

```
In [181]: """
Phần 1: Các thao tác cơ bản
Bước 1: Xử lý cơ bản
1. Xác định số lượng yếu tố (biến số) tham gia vào yêu cầu
2. Thu thập dữ liệu (data collection)
3. Tổng quan dữ liệu VD: df.info()...
4. Xử lý cơ bản:
    - Loại bỏ dòng rỗng
    - Loại bỏ dòng trùng
    - Khảo sát dữ liệu thiếu
    - Xử lý dữ liệu thiếu
5. Kiểm tra lại dữ liệu
"""
```

```
Out[181]: "\nPhần 1: Các thao tác cơ bản\nBước 1: Xử lý cơ bản\n 1. Xác định số lượng yếu tố (biến số) tham gia v
ào yêu cầu\n 2. Thu thập dữ liệu (data collection)\n 3. Tổng quan dữ liệu VD: df.info()...\n 4. Xử lý c
ơ bản:\n    - Loại bỏ dòng rỗng\n    - Loại bỏ dòng trùng\n    - Khảo sát dữ liệu thiếu\n    - Xử lý
dữ liệu thiếu\n 5. Kiểm tra lại dữ liệu\n'
```

```
In [182]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [183]: # Đọc file
df = pd.read_csv('../data/dulieuxettuyendaihoc.csv', header=0, delimiter=',', encoding='utf-8')
```

```
In [184]: # Đọc 5 dòng dữ liệu đầu tiên
df.head(5)
```

```
Out[184]:
```

	MSSV	T1	T2	T3	T4	T5	T6	GT	DT	KV	NGONNGU	TOANLOGICPHANTICH	GIAIQU
0	SV001	7.2	8.4	7.4	7.2	7.4	6.9	F	NaN	2NT	3.25		3.25
1	SV002	5.4	6.3	4.3	4.9	3.0	4.0	M	NaN	1	6.00		4.00
2	SV003	5.6	5.0	2.8	6.1	4.8	5.7	M	NaN	1	5.00		6.75
3	SV004	6.6	5.1	5.9	4.1	6.1	7.4	M	NaN	1	4.25		4.25
4	SV005	6.0	5.4	7.6	4.4	6.8	8.0	M	NaN	2NT	4.25		4.50

In [185]: *# Xem thông tin tổng quan dữ liệu*  
df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 16 columns):
#   Column                Non-Null Count  Dtype
---  -
0   MSSV                  100 non-null   object
1   T1                    100 non-null   float64
2   T2                    100 non-null   float64
3   T3                    100 non-null   float64
4   T4                    100 non-null   float64
5   T5                    100 non-null   float64
6   T6                    100 non-null   float64
7   GT                    100 non-null   object
8   DT                    3 non-null     float64
9   KV                    100 non-null   object
10  NGONNGU               100 non-null   float64
11  TOANLOGICPHANTICH     100 non-null   float64
12  GIAIQUYETVANDE        100 non-null   float64
13  KT                    100 non-null   object
14  NGAYTHI               100 non-null   object
15  DINHHUONGNGHENGHIEP  100 non-null   object
dtypes: float64(10), object(6)
memory usage: 12.6+ KB
```

In [186]: *# Lấy thông tin các cột*  
df = df[['T5','T6','GT','DT','KV','KT','NGONNGU','TOANLOGICPHANTICH',  
'GIAIQUYETVANDE','NGAYTHI','DINHHUONGNGHENGHIEP']]

In [187]: *# Đổi tên cột*  
df.rename(columns={  
 'TOANLOGICPHANTICH':'LOGIC',  
 'GIAIQUYETVANDE':'UNGXU',  
 'DINHHUONGNGHENGHIEP':'HUONGNGHIEP'},  
inplace=True)

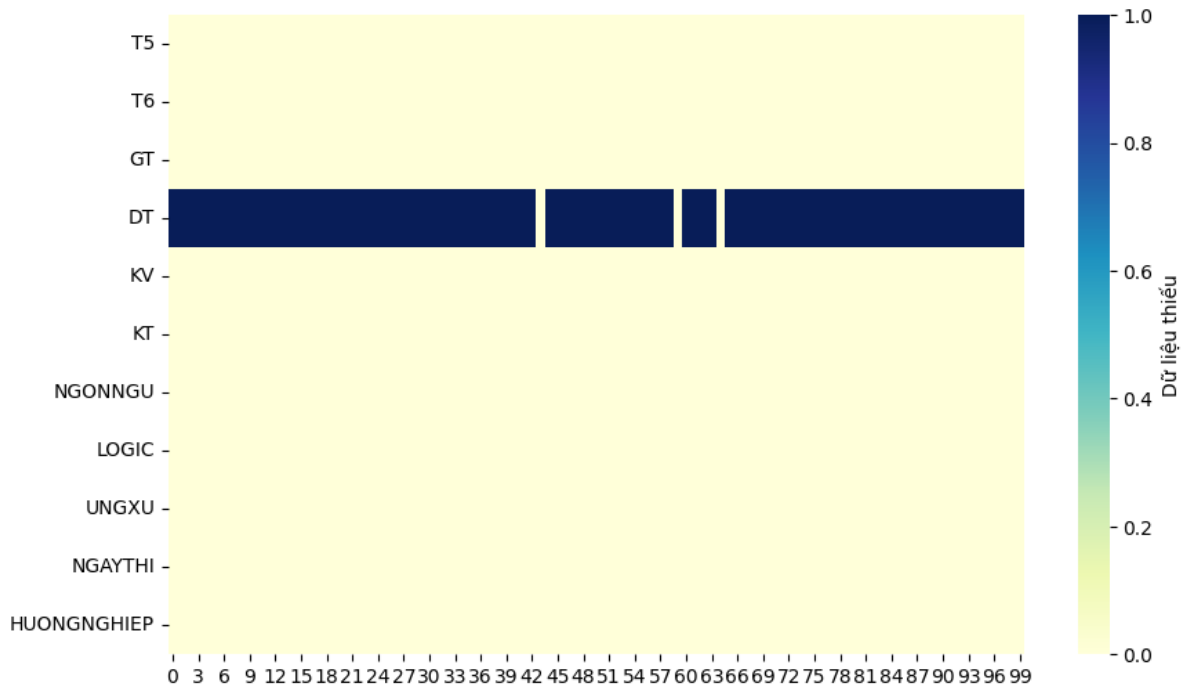
In [188]: df.head(5)

Out[188]:

	T5	T6	GT	DT	KV	KT	NGONNGU	LOGIC	UNGXU	NGAYTHI	HUONGNGHIEP
0	7.4	6.9	F	NaN	2NT	A1	3.25	3.25	4.50	12/7/2018	No
1	3.0	4.0	M	NaN	1	C	6.00	4.00	3.50	12/7/2018	Yes
2	4.8	5.7	M	NaN	1	C	5.00	6.75	4.00	12/7/2018	No
3	6.1	7.4	M	NaN	1	D1	4.25	4.25	5.25	12/7/2018	No
4	6.8	8.0	M	NaN	2NT	A	4.25	4.50	5.00	12/7/2018	No

In [189]: *# Xóa bỏ các dòng dữ liệu rỗng*  
df.dropna(how='all',inplace=True)

```
In [190]: # dùng heatmap để trực quan dữ liệu bị thiếu
plt.figure(figsize=(10,6))
sns.heatmap(df.isna().transpose(),cmap='YlGnBu',
cbar_kws={'label':'Dữ liệu thiếu'})
plt.savefig('missingdata.png',dpi=100)
```



```
In [191]: """
Note: Với dữ liệu bị thiếu:
1. Cần xác định biến số nào bị thiếu
2. Mức độ thiếu dữ liệu
3. Có cần phải xử lý không
"""
```

```
Out[191]: '\nNote: Với dữ liệu bị thiếu:\n 1. Cần xác định biến số nào bị thiếu\n 2. Mức độ thiếu dữ liệu\n 3. Có cần phải xử lý không\n'
```

```
In [192]: # Điền giá trị thiếu
df['DT'].fillna('KINH',inplace=True)
# Lưu ý: Với biến định tính ta có thể thay bằng giá trị yếu vị (mode)
# df['DT'].fillna(df['DT'].mode()[0],inplace=True)
```

C:\Users\Lan Anh\AppData\Local\Temp\ipykernel\_7276\614927762.py:2: FutureWarning: Setting an item of incompatible dtype is deprecated and will raise in a future error of pandas. Value 'KINH' has dtype incompatible with float64, please explicitly cast to a compatible dtype first.

```
df['DT'].fillna('KINH',inplace=True)
```

```
In [193]: # Điền thiếu giá trị phần NGONNGU bằng 0 (nếu có)
df['NGONNGU'].fillna(0,inplace=True)
# Điền thiếu giá trị phần LOGIC bằng trung bình (nếu có)
df['LOGIC'].fillna(df['LOGIC'].mean(), inplace=True)
# Điền thiếu giá trị phần UNGXU bằng trung vị (nếu có)
df['UNGXU'].fillna(df['UNGXU'].median(),inplace=True)
# Lưu ý: Với biến định lượng thì ta nên thay bằng trung vị
```

```
In [194]: df.head(5)
```

```
Out[194]:
```

	T5	T6	GT	DT	KV	KT	NGONNGU	LOGIC	UNGXU	NGAYTHI	HUONGNGHIEP
0	7.4	6.9	F	KINH	2NT	A1	3.25	3.25	4.50	12/7/2018	No
1	3.0	4.0	M	KINH	1	C	6.00	4.00	3.50	12/7/2018	Yes
2	4.8	5.7	M	KINH	1	C	5.00	6.75	4.00	12/7/2018	No
3	6.1	7.4	M	KINH	1	D1	4.25	4.25	5.25	12/7/2018	No
4	6.8	8.0	M	KINH	2NT	A	4.25	4.50	5.00	12/7/2018	No

```
In [195]: """
Phần 2: Kỹ thuật Feature Engineering (thường dùng cho Machine Learning)
Nếu chỉ là xử lý phân tích dữ liệu thì ta gọi là New Attribute
Đây là kỹ thuật tạo thêm hoặc biến đổi số liệu sẵn có thành các biến số mới phù
hợp với nghiệp vụ để phân tích
"""
```

```
Out[195]: "\nPhần 2: Kỹ thuật Feature Engineering (thường dùng cho Machine Learning)\n Nếu chỉ là xử lý phân tí
ch dữ liệu thì ta gọi là New Attribute\n Đây là kỹ thuật tạo thêm hoặc biến đổi số liệu sẵn có thành các bi
ến số mới phùn hợp với nghiệp vụ để phân tích\n'
```

```
In [196]: # Tạo biến TBTOAN: trung bình toán lớp 12
```

```
df['TBTOAN'] = (df['T5']+df['T6'])/2
```

```
In [197]: df.head(5)
```

```
Out[197]:
```

	T5	T6	GT	DT	KV	KT	NGONNGU	LOGIC	UNGXU	NGAYTHI	HUONGNGHIEP	TBTOAN
0	7.4	6.9	F	KINH	2NT	A1	3.25	3.25	4.50	12/7/2018	No	7.1
1	3.0	4.0	M	KINH	1	C	6.00	4.00	3.50	12/7/2018	Yes	3.5
2	4.8	5.7	M	KINH	1	C	5.00	6.75	4.00	12/7/2018	No	5.2
3	6.1	7.4	M	KINH	1	D1	4.25	4.25	5.25	12/7/2018	No	6.7
4	6.8	8.0	M	KINH	2NT	A	4.25	4.50	5.00	12/7/2018	No	7.4

```
In [198]: # Tạo biến XEPLOAI: đánh giá môn toán dựa trên df['TBTOAN']
df.loc[df['TBTOAN'] < 5.0, 'XEPLOAI'] = 'FAIL'
df.loc[(df['TBTOAN'] >= 5.0) & (df['TBTOAN'] < 7.0), 'XEPLOAI'] = 'FAIR'
df.loc[(df['TBTOAN'] >= 7.0) & (df['TBTOAN'] < 9.0), 'XEPLOAI'] = 'GOOD'
df.loc[(df['TBTOAN'] >= 9.0), 'XEPLOAI'] = 'EXCEL'
```

C:\Users\Lan Anh\AppData\Local\Temp\ipykernel\_7276\1779497314.py:2: FutureWarning: Setting an item of incompatible dtype is deprecated and will raise in a future error of pandas. Value 'FAIL' has dtype incompatible with float64, please explicitly cast to a compatible dtype first.

```
df.loc[df['TBTOAN'] < 5.0, 'XEPLOAI'] = 'FAIL'
```

```
In [199]: df.head(5)
```

```
Out[199]:
```

	T5	T6	GT	DT	KV	KT	NGONNGU	LOGIC	UNGXU	NGAYTHI	HUONGNGHIEP	TBTOAI
0	7.4	6.9	F	KINH	2NT	A1	3.25	3.25	4.50	12/7/2018	No	7.1
1	3.0	4.0	M	KINH	1	C	6.00	4.00	3.50	12/7/2018	Yes	3.5
2	4.8	5.7	M	KINH	1	C	5.00	6.75	4.00	12/7/2018	No	5.2
3	6.1	7.4	M	KINH	1	D1	4.25	4.25	5.25	12/7/2018	No	6.7
4	6.8	8.0	M	KINH	2NT	A	4.25	4.50	5.00	12/7/2018	No	7.4

```
In [200]: # Xem thông tin 5 dòng đầu gồm các cột TBTOAN, XEPLOAI
df[['TBTOAN','XEPLOAI']].head(5)
```

```
Out[200]:
```

	TBTOAN	XEPLOAI
0	7.15	GOOD
1	3.50	FAIL
2	5.25	FAIR
3	6.75	FAIR
4	7.40	GOOD

```
In [201]: """
Tạo biến nhóm khối thi NHOMKT thỏa mãn
A1: G1
C: G3
D1: G3
A: G1
B: G2
"""

dict_map = {'A1':'G1','C':'G3','D1':'G3','A':'G1','B':'G2'}
df['NHOMKT'] = df['KT'].map(dict_map)
```

In [202]: `df[['KT','NHOMKT']].head(5)`

Out[202]:

	KT	NHOMKT
0	A1	G1
1	C	G3
2	C	G3
3	D1	G3
4	A	G1

In [203]: `# Tạo biến số điểm cộng: CONG`  
`"""`  
 Nếu khối thi thuộc nhóm G1, G2 và TBTOAN >= 5.0 thì là 1.0  
 Ngược lại thì là 0.0  
`"""`  
`def fplus(x,y):`  
 `if(x == 'G1' or x == 'G2') and (y>=5.0):`  
 `return 1.0`  
 `else:`  
 `return 0.0`  
`df['CONG'] = list(map(fplus,df['NHOMKT'],df['TBTOAN']))`

In [204]: `df[['TBTOAN','NHOMKT','CONG']].head(5)`

Out[204]:

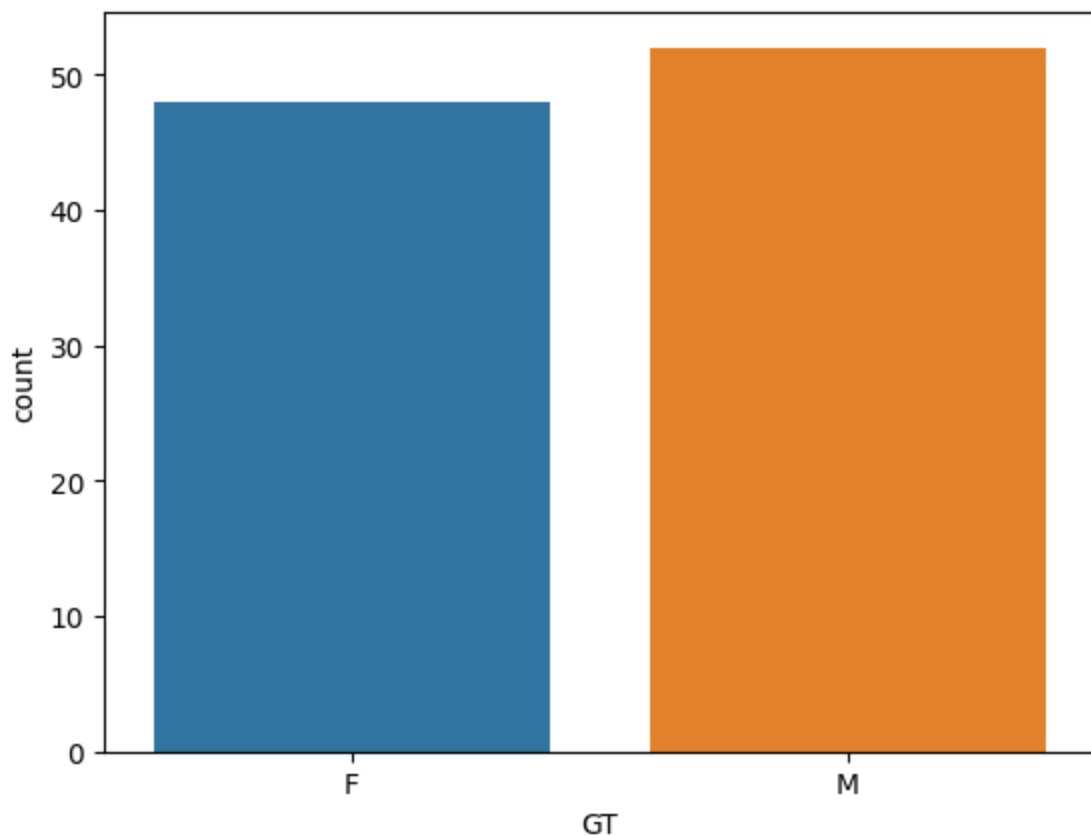
	TBTOAN	NHOMKT	CONG
0	7.15	G1	1.0
1	3.50	G3	0.0
2	5.25	G3	0.0
3	6.75	G3	0.0
4	7.40	G1	1.0

In [205]: `"""`  
 Để trực quan số liệu ta cần lưu ý: Mục đích, sự phối hợp giữa các loại biến để  
 chọn lựa biểu đồ phù hợp đi kèm số liệu trực quan  
 Định tính: bar, pie  
 Định lượng: line , histogram, box-plot, scatter  
`"""`

Out[205]: `"""Để trực quan số liệu ta cần lưu ý: Mục đích, sự phối hợp giữa các loại biến để\nchọn lựa biểu đồ phù hợp\nđi kèm số liệu trực quan\nĐịnh tính: bar, pie\nĐịnh lượng: line , histogram, box-plot, scatter\n"""`

```
In [206]: # Hãy trực quan số lượng học sinh theo giới tính
# Biểu đồ bar
sns.countplot(x='GT', data=df)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):  
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):  
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):



```
In [207]: # Lưu ý
"""
Các biến dùng để phân nhóm, gom nhóm thông thường là biến định tính
và nằm ở các thang đo mức 1,2,3,4
"""
# tương ứng định danh, phân loại, thứ bậc và khoảng
```

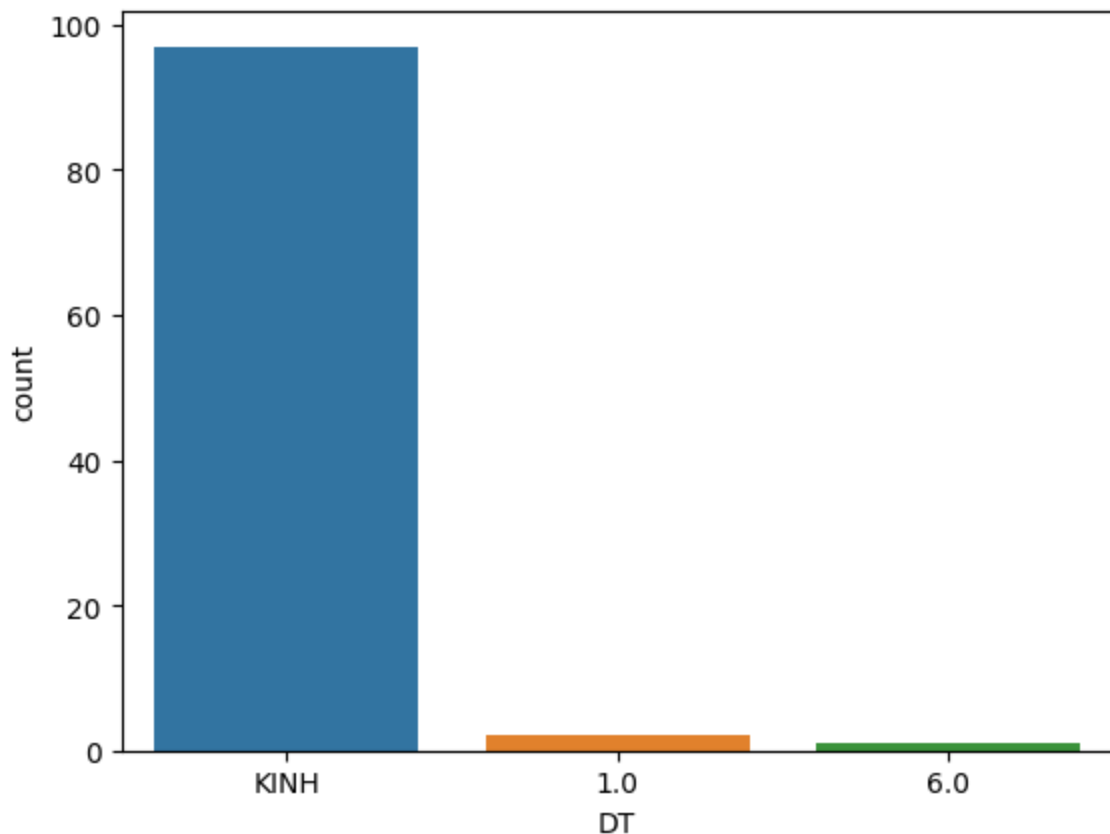
```
Out[207]: \nCác biến dùng để phân nhóm, gom nhóm thông thường là biến định tính\nvà nằm ở các thang đo mức 1, 2,3,4\n'
```

```
In [208]: """
Dựa trên biểu đồ DT cho biết tạo sao ta không phân tích theo nhóm DT:
vì đa số là dân tộc Kinh
"""
# Tương tự cho các cột KV, DT, KT
# DT
```

Out[208]: \nDựa trên biểu đồ DT cho biết tạo sao ta không phân tích theo nhóm DT:\nvì đa số là dân tộc Kinh\n'

```
In [209]: sns.countplot(x='DT', data=df)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):





```
In [210]: sns.countplot(x='KV', data=df)  
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

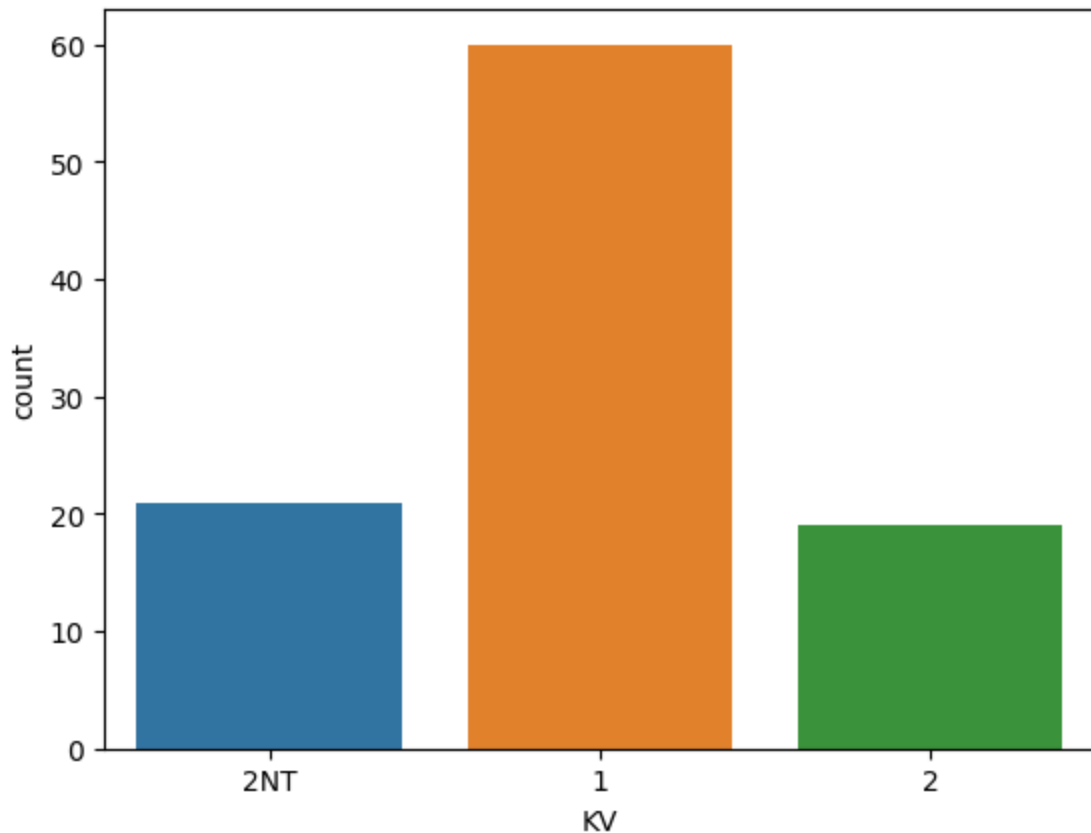
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [211]: sns.countplot(x='KT', data=df)  
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

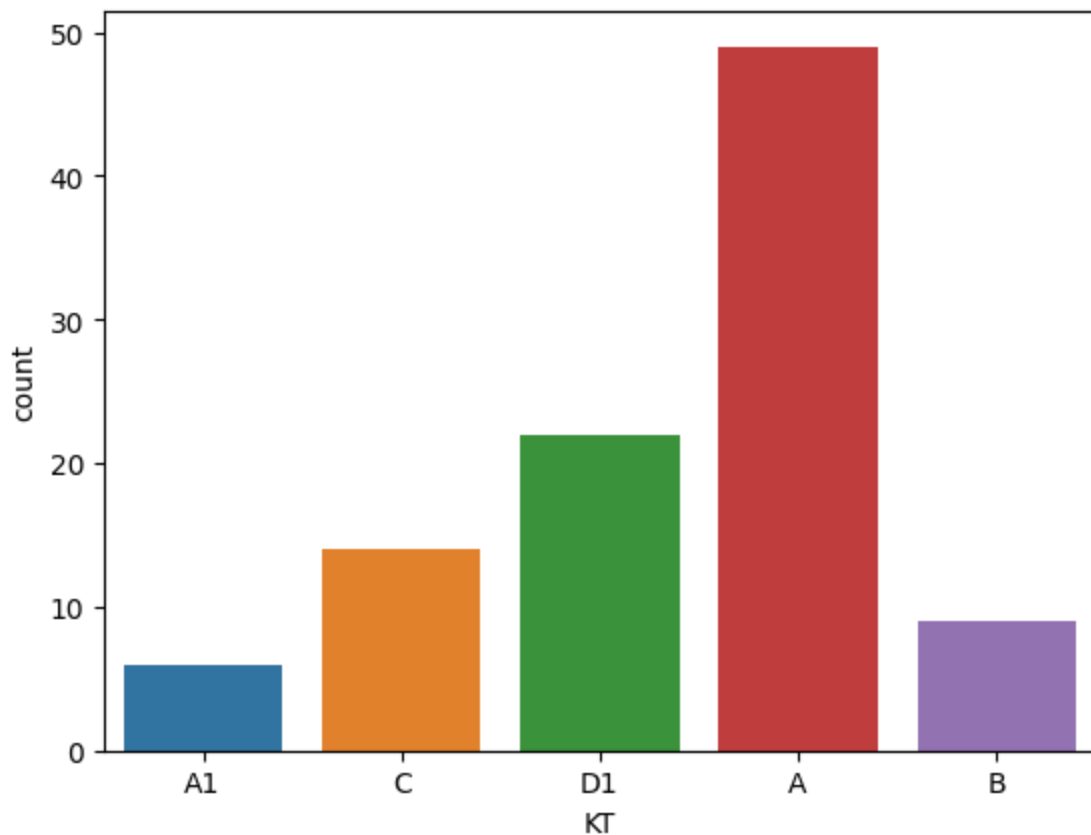
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

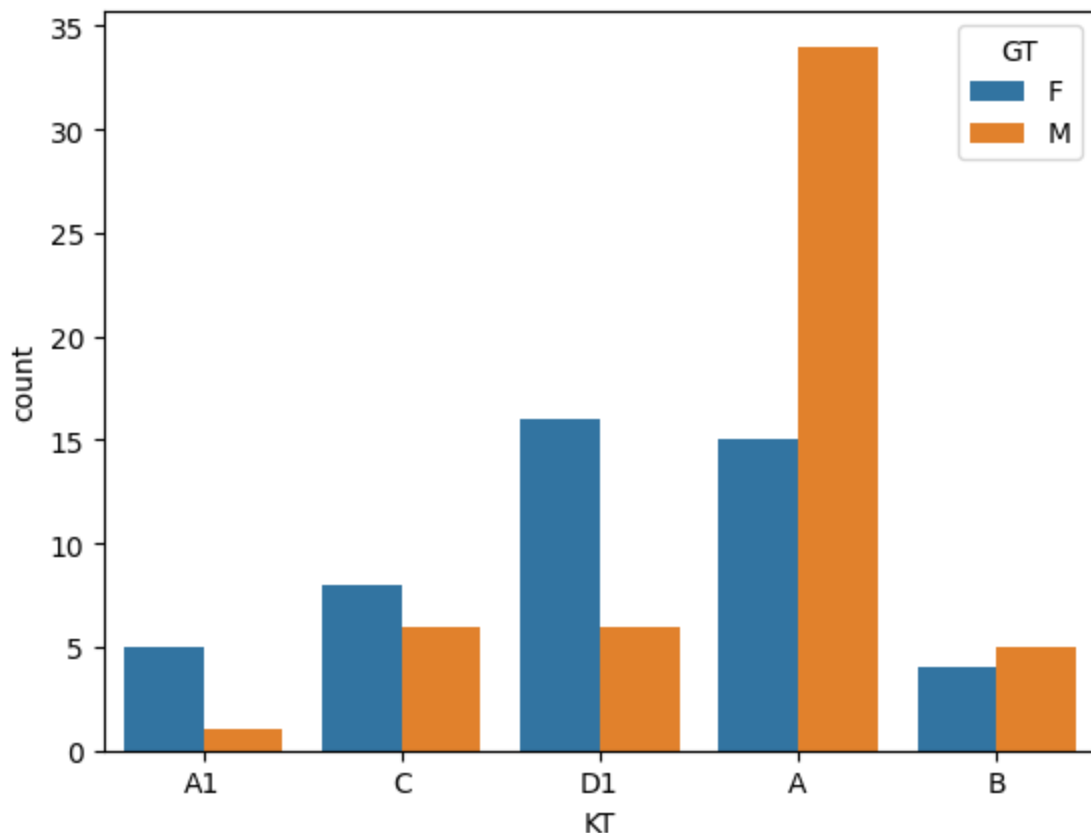
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [212]: """
Hãy so sánh số lượng học sinh đăng ký khối thi dựa trên nhóm giới tính
"""
sns.countplot(x='KT', hue='GT', data=df)
plt.show()
```

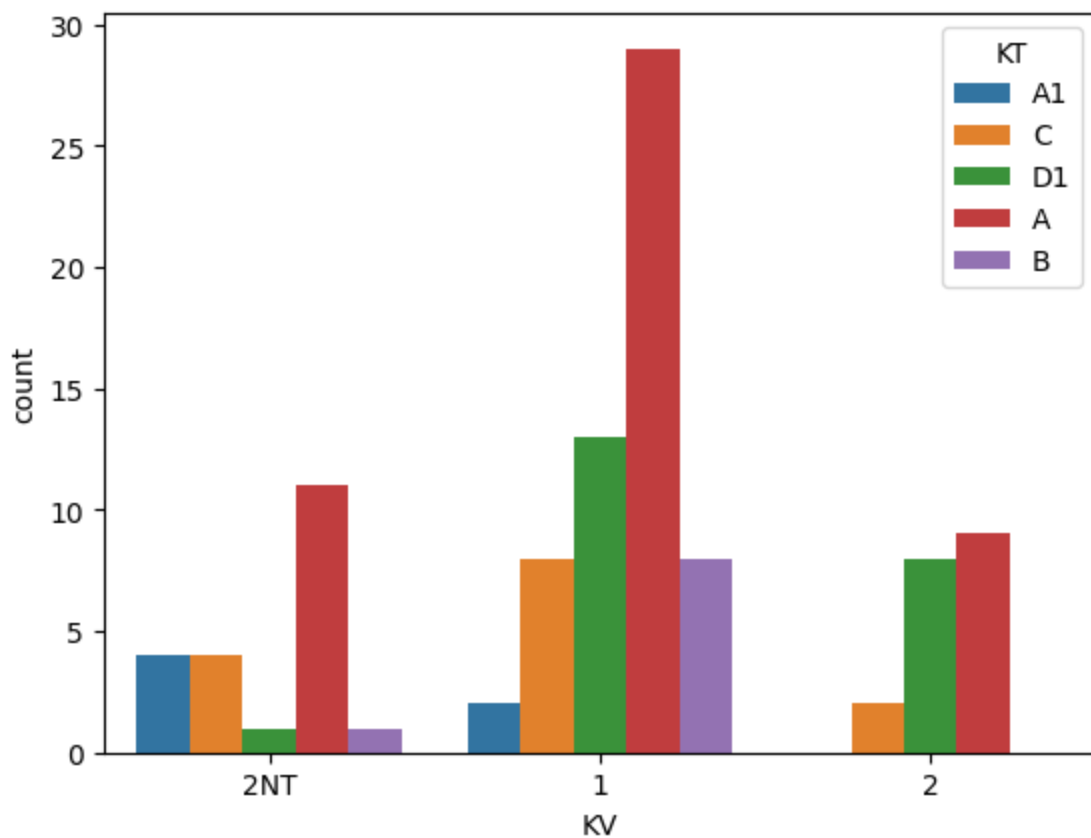
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):



```
In [213]: """
Làm tương tự cho các nhóm biến định tính (KV,KT)
Hãy cho biết khối A có sinh viên khu vực nào đăng ký cao nhất
Trả lời: KV1
"""

sns.countplot(x='KV',hue='KT',data=df)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
if pd.api.types.is\_categorical\_dtype(vector):



```
In [214]: """  
Hãy so sánh điểm trung bình NGONNGU theo nhóm giới tính  
"""  
sns.barplot(x='GT',y='NGONNGU',data=df,errorbar=None)  
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

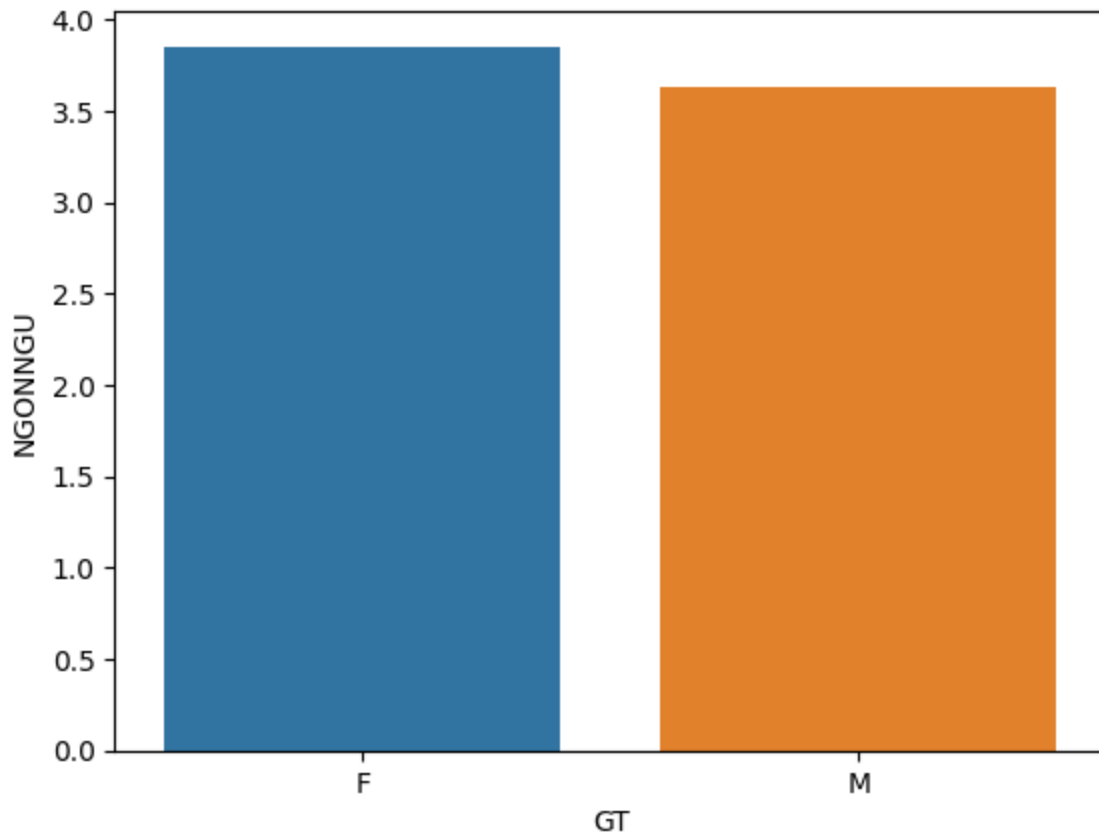
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [215]: # Hãy so sánh điểm LOGIC theo nhóm KT (nhóm khối thi)
sns.barplot(x='NHOMKT',y='LOGIC',data=df,errorbar=None)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

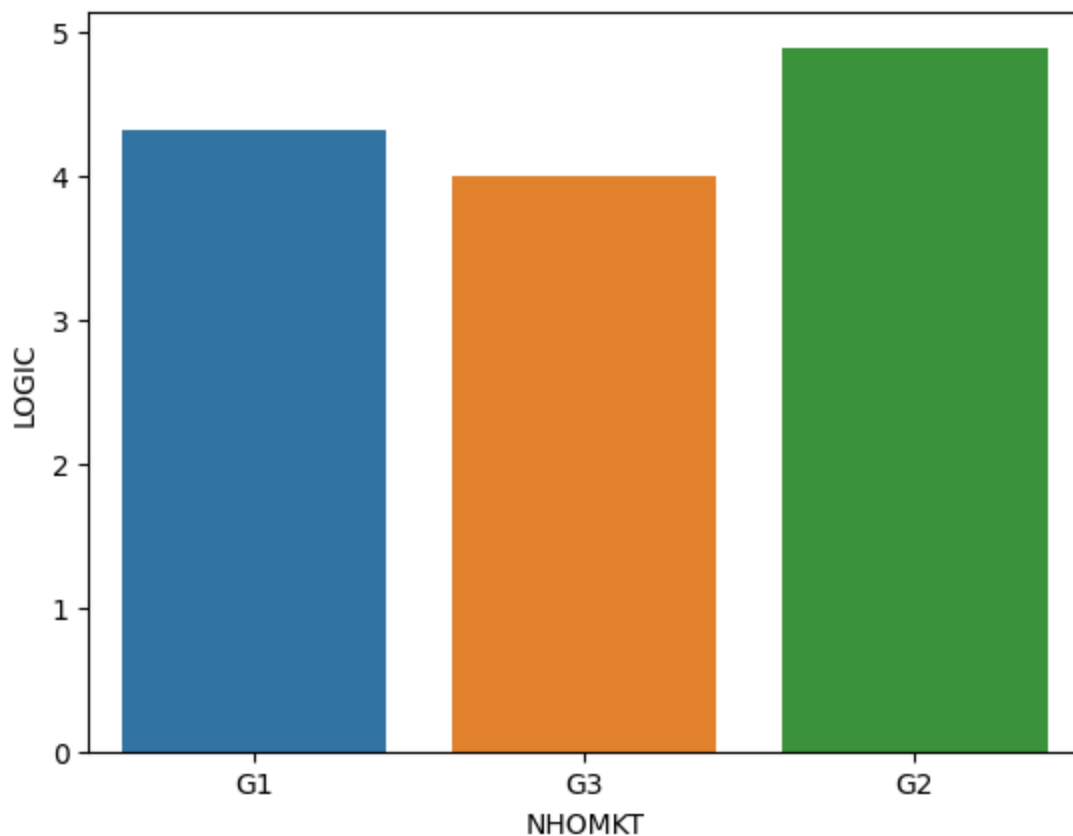
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [216]: """
So sánh điểm trung bình của NGONNGU theo nhóm GT dựa trên KT
"""
sns.barplot(x='GT',y='NGONNGU',hue='KT',data=df,errorbar=None)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

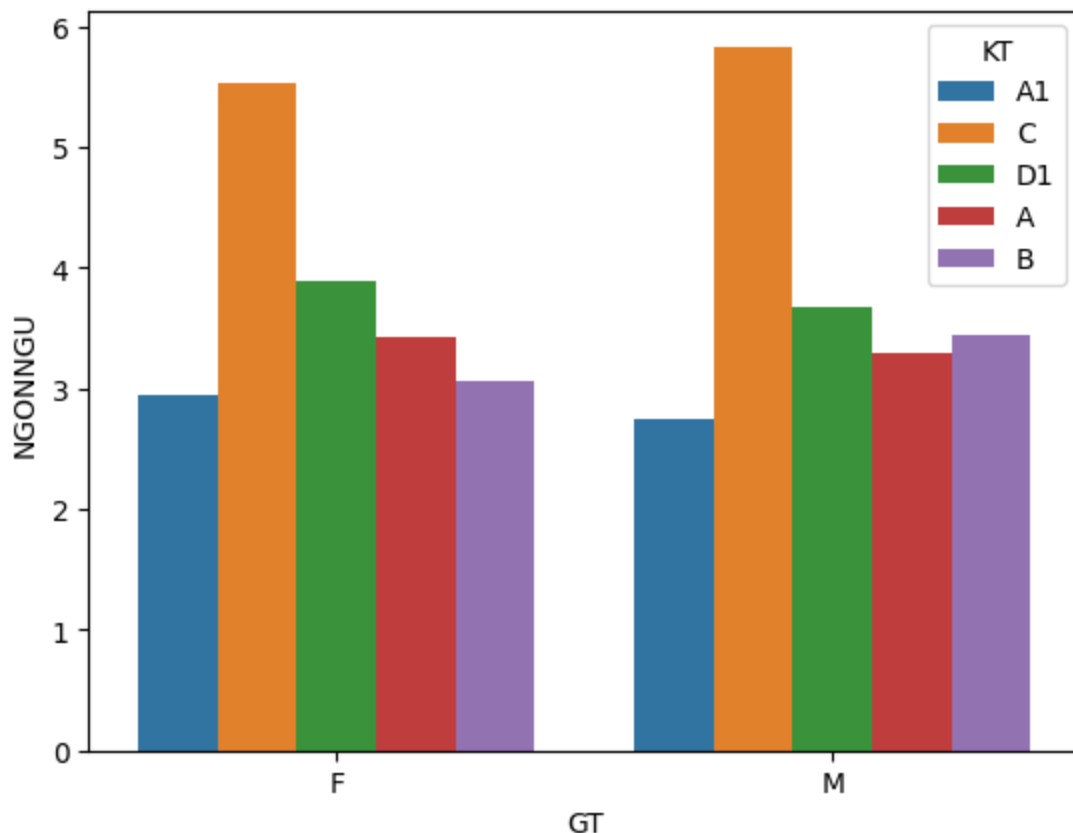
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [217]: # So sánh sai số trên NGONNGU theo nhóm GT theo KT
# Sai số càng cao độ tin cậy càng thấp
sns.barplot(x='GT',y='NGONNGU',hue='KT',data=df)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

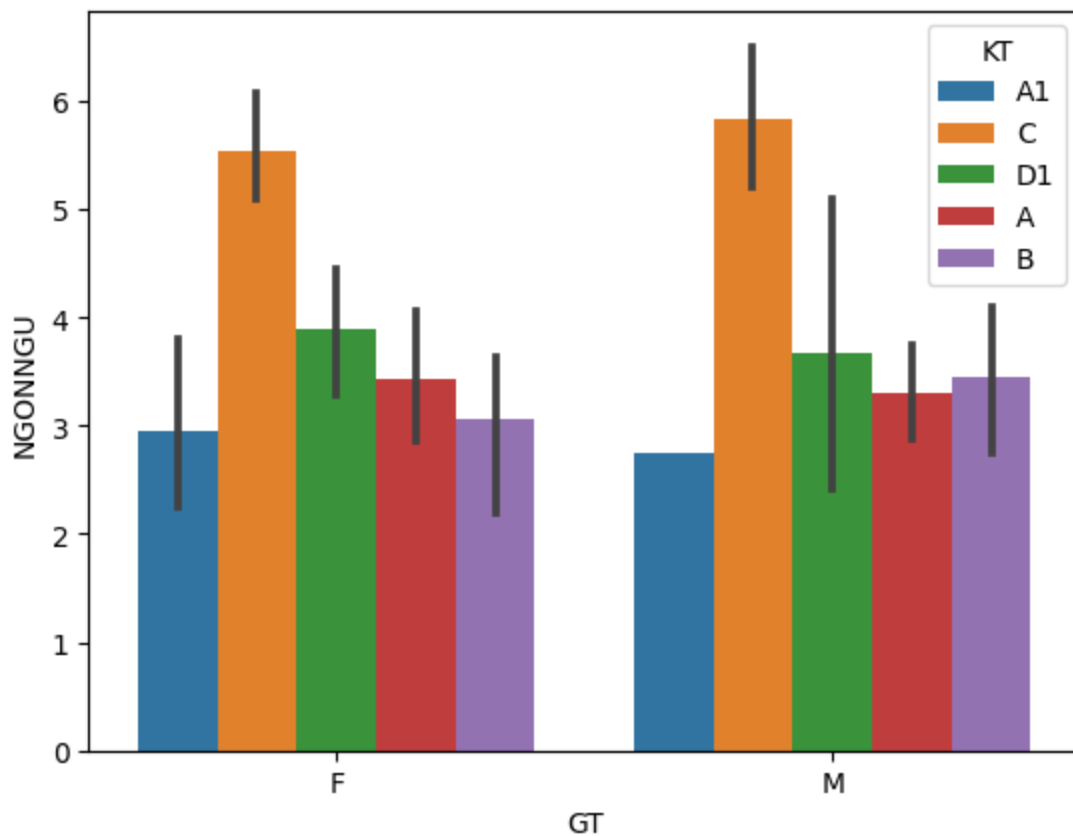
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):





```
In [218]: """
So sánh điểm cao nhất của NGONNGU theo nhóm GT theo KT
Lưu ý: không để estimator thì mặc định là mean
estimator: count, min max sum std, mean (default)
"""
sns.barplot(x='GT',y='NGONNGU',hue='KT',data=df,errorbar=None,estimator=max)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

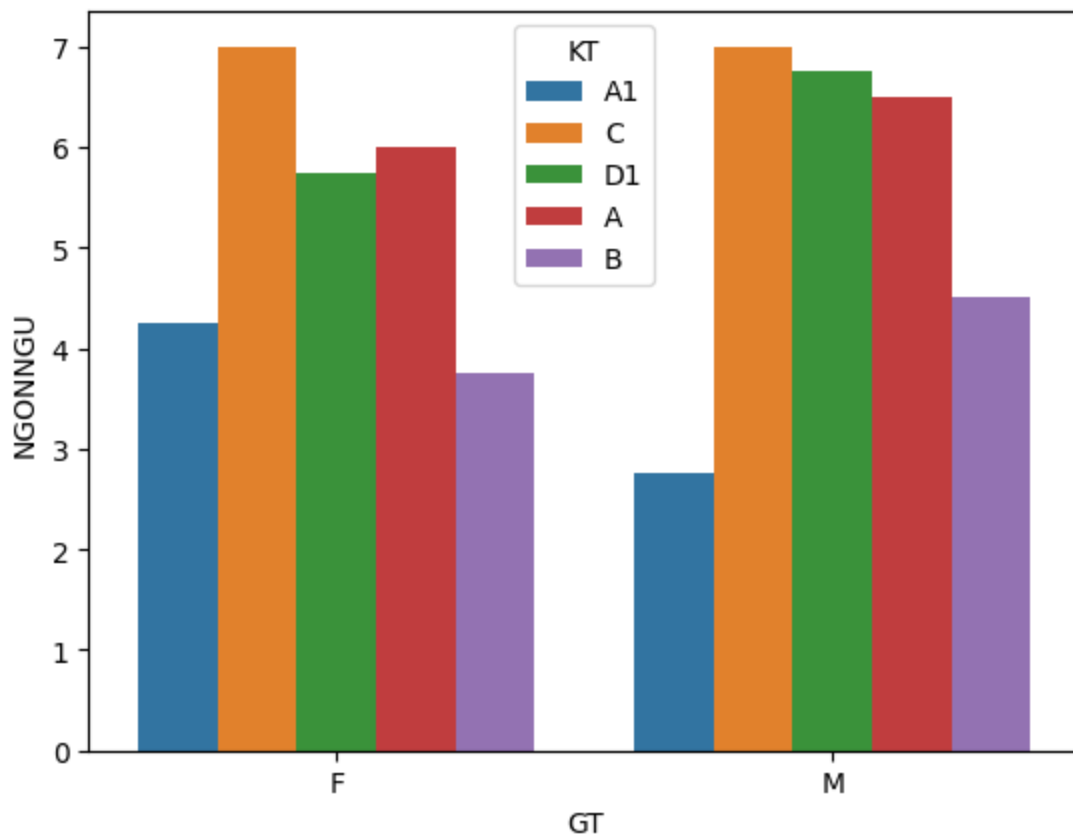
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [219]: """
So sánh điểm cao nhất của NGONNGU theo nhóm GT theo KT
Lưu ý: không để estimator thì mặc định là mean
estimator: count, min max sum std, mean (default)
"""
sns.barplot(x='GT',y='NGONNGU',hue='KT',data=df,errorbar=None,estimator=max)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

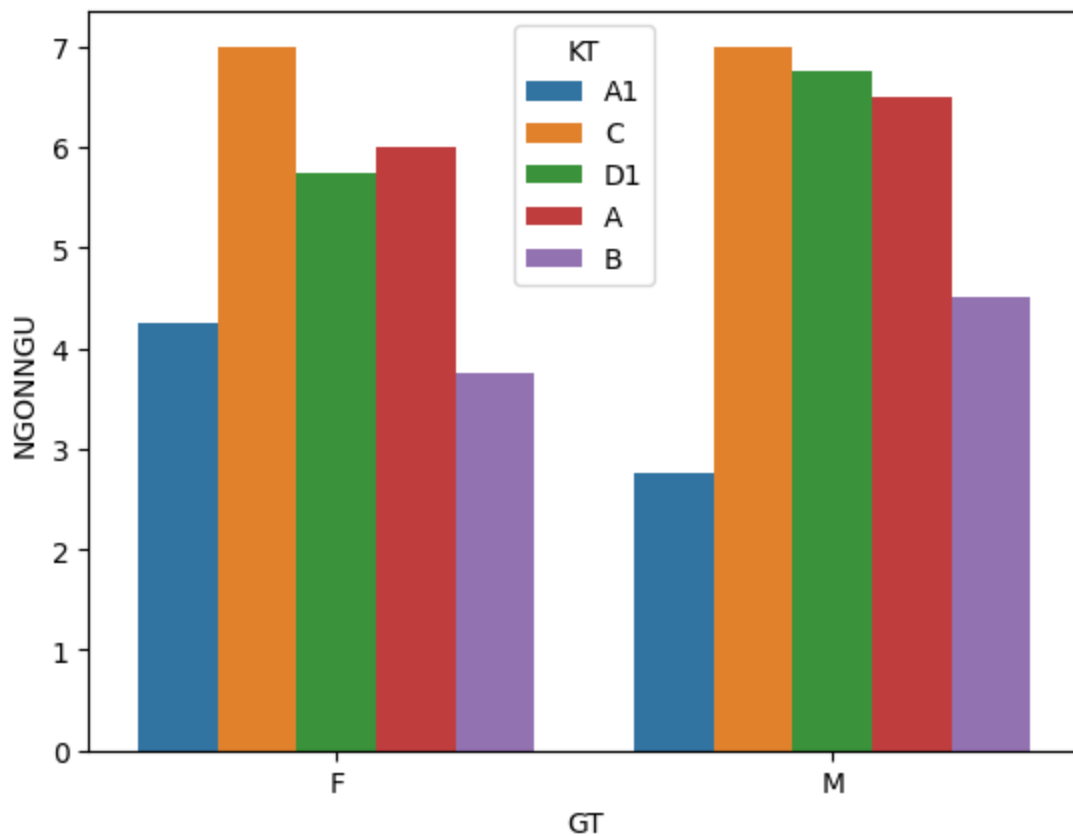
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [220]: """
So sánh điểm cao nhất của NGONNGU theo nhóm GT theo KT
Lưu ý: không để estimator thì mặc định là mean
estimator: count, min max sum std, mean (default)
"""
sns.barplot(x='GT',y='NGONNGU',hue='KT',data=df,errorbar=None,estimator=max)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

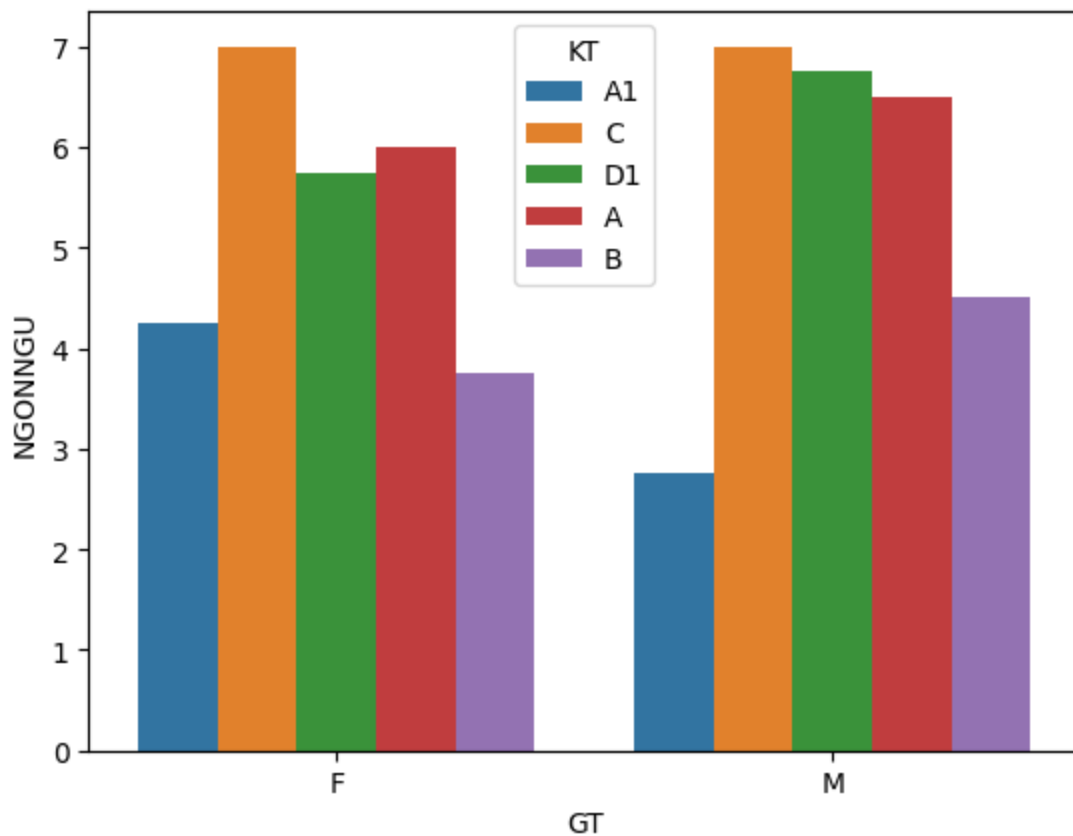
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [221]: """
Khi biến định tính dùng làm nhóm tổng hợp có nhiều hơn 2 giá trị thì ta cần dùng
hàm tổng hợp thông qua thư viện numpy
"""
sns.barplot(x='KV',y='NGONNGU',hue='KT',data=df,errorbar=None,estimator=np.max)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

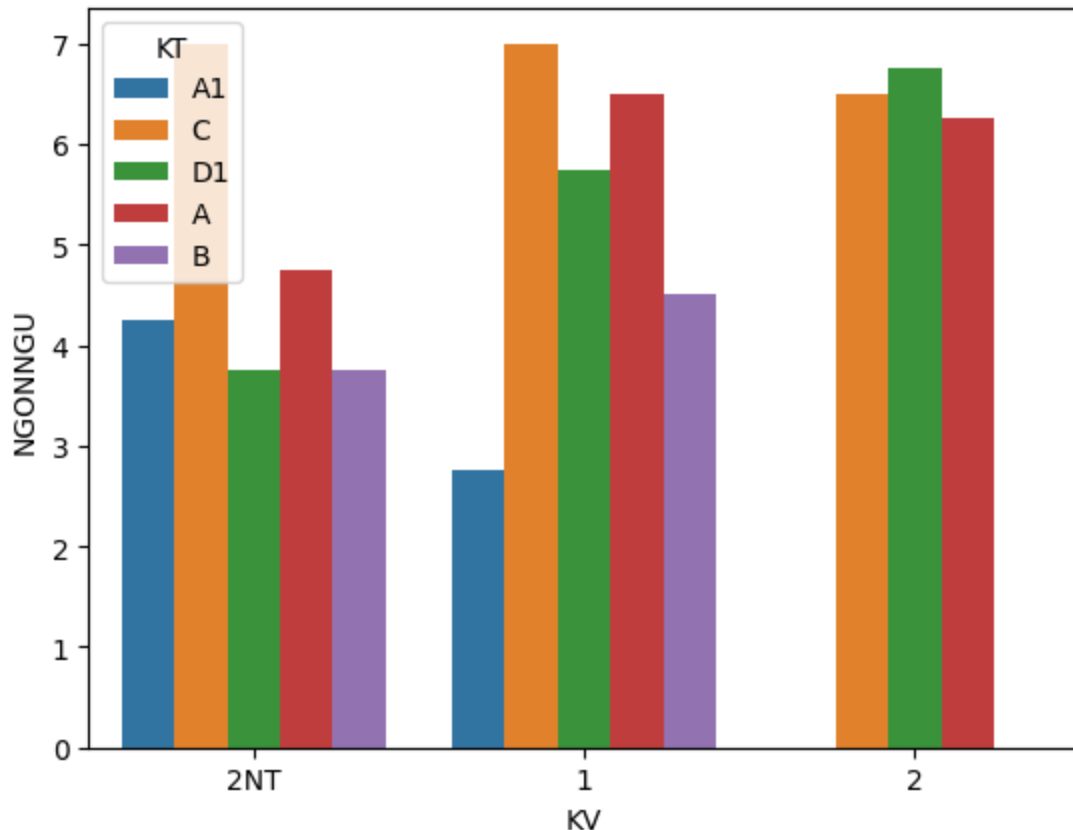
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

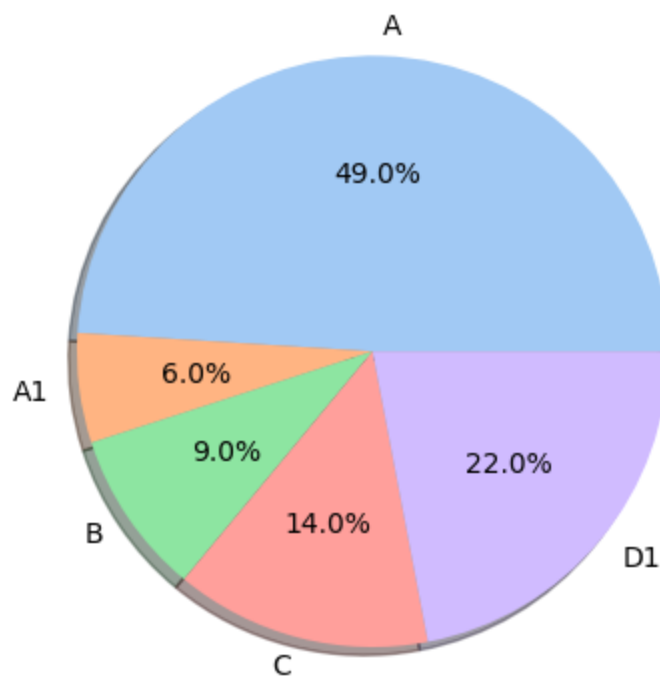


```
In [222]: """
Lưu ý:
- Với biến định tính thì ta chỉ có 1 hàm tổng hợp là hàm COUNT, MODE
- Với định lượng thì ta có thể sử dụng các hàm tổng hợp như: COUNT, MAX, MIN,
MEAN, MEDIAN, MODE, SUM, STD
"""
```

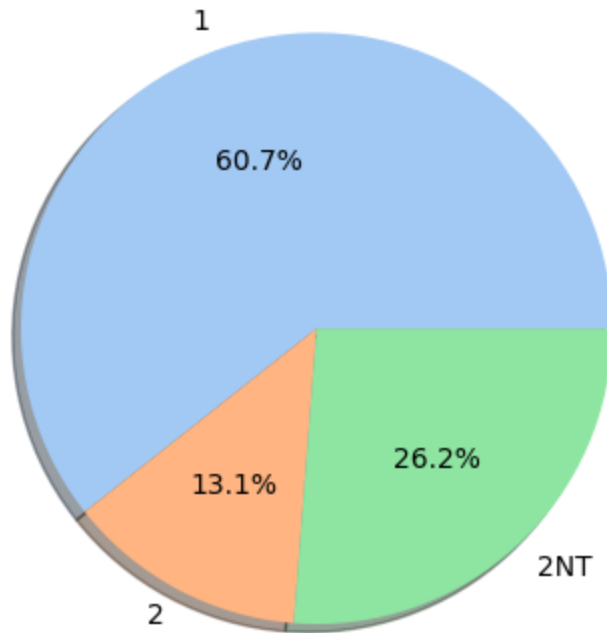
```
Out[222]: '\nLưu ý:\n- Với biến định tính thì ta chỉ có 1 hàm tổng hợp là hàm COUNT, MODE\n- Với định lượng thì ta có thể sử dụng các hàm tổng hợp như: COUNT, MAX, MIN,\nMEAN, MEDIAN, MODE, SUM, STD\n'
```

```
In [223]: """
Biểu đồ PIE
Mục đích: Trực quan dữ liệu theo nhóm tỉ lệ phần trăm
"""
gb = df.groupby(['KT'])['KT'].agg(['count'])
# group by trên nhóm khối thi trên biến khối thi và dùng hàm count
```

```
In [224]: """
Biểu đồ PIE
Mục đích: Trực quan dữ liệu theo nhóm tỉ lệ phần trăm
"""
gb = df.groupby(['KT'])['KT'].agg(['count'])
# group by trên nhóm khối thi trên biến khối thi và dùng hàm count
labels = gb.index
data = list(gb['count'])
colors = sns.color_palette('pastel') # tạo bảng màu
plt.pie(data, labels=labels, colors=colors, autopct='%1.1f%%', shadow=True)
plt.show()
```



```
In [225]: """
Trực quan tỉ lệ % tổng điểm CONG trên từng nhóm khu vực
# coi khu vực nào dc cộng điểm nhiều nhất
"""
gb = df.groupby(['KV'])['CONG'].agg(['sum'])
labels = gb.index
data = list(gb['sum'])
colors = sns.color_palette('pastel') # tạo bảng màu
plt.pie(data, labels=labels, colors=colors, autopct='%1.1f%%', shadow=True)
plt.show()
```



```
In [226]: """
Khi trực quan dữ liệu ta cần lưu ý đến loại biến đang tham gia vào trực quan
Thông thường việc chọn lựa biểu đồ sẽ căn cứ dựa trên ý nghĩa nghiệp vụ và sự
phối hợp giữa các loại biến như:
- Định tính kết hợp định tính
- Định tính kết hợp định lượng
- Định lượng kết hợp định lượng
"""
```

```
Out[226]: "\nKhi trực quan dữ liệu ta cần lưu ý đến loại biến đang tham gia vào trực quan\nThông thường việc chọn l
ựa biểu đồ sẽ căn cứ dựa trên ý nghĩa nghiệp vụ và sự\nphối hợp giữa các loại biến như:\n- Định tính kết h
ợp định tính\n- Định tính kết hợp định lượng\n- Định lượng kết hợp định lượng\n'
```

```
In [227]: """
Biểu đồ line thường dùng để tổng hợp dữ liệu theo trục "Thời gian" hoặc "có thứ
tự"
Ví dụ: tổng hợp trung bình điểm cộng theo các năm thi
"""
sns.lineplot(x='NGAYTHI',y='CONG',data=df)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

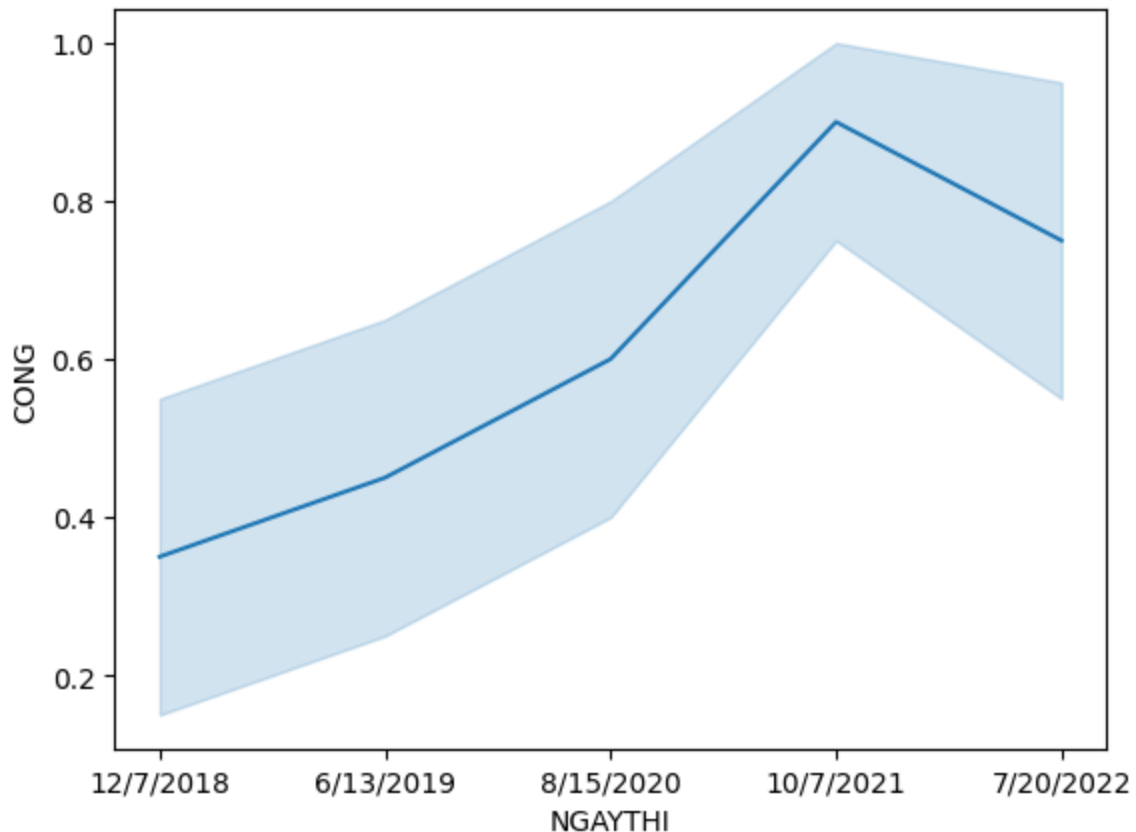
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option\_context('mode.use\_inf\_as\_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option\_context('mode.use\_inf\_as\_na', True):



```
In [228]: # Trực quan dữ liệu tổng điểm CONG dựa theo năm bằng biểu đồ line
sns.lineplot(x='NGAYTHI',y='CONG',data=df,estimator=sum)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

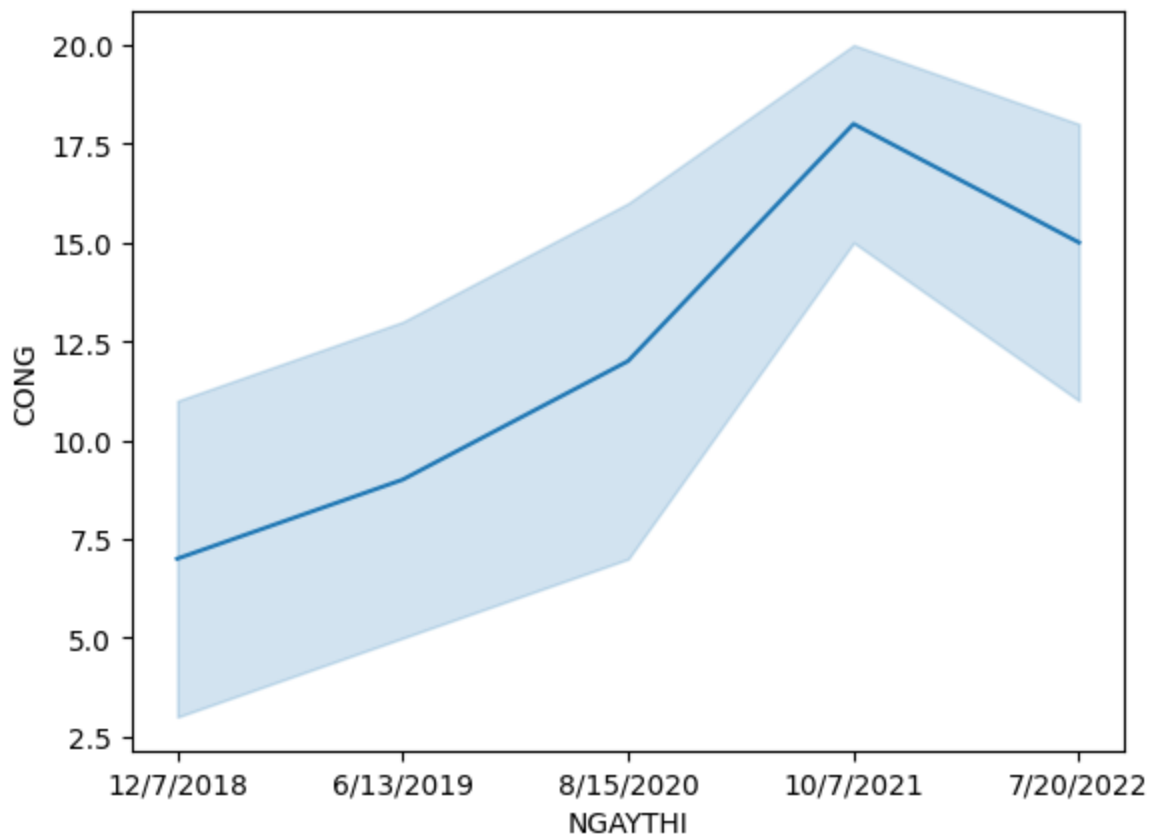
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option\_context('mode.use\_inf\_as\_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option\_context('mode.use\_inf\_as\_na', True):





```
In [229]: # tổng hợp tổng điểm cộng theo các năm thitrên từng nhóm giới tính bằng biểu đồ line
sns.lineplot(x='NGAYTHI',y='CONG',hue='GT',data=df,estimator=sum)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

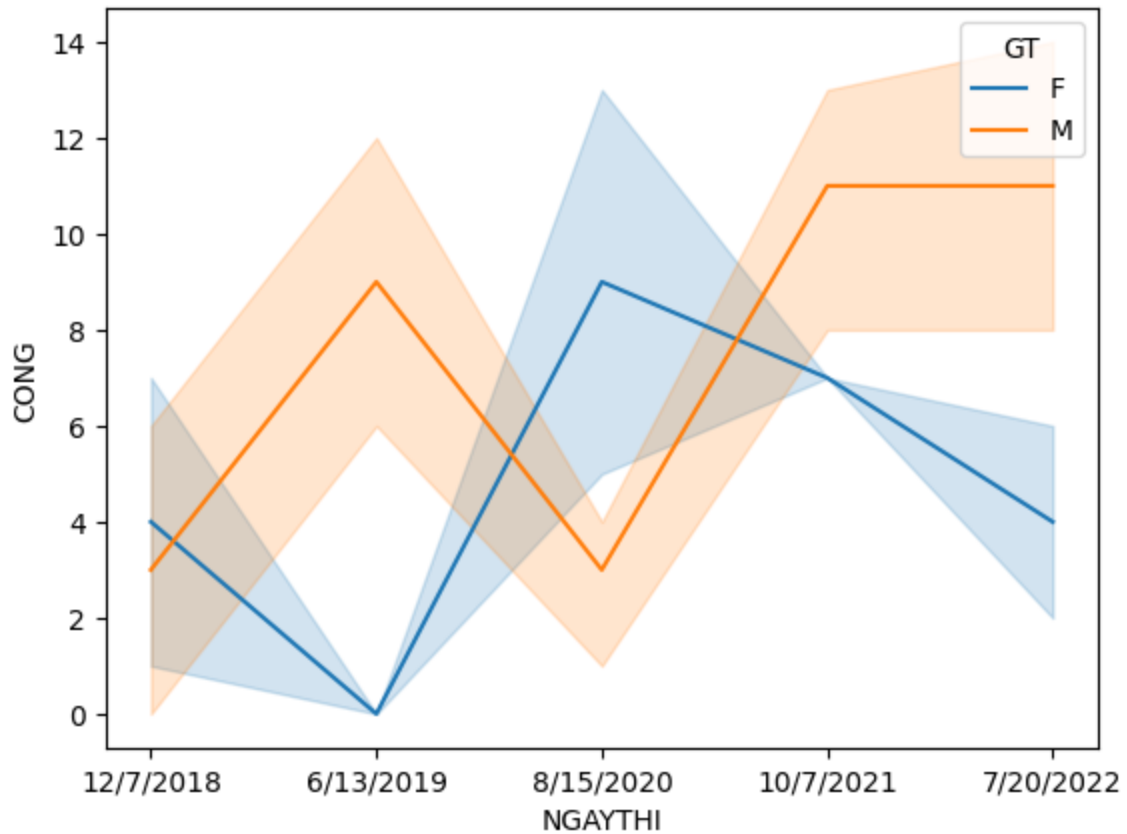
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option\_context('mode.use\_inf\_as\_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option\_context('mode.use\_inf\_as\_na', True):



In [230]: *# Giải thích ý nghĩa các đại lượng*

'''  
 Độ lệch chuẩn (std) bằng căn bậc 2 giá trị phương sai, độ lệch chuẩn và phương sai thể hiện mức độ biến thiên, biến động  
 '''

'''  
 sự đa dạng của tập dữ liệu số. Độ lệch chuẩn càng cao thì tập dữ liệu biến động mạnh => mức độ đa dạng nhiều và ngược lại thì tập dữ liệu sẽ ổn định hơn  
 '''

*# Từ phân vị*

*# Q1 : 25% -> Có 25% dữ liệu nhỏ hơn giá trị Q1*

'''  
 Q2: 50% (median trung vị) -> giá trị này cho biết có 50% sv nhỏ hơn Q2 và lớn hơn Q2  
 '''

*# Q3: 75% -> có 25% số lượng lớn hơn Q3*

'''  
 Q1 - Q3 là khoảng IQR: khoảng mà các dữ liệu được diễn ra dc coi là thông thường (50%)  
 '''

Out[230]: '\nQ1 - Q3 là khoảng IQR: khoảng mà các dữ liệu được diễn ra dc coi là thông\nthường (50%)\n'

In [231]: `df['NGONNGU'].describe()`

```
Out[231]: count    100.000000
          mean      3.740000
          std       1.424408
          min       1.000000
          25%       2.500000
          50%       3.625000
          75%       4.750000
          max       7.000000
          Name: NGONNGU, dtype: float64
```

In [232]: `df[['NGONNGU','LOGIC','UNGXU']].describe()`

```
Out[232]:
```

	NGONNGU	LOGIC	UNGXU
<b>count</b>	100.000000	100.000000	100.000000
<b>mean</b>	3.740000	4.255000	4.875000
<b>std</b>	1.424408	1.060053	1.041136
<b>min</b>	1.000000	1.500000	2.750000
<b>25%</b>	2.500000	3.500000	4.250000
<b>50%</b>	3.625000	4.125000	4.750000
<b>75%</b>	4.750000	4.750000	5.500000
<b>max</b>	7.000000	8.000000	8.000000

```
In [233]: """
          CV = std/mean (Coefficient of variant)
          So sánh mức độ ổn định của điểm số
          """
          cvNN = df['NGONNGU'].std()/df['NGONNGU'].mean()
          cvLogic = df['LOGIC'].std()/df['LOGIC'].mean()
          cvUngXu = df['UNGXU'].std()/df['UNGXU'].mean()
          print('cvNN: ', cvNN)
          print('cvLogic: ', cvLogic)
          print('cvUngXu: ', cvUngXu)
          # df[['NGONNGU','LOGIC','UNGXU']].std()/df[['NGONNGU','LOGIC','UNGXU']].mean()

          cvNN: 0.3808578707188297
          cvLogic: 0.2491311130540305
          cvUngXu: 0.2135664058628459
```

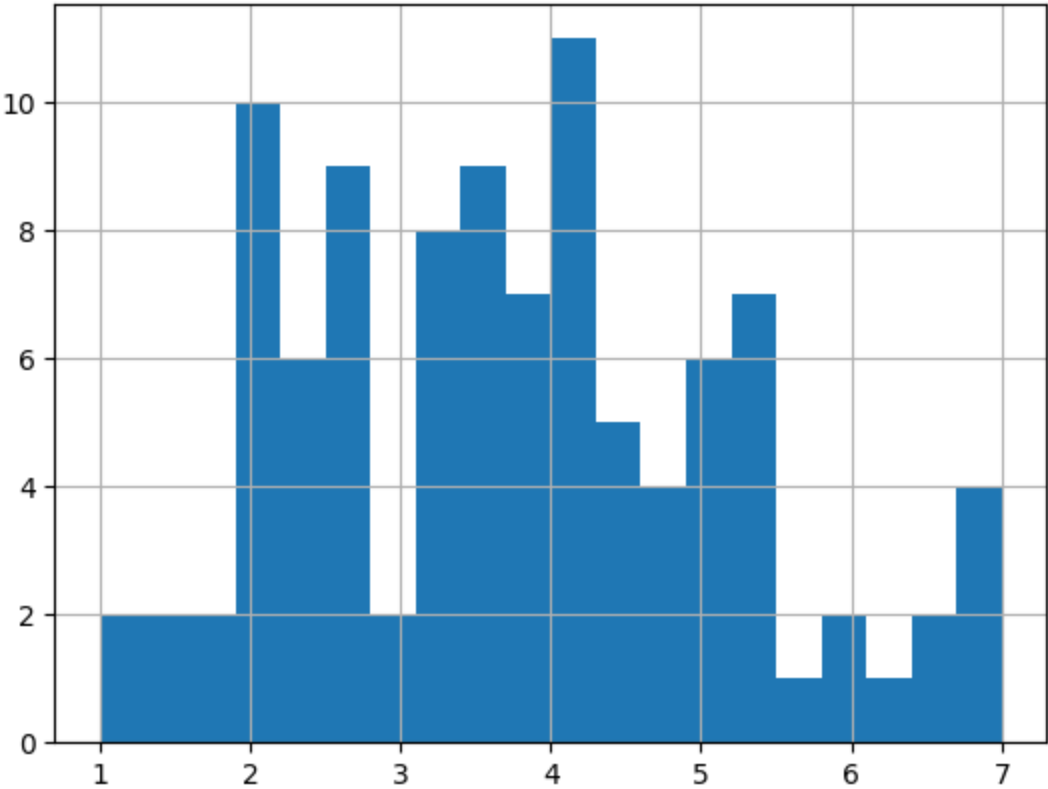
```
In [234]: df.groupby('GT')[['NGONNGU','LOGIC','UNGXU']].describe()
```

Out[234]:

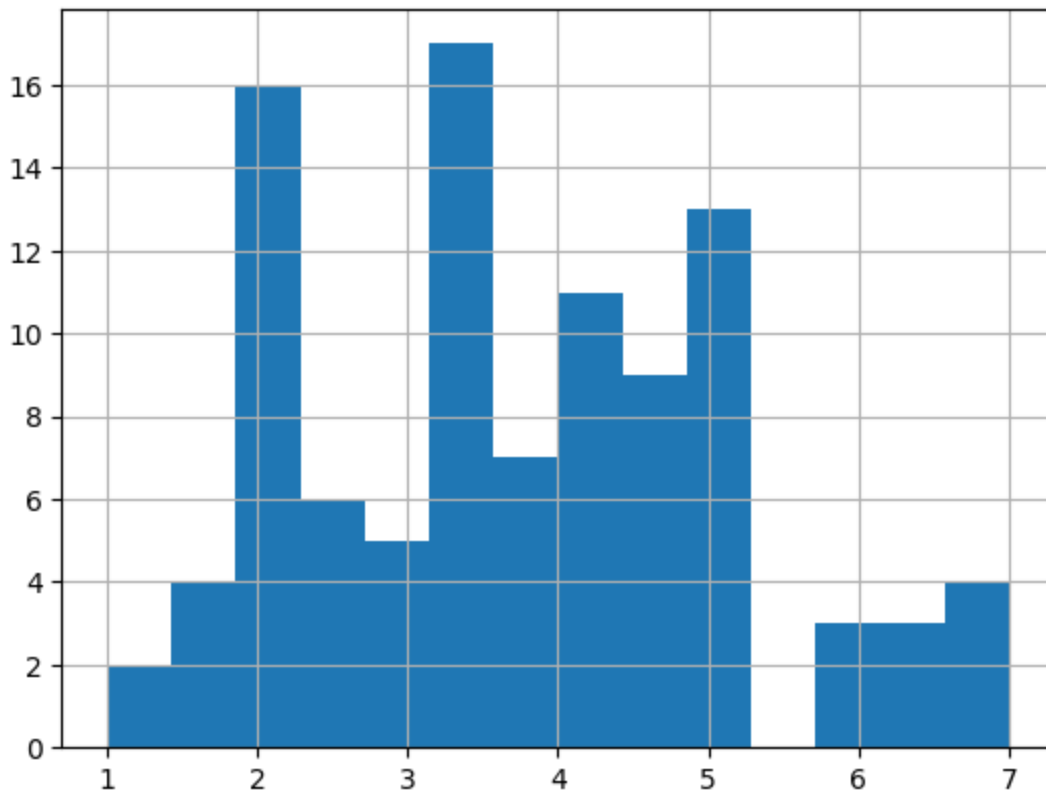
	NGONNGU								LOGIC							
	count	mean	std	min	25%	50%	75%	max	count	mean	...	75%	max	c		
GT																
F	48.0	3.854167	1.348594	1.25	3.1875	3.75	5.0	7.0	48.0	4.046875	...	4.5	6.75			
M	52.0	3.634615	1.496288	1.00	2.4375	3.50	4.5	7.0	52.0	4.447115	...	5.0	8.00			

2 rows × 24 columns

```
In [235]: """
Histogram cho biết xác suất xảy ra của biến cố trong khoảng giá trị dữ liệu nào
nhiều nhất
"""
df['NGONNGU'].hist(bins=20)
plt.show()
```

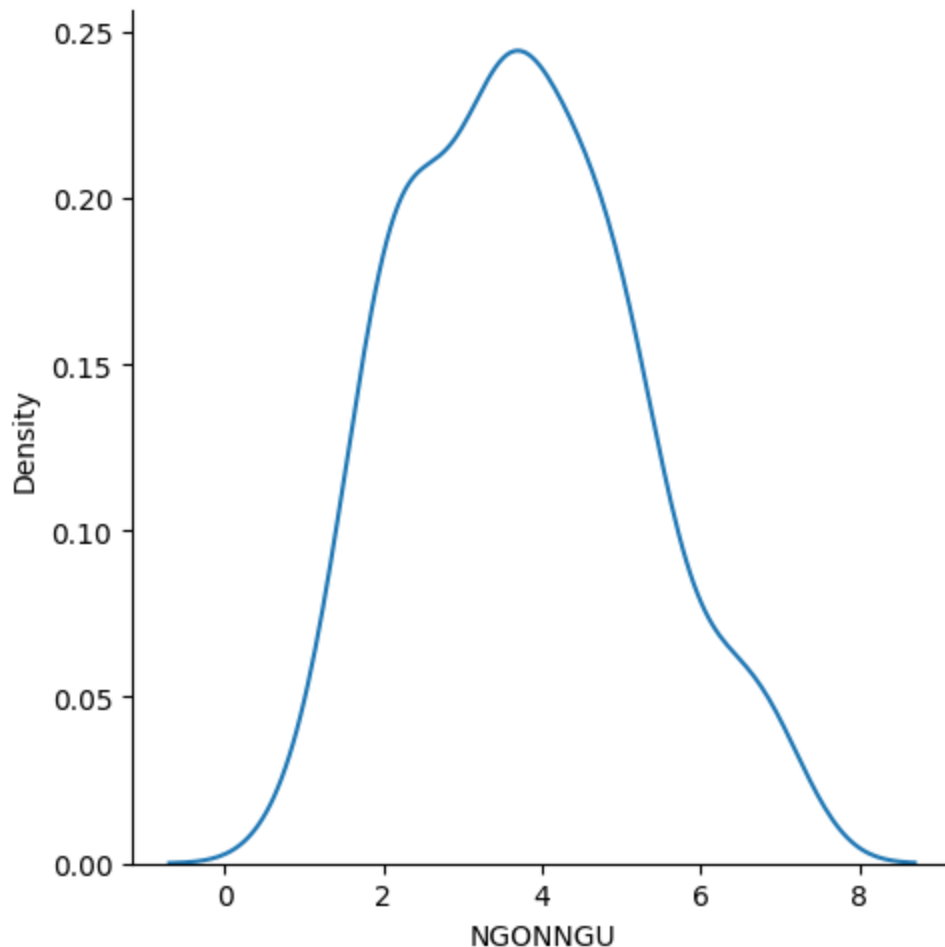


```
In [236]: # Hướng dẫn khi vẽ bins trong histogram  
# Lưu ý: khi số lượng bins khác nhau sẽ dẫn đến hình dạng của histogram khác nhau  
df['NGONNGU'].hist(bins=14)  
plt.show()
```



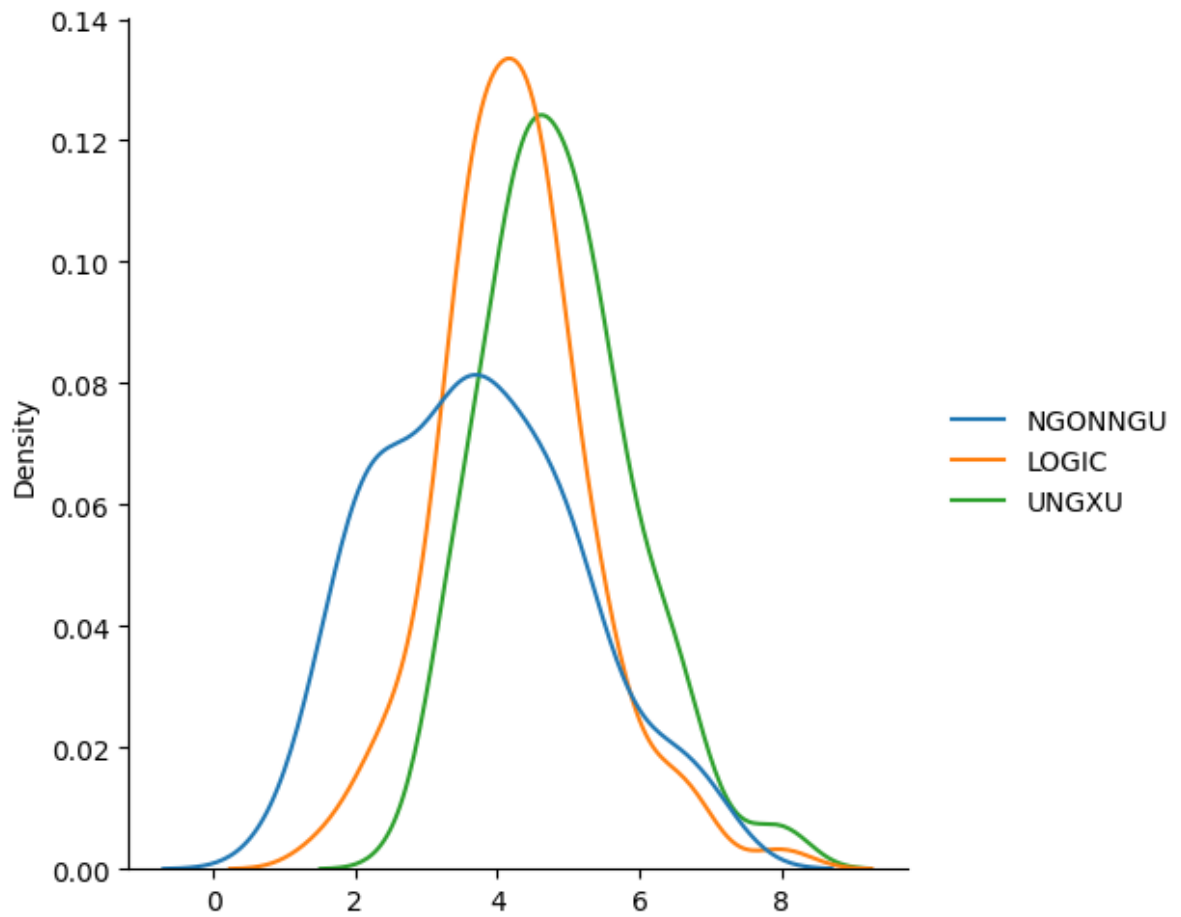
```
In [237]: """  
          Nâng cao hơn histogram thì ra khám phá dạng phân phối xác suất  
          Làm mịn với phân phối xác suất  
          """  
          sns.displot(df,x='NGONNGU',kind='kde')  
          plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.  
 with pd.option\_context('mode.use\_inf\_as\_na', True):  
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight  
 self.figure.tight\_layout(\*args, \*\*kwargs)



```
In [238]: sns.displot(data= df[['NGONNGU','LOGIC','UNGXU']],kind='kde')
plt.show()
# Hãy cho biết phân phối của biến số nào gần với phân phối chuẩn hơn => logic với ứng xử
```

```
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self.figure.tight_layout(*args, **kwargs)
```





```
In [239]: sns.displot(df,x='NGONNGU',hue='GT',kind='kde')
plt.show()
# Câu hỏi: nhóm giới tính nào gần hơn: F gần hơn
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

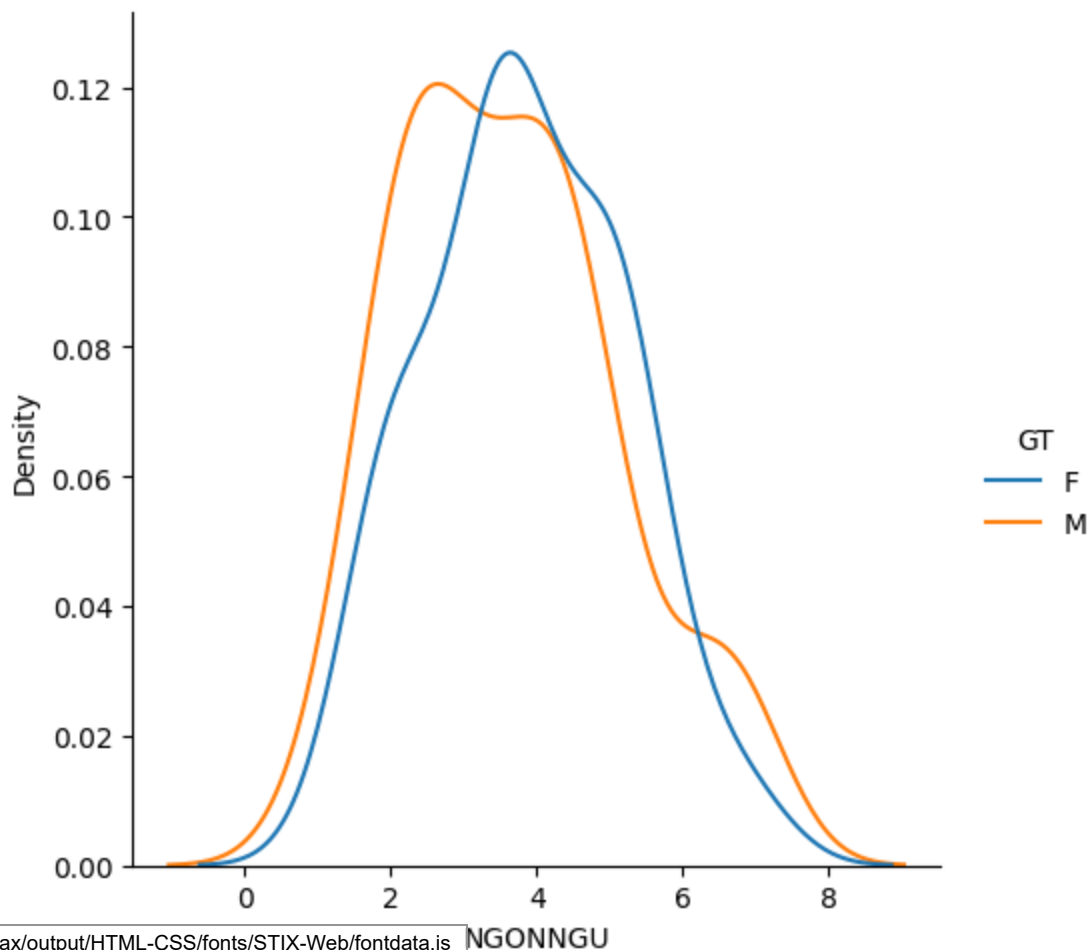
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1119: FutureWarning: use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

with pd.option\_context('mode.use\_inf\_as\_na', True):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight

self.\_figure.tight\_layout(\*args, \*\*kwargs)



```
In [240]: """
    Đây là biểu đồ quan trọng trong việc phân tích dữ liệu định lượng
    Biểu đồ này cung cấp các thông tin quan trọng như
    1. Q1: Tứ phân vị 25%
    2. Q2: Tứ phân vị 50% (median)
    3. Q3: Tứ phân vị 75%
    4. Độ lớn IQR = |Q3-Q1|
    5. Lower bound:  $Q1 - 1.5 * IQR$ 
    6. Upper bound =  $Q3 + 1.5 * IQR$ 
    7. Các ngoại biên, bất thường (outlier) cần xử lý trong dữ liệu
    Outlier: là điểm dữ liệu khác biệt quá nhiều so với đa số
    Hướng dẫn
    + Tính khoảng nghi ngờ chứa outliers
    + Tính khoảng chắc chắn chứa outliers
    """
```

```
Out[240]: '\nĐây là biểu đồ quan trọng trong việc phân tích dữ liệu định lượng\nBiểu đồ này cung cấp các thông tin
quan trọng như\n1. Q1: Tứ phân vị 25%\n2. Q2: Tứ phân vị 50% (median)\n3. Q3: Tứ phân vị 75%\n4. Đ
ộ lớn IQR = |Q3-Q1|\n5. Lower bound:  $Q1 - 1.5 * IQR$ \n6. Upper bound =  $Q3 + 1.5 * IQR$ \n7. Các ngoại biê
n, bất thường (outlier) cần xử lý trong dữ liệu\nOutlier: là điểm dữ liệu khác biệt quá nhiều so với đa số\n
Hướng dẫn\n+ Tính khoảng nghi ngờ chứa outliers\n+ Tính khoảng chắc chắn chứa outliers\n'
```

```
In [241]: sns.boxplot(data=df['LOGIC'],orient="h")
print('lower bound = ', df['LOGIC'].quantile(0.25) -
1.5*(df['LOGIC'].quantile(0.75) - df['LOGIC'].quantile(0.25)))
print('upper bound = ', df['LOGIC'].quantile(0.75) +
1.5*(df['LOGIC'].quantile(0.75) - df['LOGIC'].quantile(0.25)))
IQR = df['LOGIC'].quantile(0.75) - df['LOGIC'].quantile(0.25)
print('IQR',IQR)
print('ngoai bien duoi', 1.625 -1.5*IQR)
print('ngoai bien tren', 6.625 +1.5*IQR)
```

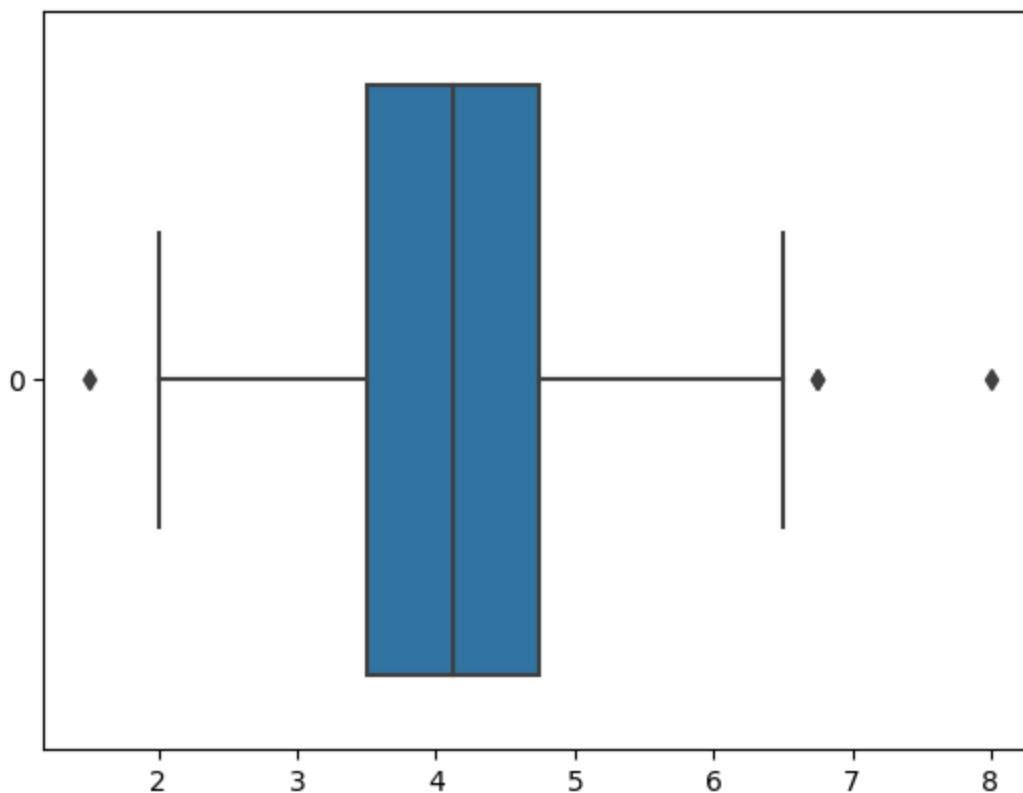
lower bound = 1.625

upper bound = 6.625

IQR 1.25

ngoai bien duoi -0.25

ngoai bien tren 8.5



```
In [242]: # Tính khoảng giá trị nghi ngờ bất thường
# Tính khoảng giá trị được cho là bất thường
# Tính xem có bao nhiêu sinh viên có điểm thi là bất thường
sns.boxplot(data=df[['NGONNGU','LOGIC','UNGXU']],orient='h')
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

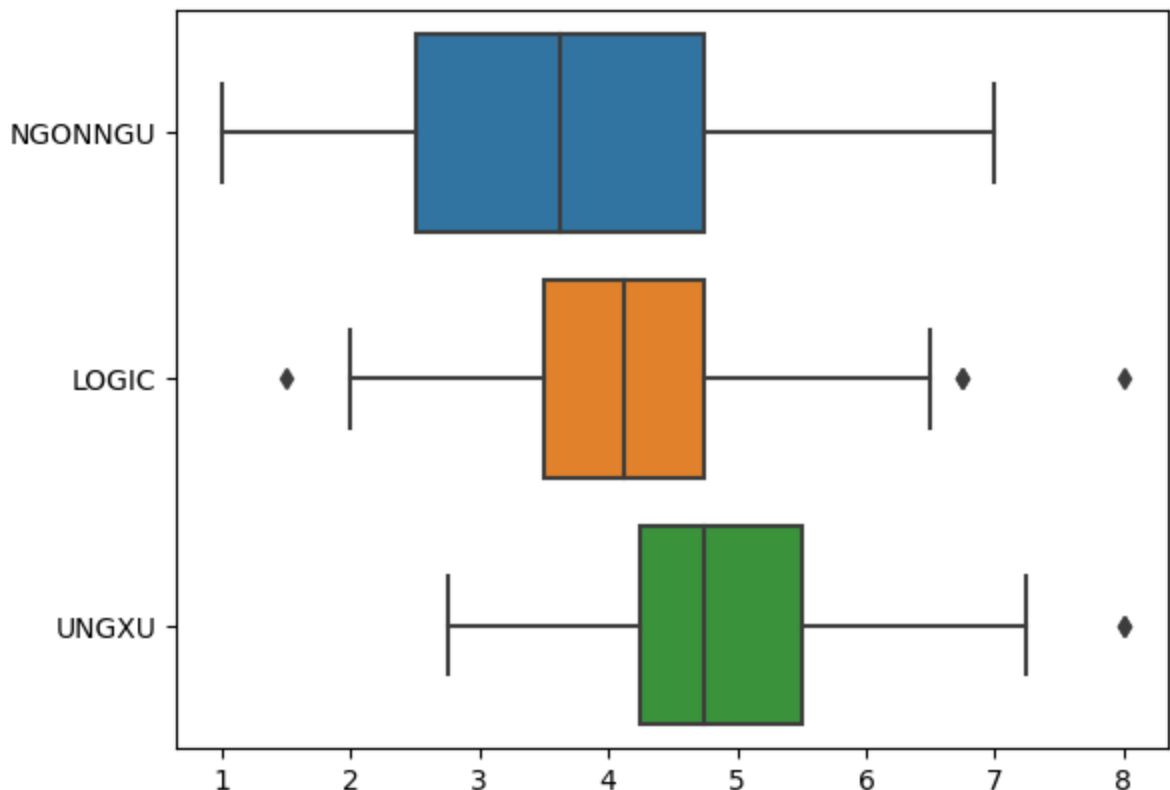
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

Out[242]: <Axes: >



```
In [243]: # Hãy cho biết điểm số môn nào không xảy ra bất thường => NGONNGU
sns.boxplot(x='NGONNGU',y='KT',data=df,orient='h')
plt.show()
# Câu hỏi: khối thi nào có lower bound trùng với phân vị thứ 1 (Q1) => khối B
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

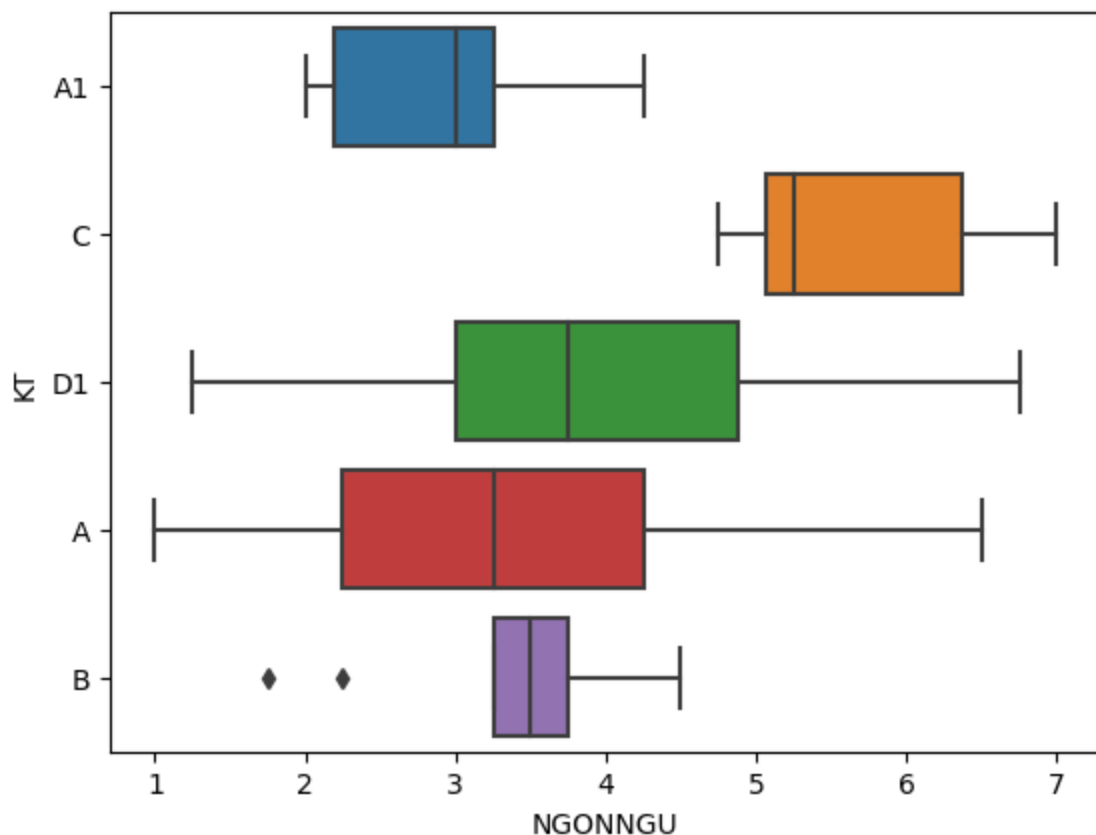
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



```
In [244]: sns.boxplot(x='NGONNGU',y='KV',data=df,orient='h')
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

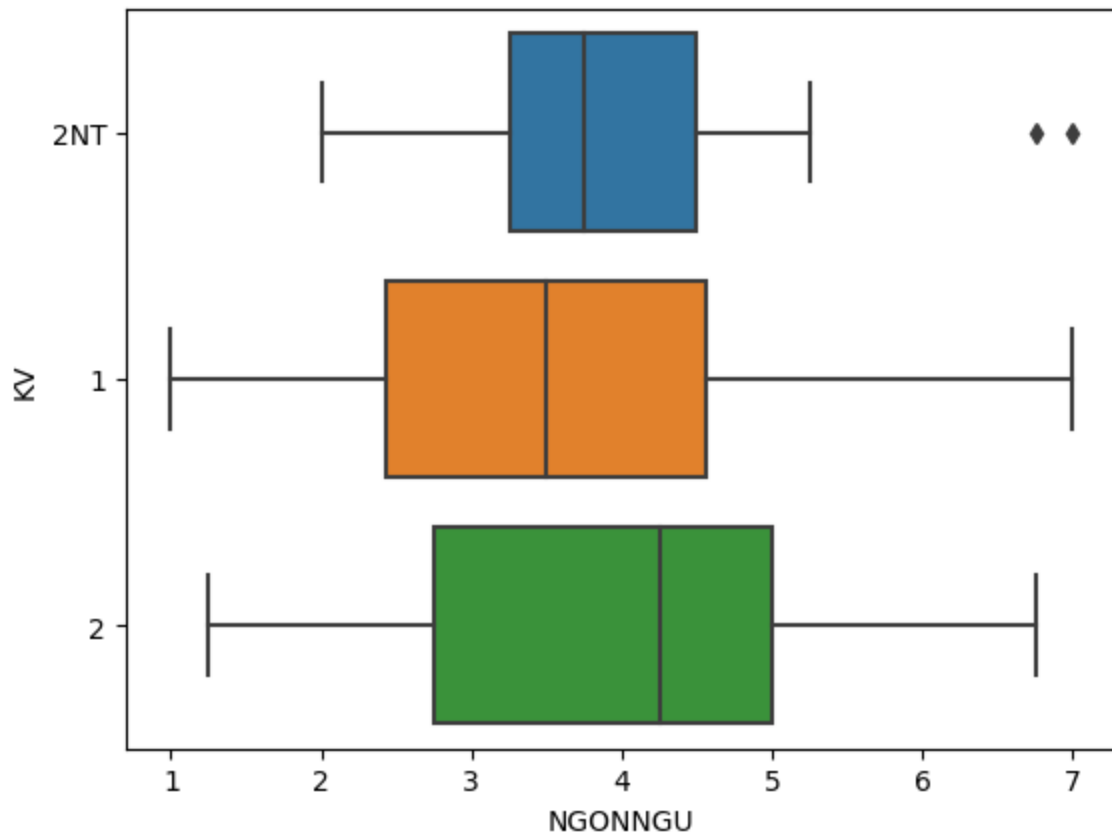
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):



In [245]: `sns.boxplot(x='KT',y='NGONNGU',hue='GT',data=df)`

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

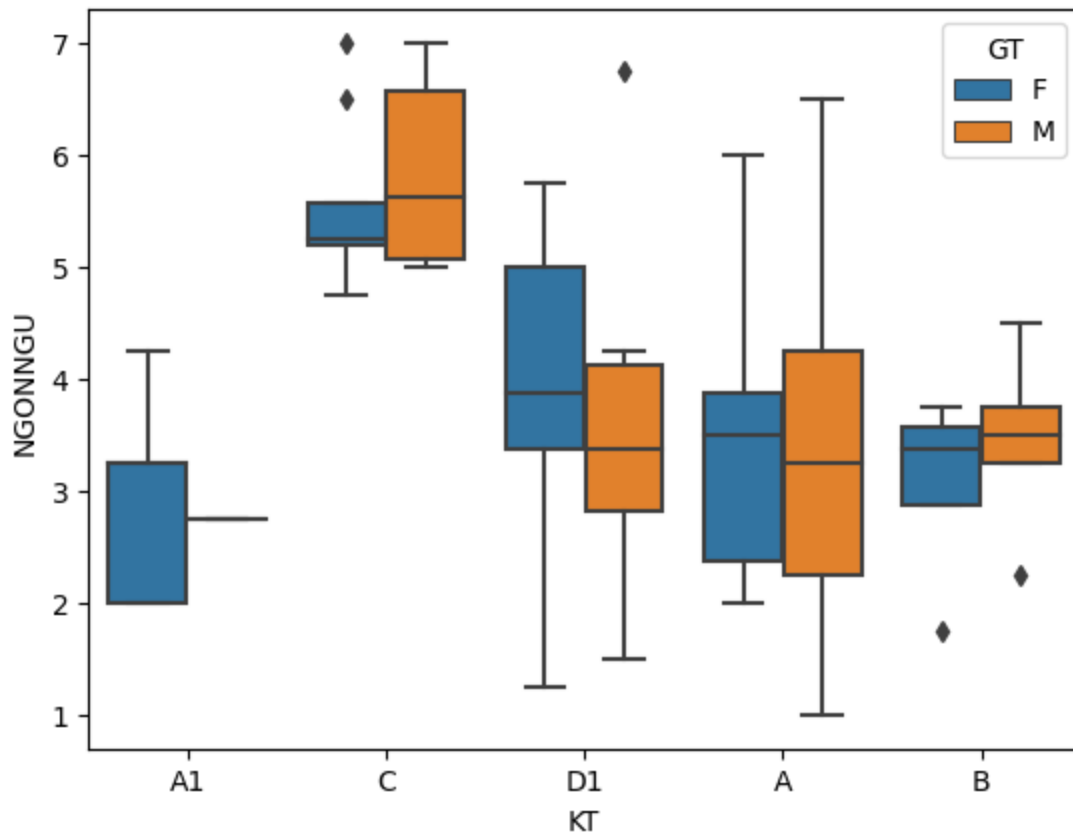
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

Out[245]: <Axes: xlabel='KT', ylabel='NGONNGU'>



```
In [246]: sns.boxplot(x='KT',y='NGONNGU',hue='GT',data=df)
plt.show()
# câu hỏi: xác định các biểu đồ bất thường
# Khối A1 không đủ dữ liệu để vẽ
```

e:\LAP TRÌNH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRÌNH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

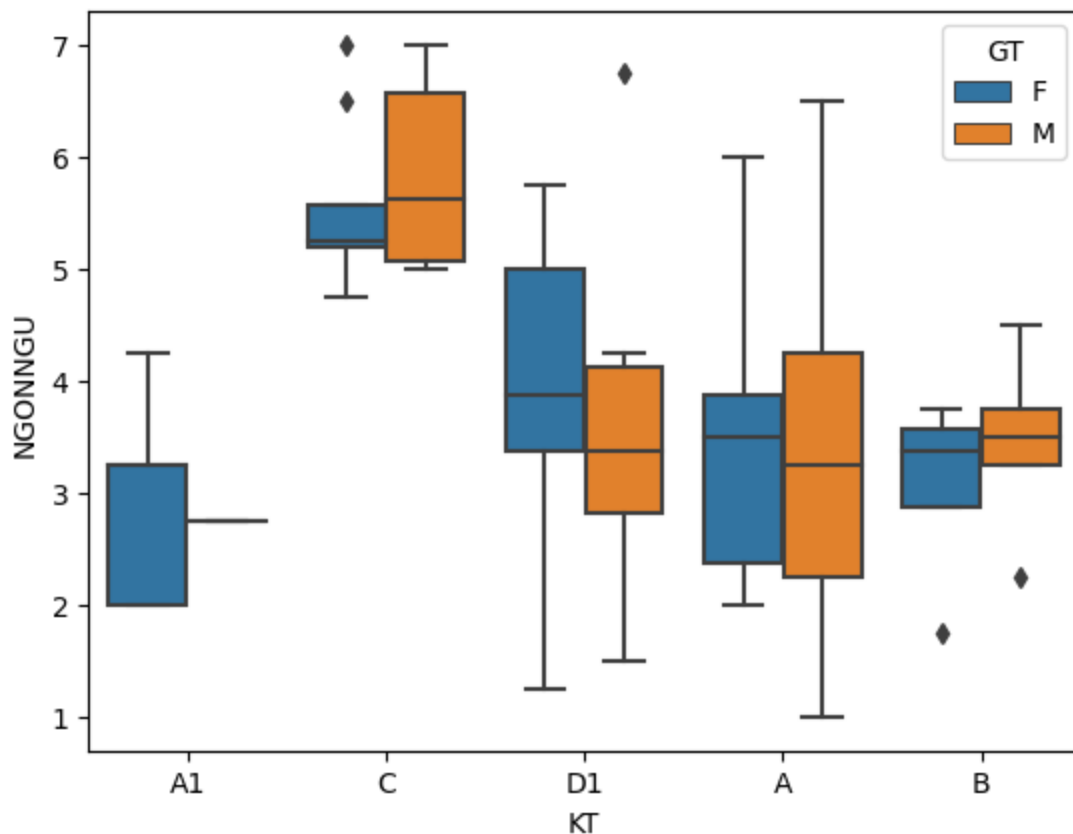
if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRÌNH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRÌNH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):





In [247]:

```
"""
Skewness = độ xiên, độ lớn (trị tuyệt đối) cho biết mức độ dữ liệu lệch nhiều hay
ít so với đường cong phân phối chuẩn
Cho biết xác suất được phân bố lệch về phía nào nhiều
Trị tuyệt đối giá trị càng lớn thì dữ liệu phân phối nghiêng càng nhiều (lệch)
Diễn giải cho skewness
skewness > 0 tức là mean > median: ta gọi là positive skewness
hay lệch phải, tức là giá trị ngoại biên outliers nhận giá trị lớn sẽ đẩy giá trị
trung bình về phía cuối
skewness < 0 tức là mean < median: ta gọi là negative skewness hay lệch trái, tức
là giá trị outliers nhận giá trị nhỏ sẽ đẩy giá trị trung
bình về phía đầu
skewness = 0 tức là mean = median = mode: phân phối không lệch còn được gọi là
phân phối đối xứng
"""
```

Out[247]: \nSkewness = độ xiên, độ lớn (trị tuyệt đối) cho biết mức độ dữ liệu lệch nhiều hay ít so với đường cong phân phối chuẩn\nCho biết xác suất được phân bố lệch về phía nào nhiều\nTrị tuyệt đối giá trị càng lớn thì dữ liệu phân phối nghiêng càng nhiều (lệch)\nDiễn giải cho skewness\nskewness > 0 tức là mean > median: ta gọi là positive skewness\nhay lệch phải, tức là giá trị ngoại biên outliers nhận giá trị lớn sẽ đẩy giá trị trung bình về phía cuối\nskewness < 0 tức là mean < median: ta gọi là negative skewness hay lệch trái, tức là giá trị outliers nhận giá trị nhỏ sẽ đẩy giá trị trung bình về phía đầu\nskewness = 0 tức là mean = median = mode: phân phối không lệch còn được gọi là phân phối đối xứng\n'

In [248]:

```
df['NGONNGU'].skew()
```

Out[248]: 0.33322084231625376

In [249]:

```
"""
Note: Khi ptich dữ liệu với các phương pháp có liên quan phân phối chuẩn thì cần
kiểm tra skewness
nếu dl quá lệch so với phân phối chuẩn thì ta cần điều chỉnh bằng các hàm
transform cho bớt lệch
đặc biệt là phân tích hồi quy
"""
```

Out[249]: \nNote: Khi ptich dữ liệu với các phương pháp có liên quan phân phối chuẩn thì cần kiểm tra skewness\nnếu dl quá lệch so với phân phối chuẩn thì ta cần điều chỉnh bằng các hàm transform cho bớt lệch\nđặc biệt là phân tích hồi quy\n'

In [250]:

```
df[['NGONNGU','LOGIC','UNGXU']].skew()
```

Out[250]: NGONNGU 0.333221  
LOGIC 0.472551  
UNGXU 0.607063  
dtype: float64

```
In [251]: """
Kurtosis: Độ nhọn, trị tuyệt đối cho biết mức độ nhọn của phân phối
Độ lớn của kurtosis càng gần 3 thì fit
Dưới 3 thì fat
Trên 3 thì thin
Thông thường để đánh giá hình dáng độ nhọn ta dùng đại lượng excess kurtosis
(theo Pearson) - 3
+ Nếu excess > 0 -> thin
+ Nếu excess = 0 -> fit
+ Nếu excess < 0 -> fat
Trong pandas sử dụng Fisher's Kurtosis tức là đã chuẩn hóa giá trị về excess
kurtosis theo mean = 0
+ Trị tuyệt đối excess kurtosis càng cao thì mức độ thin, fat càng lớn
Lưu ý
"""
```

```
Out[251]: """Kurtosis: Độ nhọn, trị tuyệt đối cho biết mức độ nhọn của phân phối
Độ lớn của kurtosis càng gần 3 thì fit
Dưới 3 thì fat
Trên 3 thì thin
Thông thường để đánh giá hình dáng độ nhọn ta dùng đại lượng excess kurtosis
(theo Pearson) - 3
+ Nếu excess > 0 -> thin
+ Nếu excess = 0 -> fit
+ Nếu excess < 0 -> fat
Trong pandas sử dụng Fisher's Kurtosis tức là đã chuẩn hóa giá trị về excess kurtosis theo mean = 0
Trị tuyệt đối excess kurtosis càng cao thì mức độ thin, fat càng lớn
Lưu ý
"""
```

```
In [252]: df[['NGONNGU']].kurtosis()
# Câu hỏi: biến ngôn ngữ có độ nhọn như thế nào
# Trả lời là: fat
```

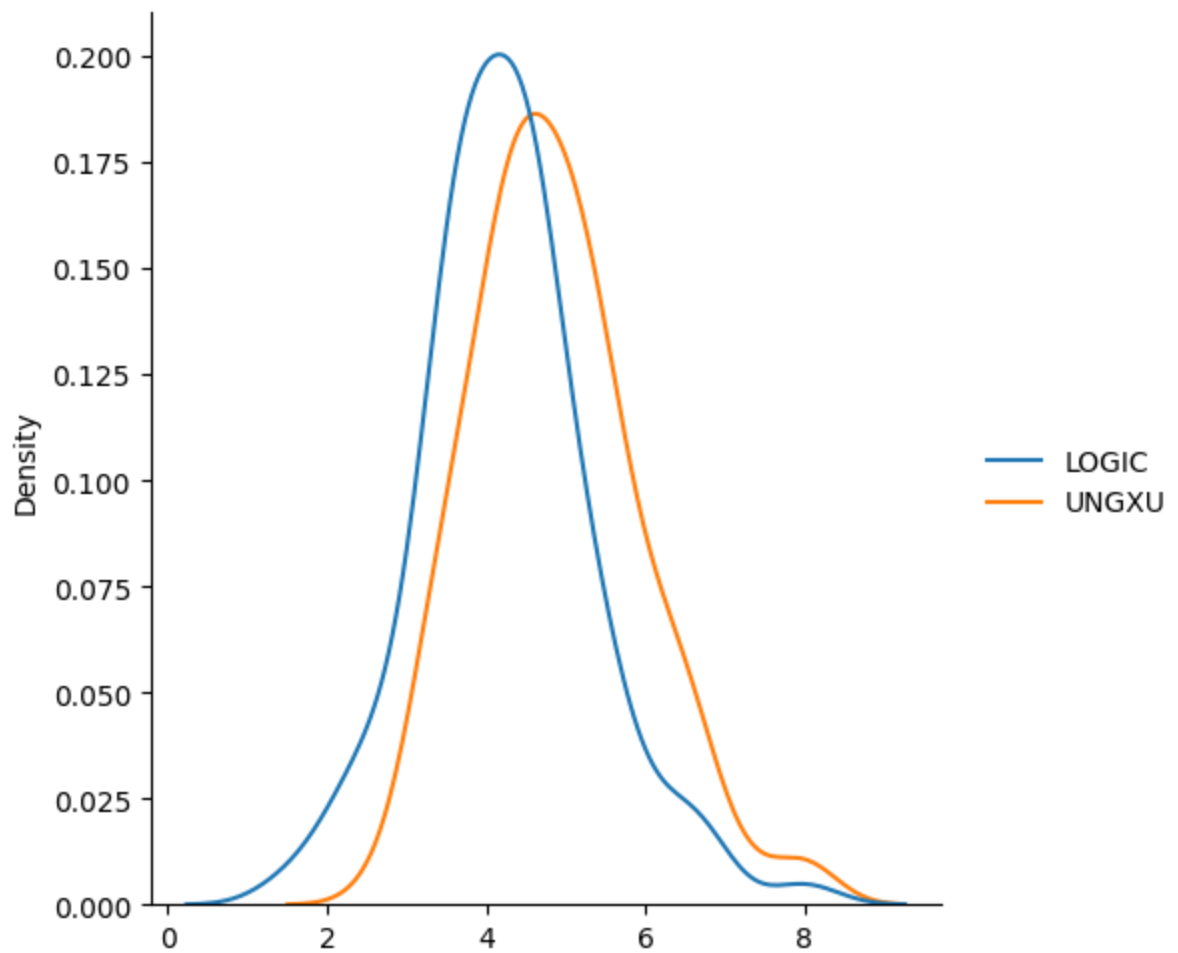
```
Out[252]: NGONNGU    -0.517004
dtype: float64
```

```
In [253]: df[['NGONNGU','LOGIC','UNGXU']].kurtosis()
```

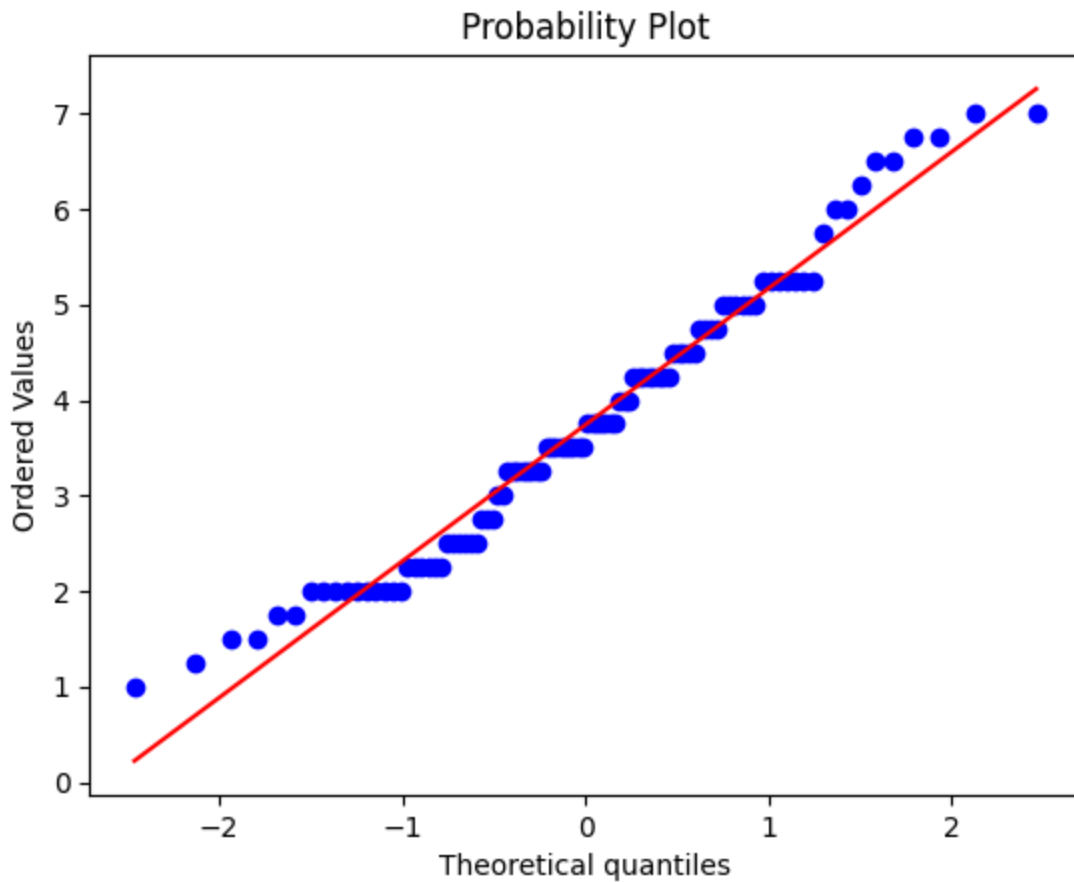
```
Out[253]: NGONNGU    -0.517004
LOGIC      1.372439
UNGXU      0.442173
dtype: float64
```

```
In [254]: sns.displot(data=df[['LOGIC','UNGXU']],kind='kde')
plt.show()
# Nhìn biểu đồ cho biết ý nghĩa kutorsis cho LOGIC UNGXU
```

```
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning:
is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, Categori
calDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning:
use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN bef
ore operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The
figure layout has changed to tight
  self.figure.tight_layout(*args, **kwargs)
```



```
In [255]: # Kiểm định phân phối chuẩn
from scipy import stats
stats.probplot(df['NGONNGU'],plot=sns.mpl.pyplot)
plt.show()
# không có phân phối chuẩn
```



```
In [256]: """
Phân tích sự tác động (ảnh hưởng) qua lại giữa 2 biến định lượng
"""
Buổi 3:
Hiệp phương sai: co-variance
Giá trị co-variance > 0 thì 2 biến có tương quan thuận (đồng biến)
Giá trị co-variance < 0 thì 2 biến có tương quan nghịch (nghịch biến)
Độ lớn (trị tuyệt đối của giá trị) càng lớn thì mức độ quan hệ (tương quan) càng
chặt chẽ
Ma trận hiệp phương sai: co-variance matrix
"""
```

```
Out[256]: "\nBuổi 3:\nHiệp phương sai: co-variance\nGiá trị co-variance > 0 thì 2 biến có tương quan thuận (đồng bi
ến)\nGiá trị co-variance < 0 thì 2 biến có tương quan nghịch (nghịch biến)\nĐộ lớn (trị tuyệt đối của giá tr
ị) càng lớn thì mức độ quan hệ (tương quan) càng\nchặt chẽ\nMa trận hiệp phương sai: co-variance matrix
\n'
```

In [257]: `df[['T5','T6']].cov()`

Out[257]:

	T5	T6
T5	2.184658	1.568961
T6	1.568961	1.858314

In [258]: `# So sánh mức độ tương quan giữa T5 so với LOGIC và T6 so với LOGIC`  
`df[['T5','T6','LOGIC']].cov()`

Out[258]:

	T5	T6	LOGIC
T5	2.184658	1.568961	0.289308
T6	1.568961	1.858314	0.435672
LOGIC	0.289308	0.435672	1.123712

In [259]: """  
 Với phương pháp so sánh tương quan bằng co-variance thì ta không đo lường được  
 cường độ  
 tương quan giữa 2 biến định lượng.  
 Pearson Correlation: tương quan tuyến tính  
 r nằm trong khoảng [-1,1]  
 r = 0 => Không tương quan  
 r < 0: Tương quan nghịch  
 r > 0: tương quan thuận  
 |r| (độ lớn) càng gần 1 thì tương quan càng cao  
 |r| < 0.5 thì tương quan thấp  
 [0.5,0.65]: Khá  
 [0.65,0.75]: Tốt  
 [0.75,0.9]: Rất tốt  
 > 0.9: hoàn hảo  
 Ma trận tương quan: correlation matrix  
 Lưu ý: được sử dụng khảo sát tương quan tuyến tính nhằm phân tích mối quan hệ  
 tuyến tính giữa 2 biến định lượng  
 """

Out[259]: "\nVới phương pháp so sánh tương quan bằng co-variance thì ta không đo lường được\ncường độ\ntương q  
 uan giữa 2 biến định lượng.\nPearson Correlation: tương quan tuyến tính\nr nằm trong khoảng [-1,1]\nr =  
 0 => Không tương quan\nr < 0: Tương quan nghịch\nr > 0: tương quan thuận\n|r| (độ lớn) càng gần 1 thì t  
 ương quan càng cao\n|r| < 0.5 thì tương quan thấp\n[0.5,0.65]: Khá\n[0.65,0.75]: Tốt\n[0.75,0.9]: Rất tốt\n  
 > 0.9: hoàn hảo\nMa trận tương quan: correlation matrix\nLưu ý: được sử dụng khảo sát tương quan tuyến  
 tính nhằm phân tích mối quan hệ\ntuyến tính giữa 2 biến định lượng\n'

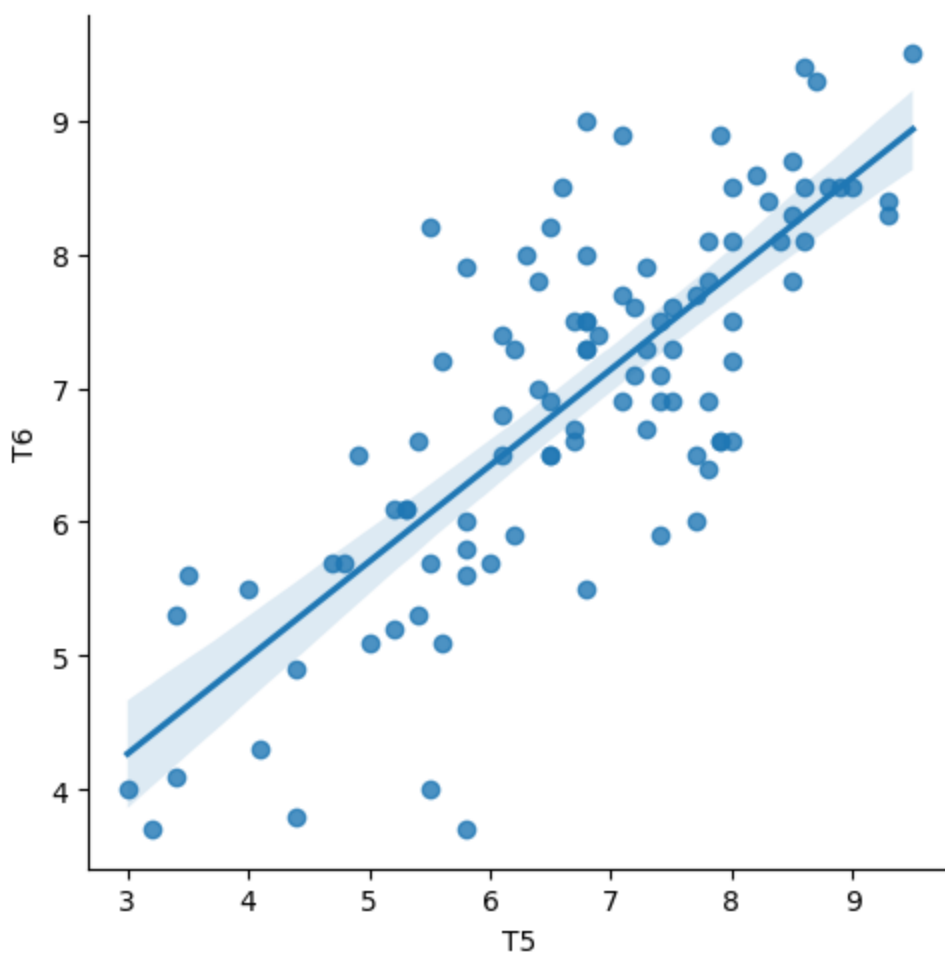
In [260]: `df[['T5','T6']].corr()`

Out[260]:

	T5	T6
T5	1.000000	0.778683
T6	0.778683	1.000000

```
In [261]: """  
Trực quan hóa tương quan tuyến tính giữa 2 biến định lượng  
Khám phá tương quan tuyến tính của 2 biến định lượng  
thông qua biểu đồ phân tán (scatter)  
"""  
sns.lmplot(data=df, x='T5', y='T6', fit_reg=True)  
plt.show()
```

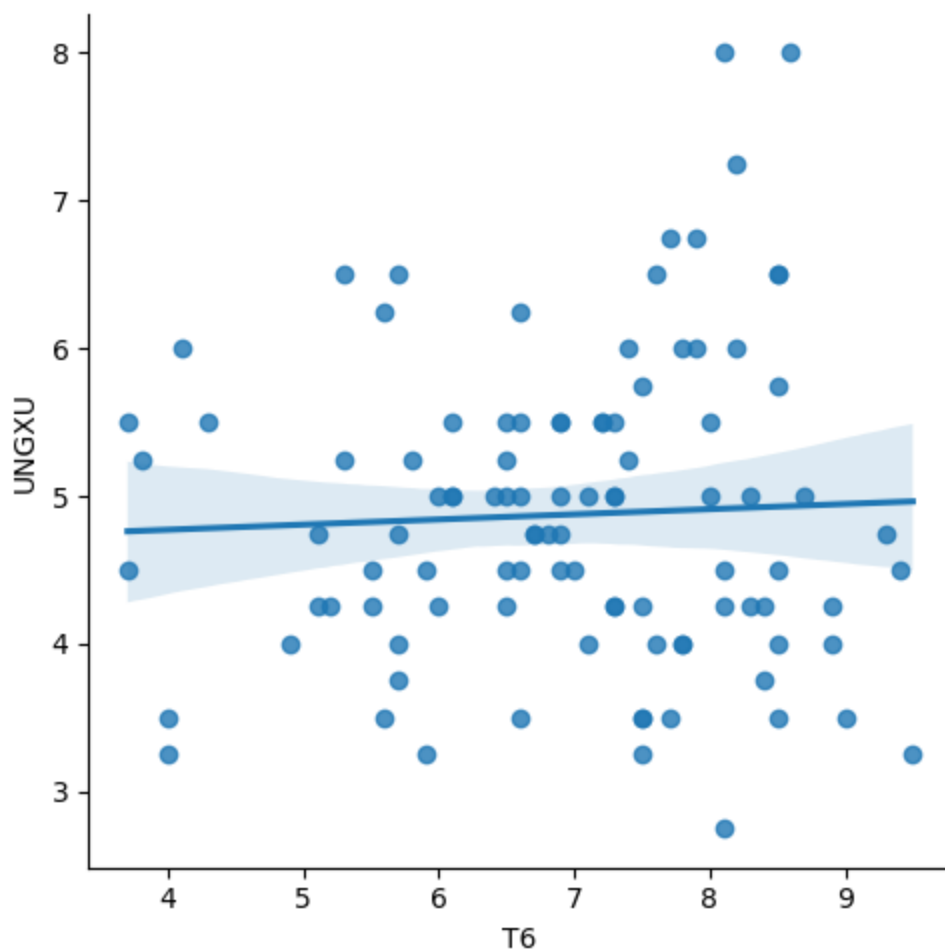
e:\LAP TRÌNH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight  
self.\_figure.tight\_layout(\*args, \*\*kwargs)



In [262]: *# Sinh viên tự khám phá độ tương quan giữa biến T6 và UNGXU*

```
df[['T6','UNGXU']].corr()
sns.lmplot(data=df, x='T6',y='UNGXU', fit_reg=True)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight  
self.\_figure.tight\_layout(\*args, \*\*kwargs)



In [263]: `df[['T6','UNGXU']].corr()`

Out[263]:

	T6	UNGXU
T6	1.000000	0.045638
UNGXU	0.045638	1.000000

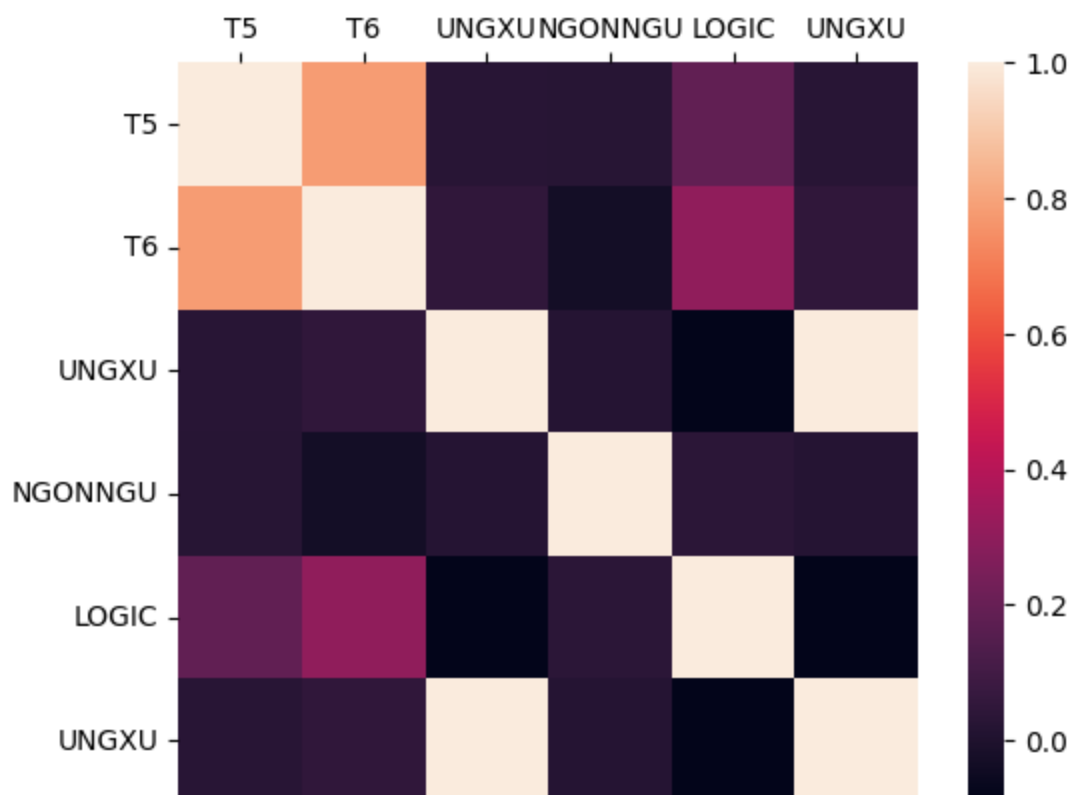


In [264]: `df[['T6','UNGXU','NGONNGU','LOGIC','UNGXU']].corr()`

Out[264]:

	T6	UNGXU	NGONNGU	LOGIC	UNGXU
T6	1.000000	0.045638	-0.033360	0.301490	0.045638
UNGXU	0.045638	1.000000	0.018731	-0.091523	1.000000
NGONNGU	-0.033360	0.018731	1.000000	0.035154	0.018731
LOGIC	0.301490	-0.091523	0.035154	1.000000	-0.091523
UNGXU	0.045638	1.000000	0.018731	-0.091523	1.000000

In [265]: `sns.heatmap(df[['T5','T6','UNGXU','NGONNGU','LOGIC','UNGXU']].corr(),vmax=1.0,square=True).xaxis.t`



In [266]: *# Biểu đồ tổng hợp khám phá tổng hợp nhiều biến định lượng*  
 sns.pairplot(df[['T5','T6','NGONNGU','LOGIC','UNGXU']],  
 diag\_kind='kde',kind='reg')  
 plt.show()

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):  
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):  
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):  
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):  
 e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead  
 if pd.api.types.is\_categorical\_dtype(vector):

```
In [267]: # Trực quan tương quan tuyến tính theo nhóm (định tính) giữa 2 biến định lượng
sns.lmplot(data=df,x='T5',y='T6',hue='GT',fit_reg=True)
plt.show()
```

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\\_oldcore.py:1498: FutureWarning: is\_categorical\_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

if pd.api.types.is\_categorical\_dtype(vector):

e:\LAP TRINH PYTHON\LT PTDL 1\venv\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight

self.\_figure.tight\_layout(\*args, \*\*kwargs)

