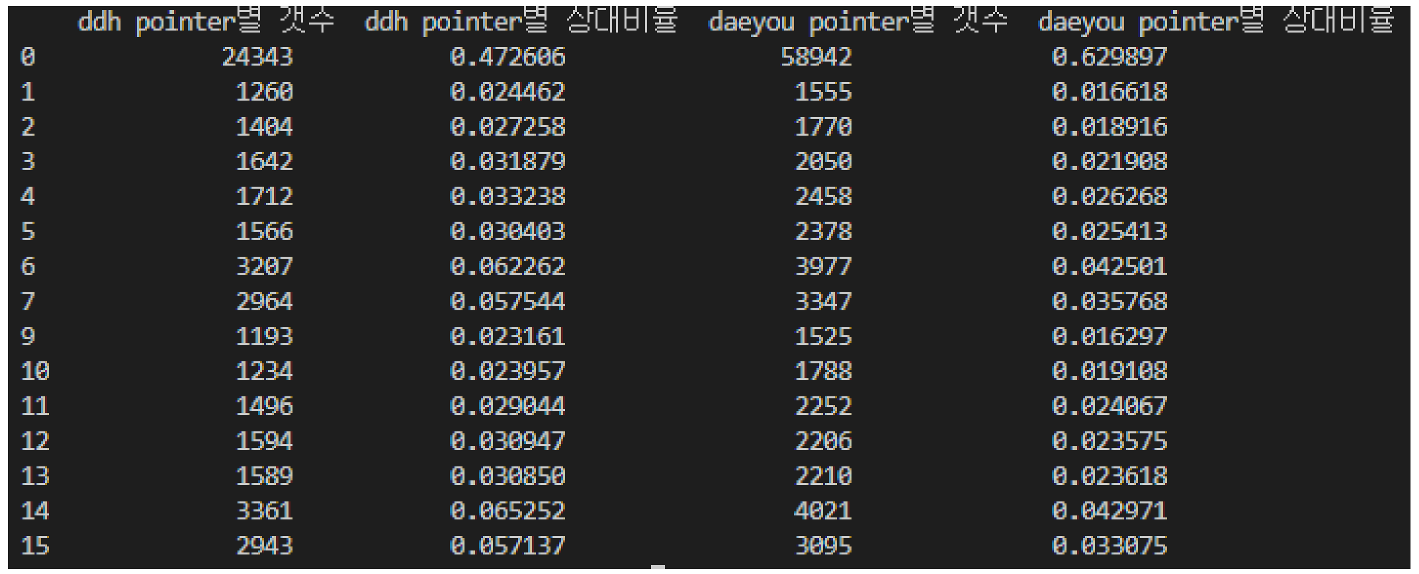
0	83,720	57.7%
1	2,664	1.8%
2	2,856	2.0%
3	3,650	2.5%
4	4,287	3.0%
5	3,838	2.6%
6	7,248	5.0%
7	6,101	4.2%
9	2,677	1.8%
10	2,942	2.0%
11	3,814	2.6%
12	3,859	2.7%
13	3,802	2.6%
14	7,300	5.0%
15	6,324	4.4%

LARGEST VALUES

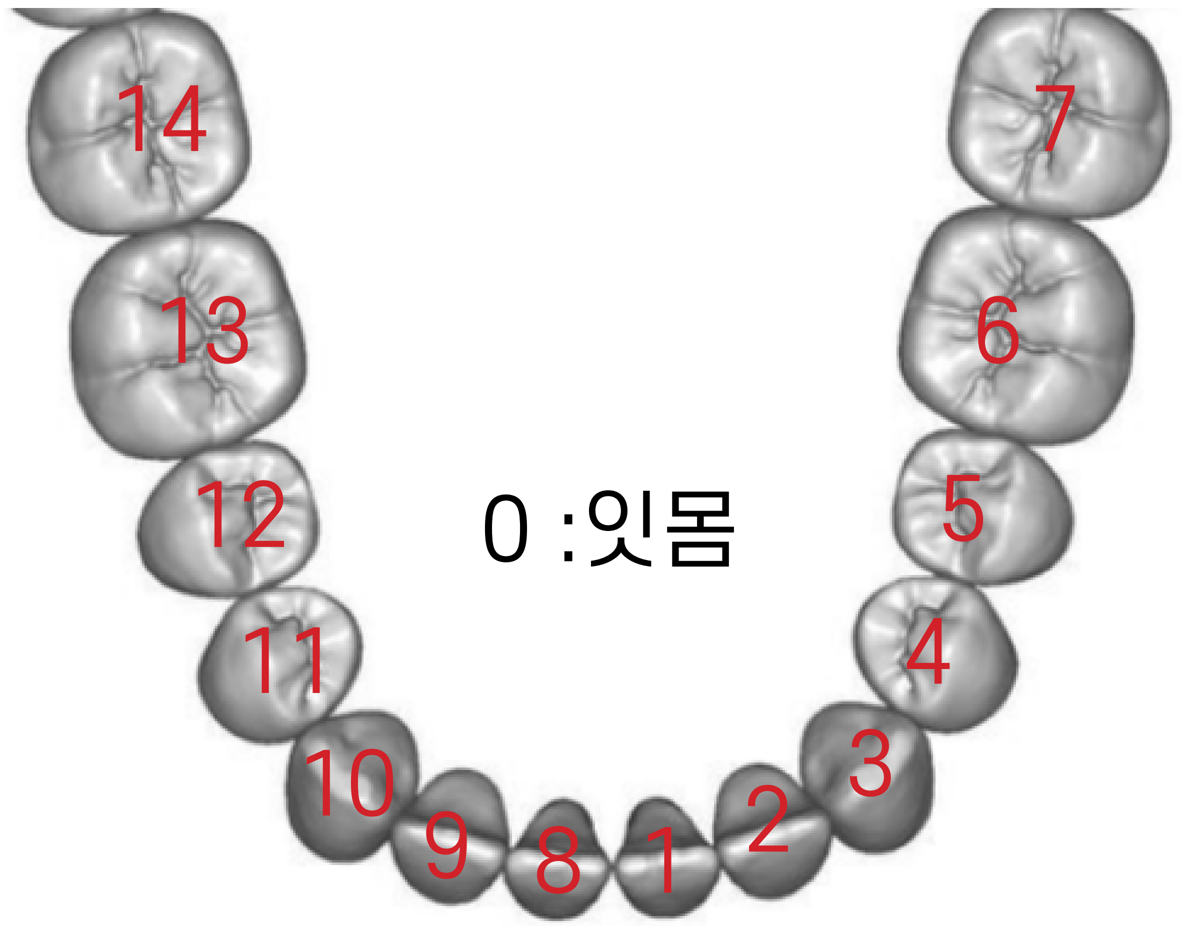
15	6,324	4.4%
14	7,300	5.0%
13	3,802	2.6%
12	3,859	2.7%
11	3,814	2.6%
10	2,942	2.0%
9	2,677	1.8%
7	6,101	4.2%
6	7,248	5.0%
5	3,838	2.6%
4	4,287	3.0%
3	3,650	2.5%
2	2,856	2.0%
1	2,664	1.8%
0	83,720	57.7%

문득 pointer가 뭔지 궁금해서

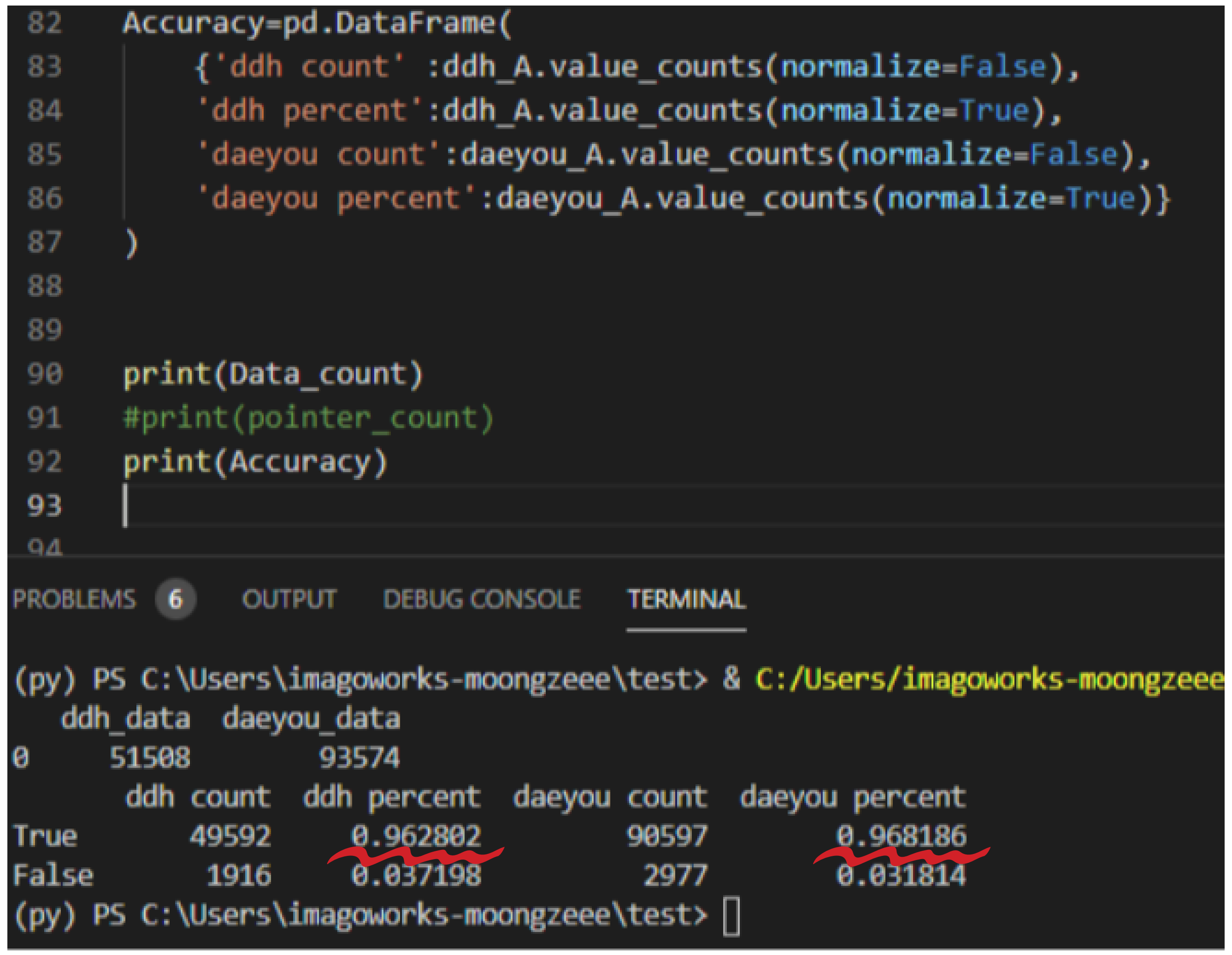




```
    ddh pointer별 갯수  ddh pointer별 상대비율  daeyou pointer별 갯수  daeyou pointer별 상대비율
0             24343          0.472606             58942             0.629897
1              1260          0.024462              1555             0.016618
2              1404          0.027258              1770             0.018916
3              1642          0.031879              2050             0.021908
4              1712          0.033238              2458             0.026268
5              1566          0.030403              2378             0.025413
6              3207          0.062262              3977             0.042501
7              2964          0.057544              3347             0.035768
9              1193          0.023161              1525             0.016297
10             1234          0.023957              1788             0.019108
11             1496          0.029044              2252             0.024067
12             1594          0.030947              2206             0.023575
13             1589          0.030850              2210             0.023618
14             3361          0.065252              4021             0.042971
15             2943          0.057137              3095             0.033075
```



DDH vs DAEYOU



```python
Accuracy=pd.DataFrame(
    {'ddh count' :ddh_A.value_counts(normalize=False),
    'ddh percent':ddh_A.value_counts(normalize=True),
    'daeyou count':daeyou_A.value_counts(normalize=False),
    'daeyou percent':daeyou_A.value_counts(normalize=True)}
)


print(Data_count)
#print(pointer_count)
print(Accuracy)

```

```
(py) PS C:\Users\imagoworks-moongzeee\test> & C:/Users/imagoworks-moongzeee
   ddh_data  daeyou_data
0     51508        93574
       ddh count  ddh percent  daeyou count  daeyou percent
True       49592     0.962802         90597        0.968186
False       1916     0.037198          2977        0.031814
(py) PS C:\Users\imagoworks-moongzeee\test> 
```

DDH Accuracy True의 비율

96.28%

DAEYOU Accuracy True 비율

96.81%

DAEYOU 가

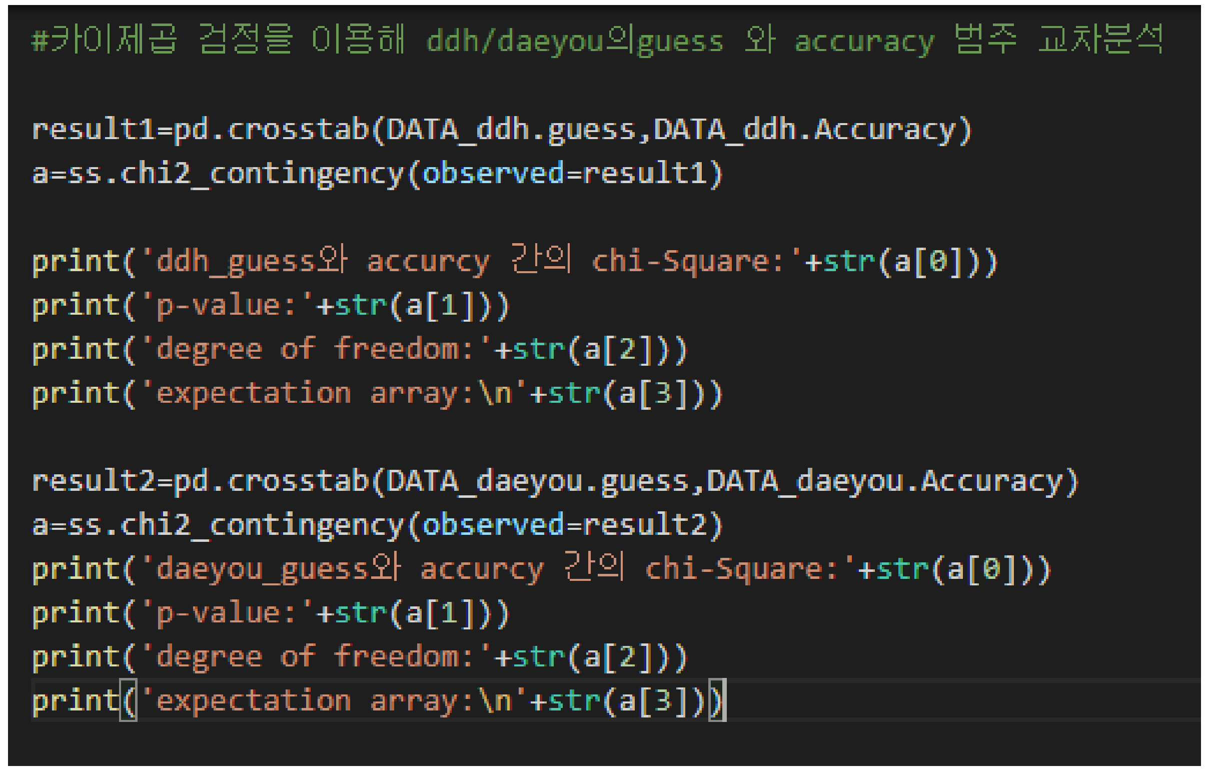
0.2 % 더 정확함

PROJECT 3

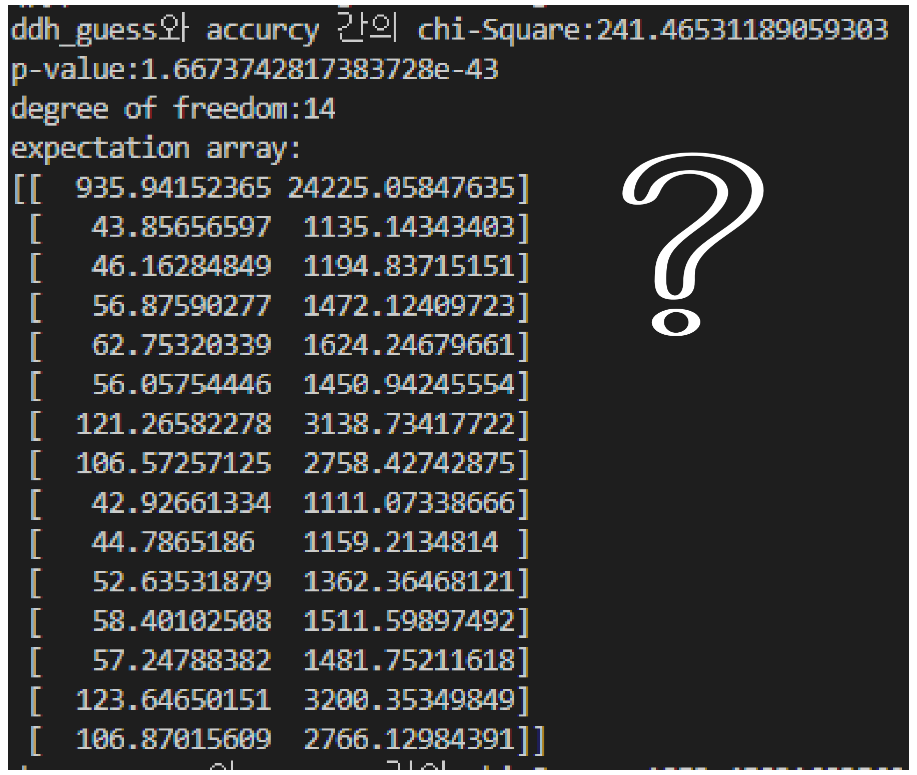
χ^2 - test

H0 : 두 범주간의 관계가 독립적

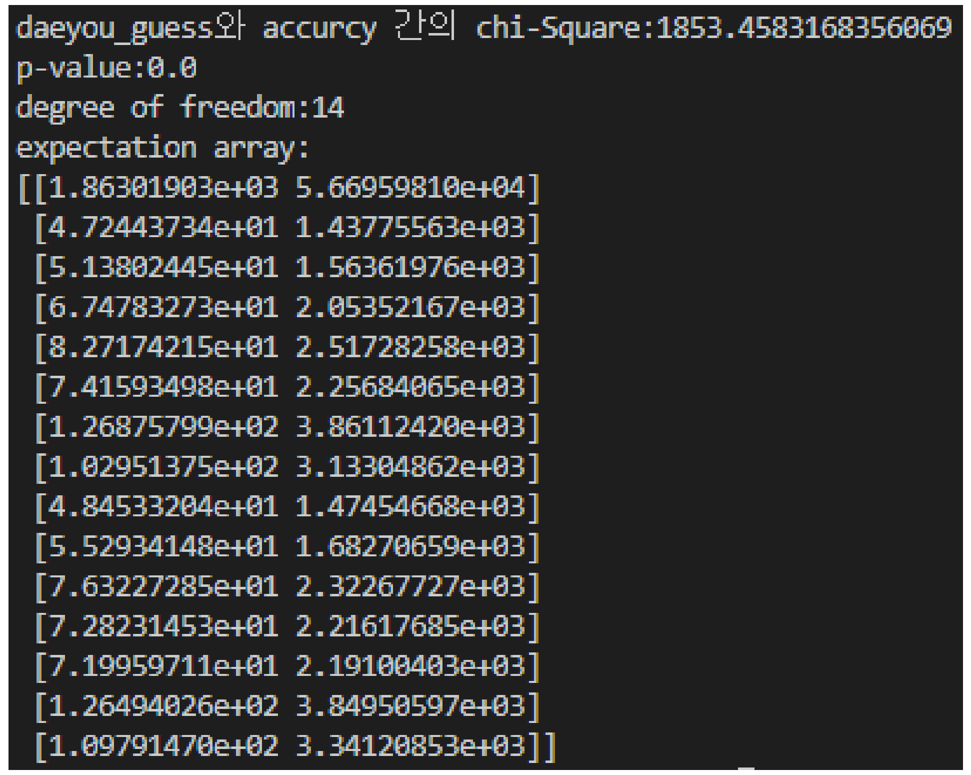
H1 : 두 범주간의 관계가 종속적



```python
#카이제곱 검정을 이용해 ddh/daeyou의guess 와 accuracy 범주 교차분석

result1=pd.crosstab(DATA_ddh.guess,DATA_ddh.Accuracy)
a=ss.chi2_contingency(observed=result1)

print('ddh_guess와 accurcy 간의 chi-Square:'+str(a[0]))
print('p-value:'+str(a[1]))
print('degree of freedom:'+str(a[2]))
print('expectation array:\n'+str(a[3]))

result2=pd.crosstab(DATA_daeyou.guess,DATA_daeyou.Accuracy)
a=ss.chi2_contingency(observed=result2)
print('daeyou_guess와 accurcy 간의 chi-Square:'+str(a[0]))
print('p-value:'+str(a[1]))
print('degree of freedom:'+str(a[2]))
print('expectation array:\n'+str(a[3]))
```



```
ddh_guess와 accurcy 간의 chi-Square:241.46531189059303
p-value:1.6673742817383728e-43
degree of freedom:14
expectation array:
[[  935.94152365 24225.05847635]
 [   43.85656597  1135.14343403]
 [   46.16284849  1194.83715151]
 [   56.87590277  1472.12409723]
 [   62.75320339  1624.24679661]
 [   56.05754446  1450.94245554]
 [  121.26582278  3138.73417722]
 [  106.57257125  2758.42742875]
 [   42.92661334  1111.07338666]
 [   44.7865186   1159.2134814 ]
 [   52.63531879  1362.36468121]
 [   58.40102508  1511.59897492]
 [   57.24788382  1481.75211618]
 [  123.64650151  3200.35349849]
 [  106.87015609  2766.12984391]]
```



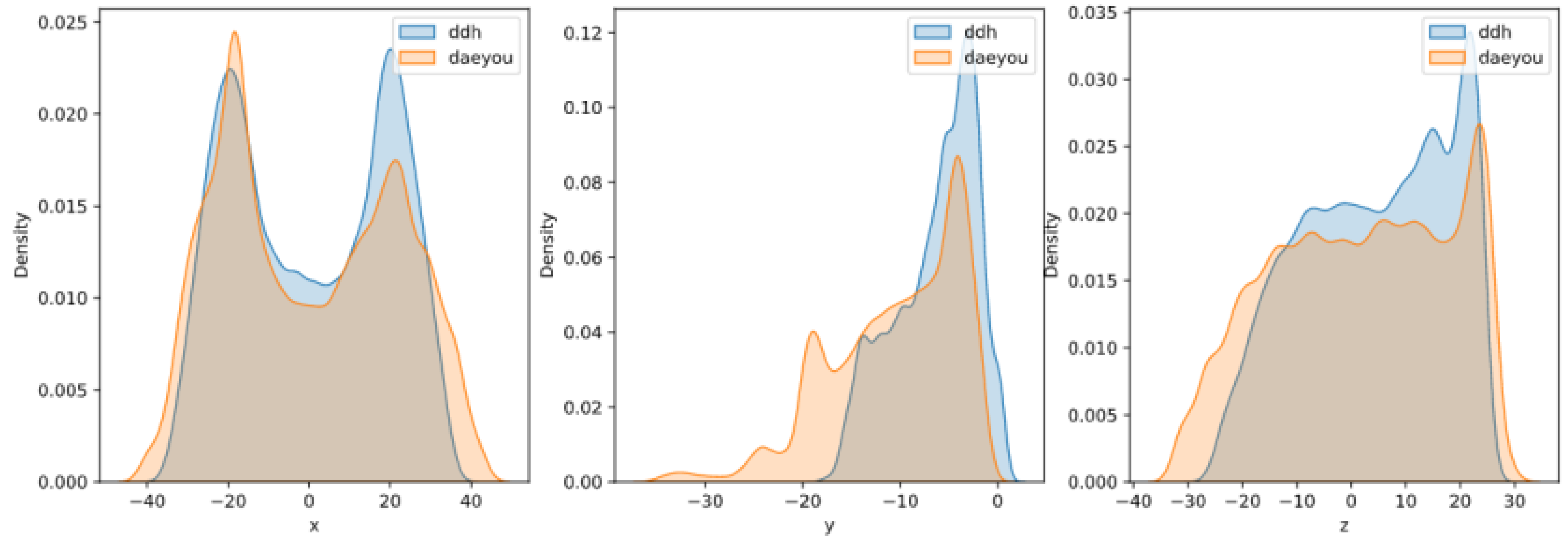
```
daeyou_guess와 accurcy 간의 chi-Square:1853.4583168356069
p-value:0.0
degree of freedom:14
expectation array:
[[1.86301903e+03 5.66959810e+04]
 [4.72443734e+01 1.43775563e+03]
 [5.13802445e+01 1.56361976e+03]
 [6.74783273e+01 2.05352167e+03]
 [8.27174215e+01 2.51728258e+03]
 [7.41593498e+01 2.25684065e+03]
 [1.26875799e+02 3.86112420e+03]
 [1.02951375e+02 3.13304862e+03]
 [4.84533204e+01 1.47454668e+03]
 [5.52934148e+01 1.68270659e+03]
 [7.63227285e+01 2.32267727e+03]
 [7.28231453e+01 2.21617685e+03]
 [7.19959711e+01 2.19100403e+03]
 [1.26494026e+02 3.84950597e+03]
 [1.09791470e+02 3.34120853e+03]]
```

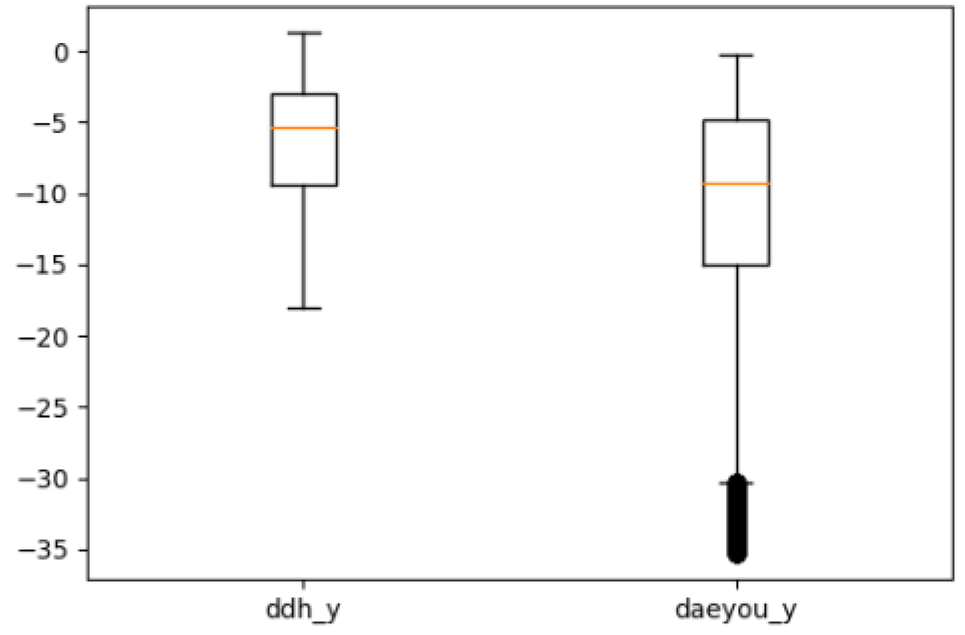
daeyou_guess와 accuracy간의 p-value 값이 0.05 보다 낮으므로 H0 가설은 기각

즉, daeyou_guess 와 accuracy는 연관이 있다.

PROJECT 4

t - test



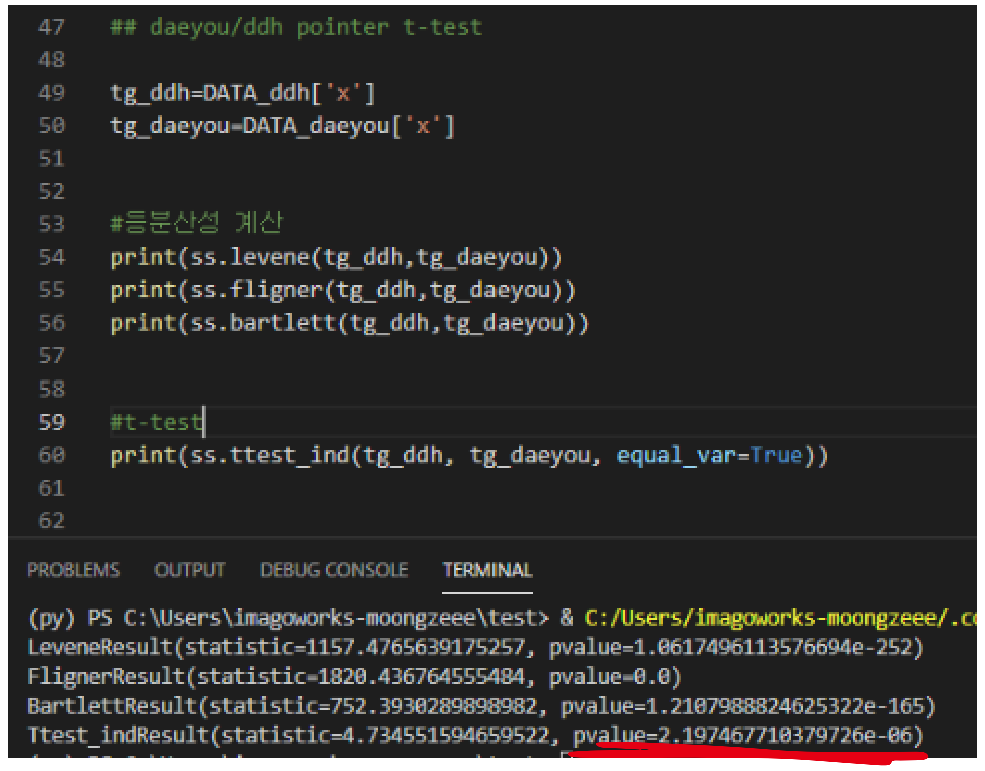


daeyou와 ddh의 y-pointer 값의 차이가 있다

PROJECT 4

t - test

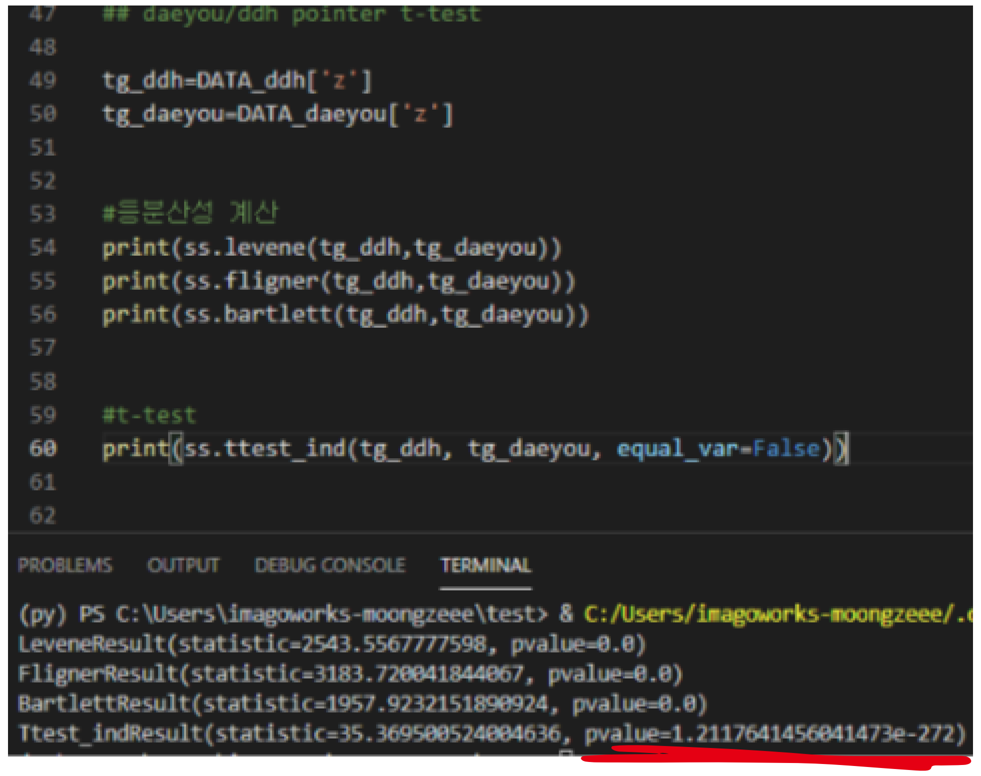
H0 : 두 집단간의 평균의 차이가 없다
H1 : 두 집단간의 평균의 차이가 있다



```
## daeyou/ddh pointer t-test

tg_ddh=DATA_ddh['x']
tg_daeyou=DATA_daeyou['x']


#등분산성 계산
print(ss.levene(tg_ddh,tg_daeyou))
print(ss.fligner(tg_ddh,tg_daeyou))
print(ss.bartlett(tg_ddh,tg_daeyou))


#t-test
print(ss.ttest_ind(tg_ddh, tg_daeyou, equal_var=True))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

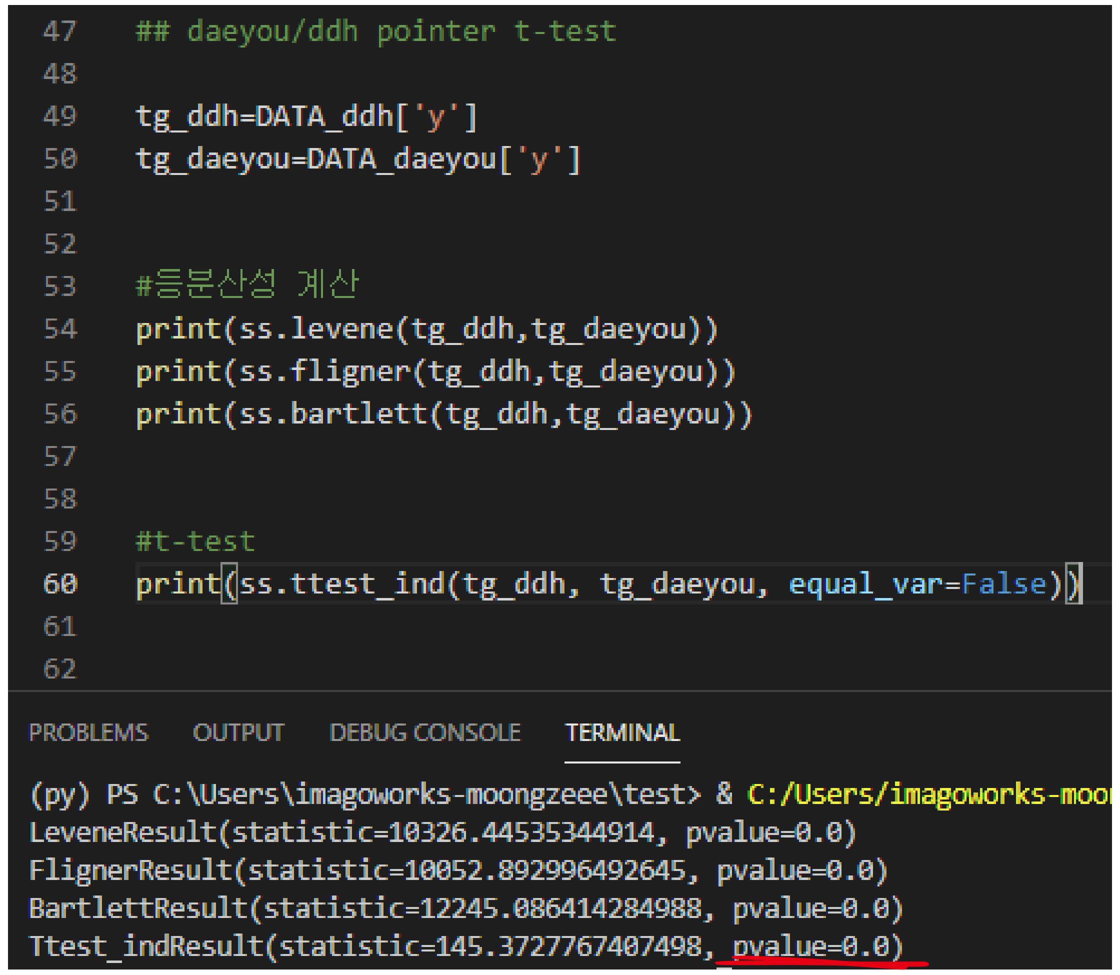
```
(py) PS C:\Users\imagoworks-moongzeee\test> & C:/Users/imagoworks-moongzeee/.c
LeveneResult(statistic=1157.4765639175257, pvalue=1.0617496113576694e-252)
FlignerResult(statistic=1820.436764555484, pvalue=0.0)
BartlettResult(statistic=752.3930289898982, pvalue=1.2107988824625322e-165)
Ttest_indResult(statistic=4.734551594659522, pvalue=2.197467710379726e-06)
```



```
## daeyou/ddh pointer t-test

tg_ddh=DATA_ddh['z']
tg_daeyou=DATA_daeyou['z']


#등분산성 계산
print(ss.levene(tg_ddh,tg_daeyou))
print(ss.fligner(tg_ddh,tg_daeyou))
print(ss.bartlett(tg_ddh,tg_daeyou))


#t-test
print(ss.ttest_ind(tg_ddh, tg_daeyou, equal_var=False))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
(py) PS C:\Users\imagoworks-moongzeee\test> & C:/Users/imagoworks-moongzeee/.
LeveneResult(statistic=2543.5567777598, pvalue=0.0)
FlignerResult(statistic=3183.720041844067, pvalue=0.0)
BartlettResult(statistic=1957.923215189024, pvalue=0.0)
Ttest_indResult(statistic=35.369500524004636, pvalue=1.2117641456041473e-272)
```



```
## daeyou/ddh pointer t-test

tg_ddh=DATA_ddh['y']
tg_daeyou=DATA_daeyou['y']


#등분산성 계산
print(ss.levene(tg_ddh,tg_daeyou))
print(ss.fligner(tg_ddh,tg_daeyou))
print(ss.bartlett(tg_ddh,tg_daeyou))


#t-test
print(ss.ttest_ind(tg_ddh, tg_daeyou, equal_var=False))
```

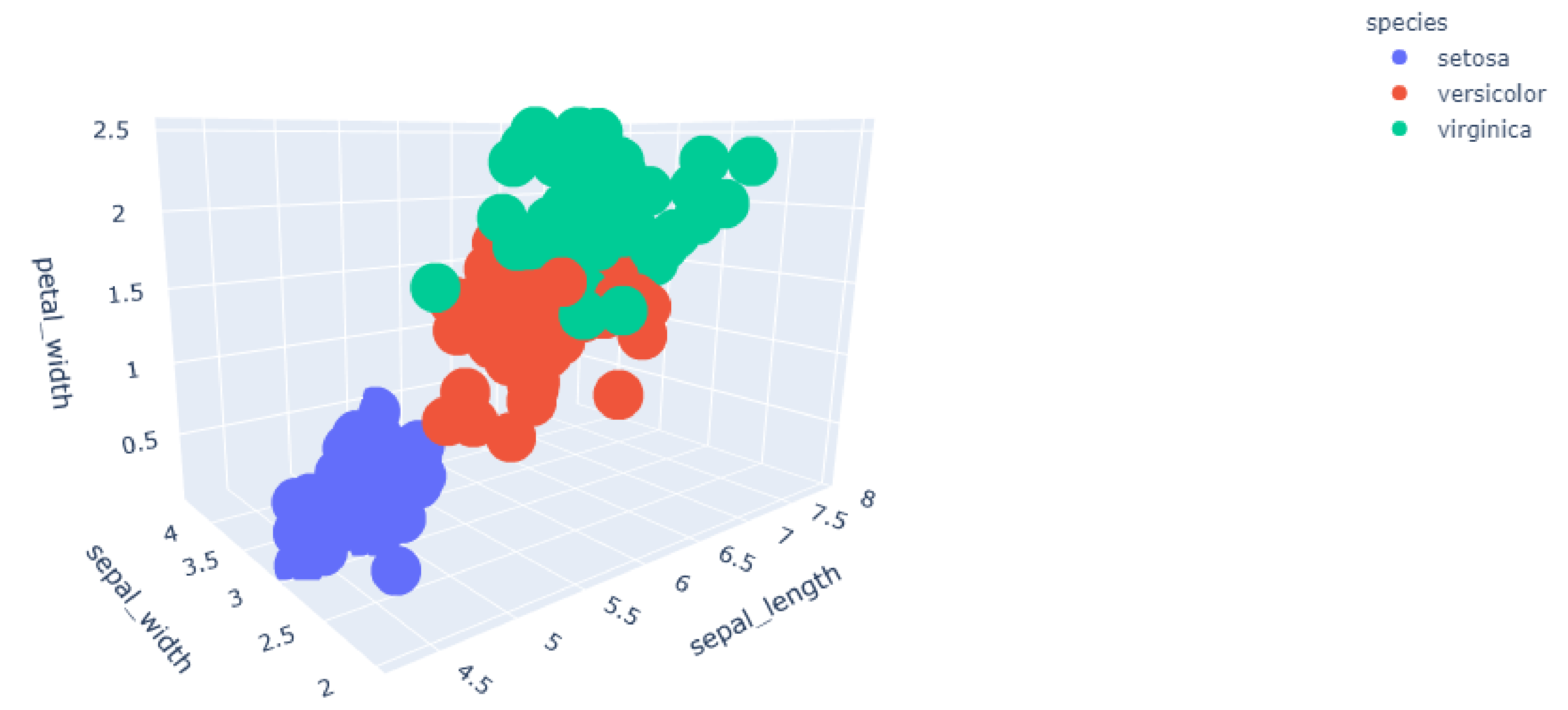
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
(py) PS C:\Users\imagoworks-moongzeee\test> & C:/Users/imagoworks-moo
LeveneResult(statistic=10326.44535344914, pvalue=0.0)
FlignerResult(statistic=10052.892996492645, pvalue=0.0)
BartlettResult(statistic=12245.086414284988, pvalue=0.0)
Ttest_indResult(statistic=145.3727767407498, pvalue=0.0)
```

daeyou_y와 ddh_y간의 p-value 값이 0.05 보다 낮으므로 H0 가설은 기각
즉, daeyou_y 와 ddh_y 의 평균값 차이가 있다.

plotly 패키지

https://plotly.com/python/3d-scatter-plots/


species
● setosa
● versicolor
● virginica
petal_width
sepal_width
sepal_length