Python for Data Analysis

Drug Consumption

Table of content

Missing Values

We observe that there are no missings values in the DataFrame.

We won’t need do delete any rows.

```
print(df.isna().sum())
```

```
Age          0
Gender       0
Education    0
Country      0
```

...

```
Ketamine     0
Legalh       0
LSD          0
Meth         0
Mushrooms    0
Nicotine     0
Semer        0
VSA          0
dtype: int64
```

Database features

Personal Information

- Age range
- Gender
- Level of education
- Country
- Ethnicity

Personality Scores

- N score
- E score
- O score
- A score
- C score
- BIS-11
- ImpSS

Consumption Frequency

- Alcohol
- Amphetamines
- Amyl nitrite
- Benzodiazepine
- Cannabis
- Chocolate
- Cocaïne
- Caffeine
- Crack
- Ecstasy
- Heroin
- Ketamine
- Legal highs
- LSD
- Methadone
- Mushrooms
- Nicotine
- volatile substance
- Semeron (fictionnal)

Presentation of the database

- Created by E. Fehrman, V. Egan and E. Mirkes [1]

- Record of 1885 individuals regarding their profile and drugs consumption

[1] Fehrman,Elaine, Egan,Vincent, and Mirkes,Evgeny. (2016). Drug consumption (quantified). UCI Machine Learning Repository. https://doi.org/10.24432/C5TC7S.

Data Pre Processing – Overview

Goal of this task : Prepare the dataframe for the followings parts :

- Data visualization
- Modeling

A **specific** data pre processing is required for each part.

Missing Values

We observe that there are no missings values in the DataFrame.

Thus we won't need do delete any rows.

```
print(df.isna().sum())
```

```
Age          0
Gender       0
Education    0
Country      0
```

...

```
Ketamine     0
Legalh       0
LSD          0
Meth         0
Mushrooms    0
Nicotine     0
Semer        0
VSA          0
dtype: int64
```

A quick look at the DataFrame...

	Age	Gender	Education	Country	Ethnicity	Nscore	Escore	Oscore	Ascore
ID									
1	0.49788	0.48246	-0.05921	0.96082	0.12600	0.31287	-0.57545	-0.58331	-0.91699
2	-0.07854	-0.48246	1.98437	0.96082	-0.31685	-0.67825	1.93886	1.43533	0.76096
3	0.49788	-0.48246	-0.05921	0.96082	-0.31685	-0.46725	0.80523	-0.84732	-1.62090
4	-0.95197	0.48246	1.16365	0.96082	-0.31685	-0.14882	-0.80615	-0.01928	0.59042
5	0.49788	0.48246	1.98437	0.96082	-0.31685	0.73545	-1.63340	-0.45174	-0.30172

- At this point, the values are not understable. We'll replace them for the visualization's purpose by their corresponding readable values given in the study

Example : Replacement of the Age column

ID	
1	0.49788
2	-0.07854
3	0.49788
4	-0.95197
5	0.49788

```python
# Replacement of the Age column's values
age_dict = {
    -0.95197: '18-24',
    -0.07854: '25-34',
    0.49788: '35-44',
    1.09449: '45-54',
    1.82213: '55-64',
    2.59171: '65+'
}
df_v.replace({"Age": age_dict},inplace=True)
```

ID	
1	35-44
2	25-34
3	35-44
4	18-24
5	35-44

We repeated this processes for every column of the DataFrame to get understandable values

DataFrame ready for visualization

ID	Age	Gender	Education	Country	Ethnicity	Nscore	Escore	Oscore	Ascore
1	35-44	Female	Professional certificate/ diploma	UK	Mixed-White/Asian	39.0	36.0	42.0	37.0
2	25-34	Male	Doctorate degree	UK	White	29.0	52.0	55.0	48.0
3	35-44	Male	Professional certificate/ diploma	UK	White	31.0	45.0	40.0	32.0
4	18-24	Female	Masters degree	UK	White	34.0	34.0	46.0	47.0
5	35-44	Female	Doctorate degree	UK	White	43.0	28.0	43.0	41.0

Data Pre Processing – Modeling

In this section, we prepare the DataFrame for the Modeling part.

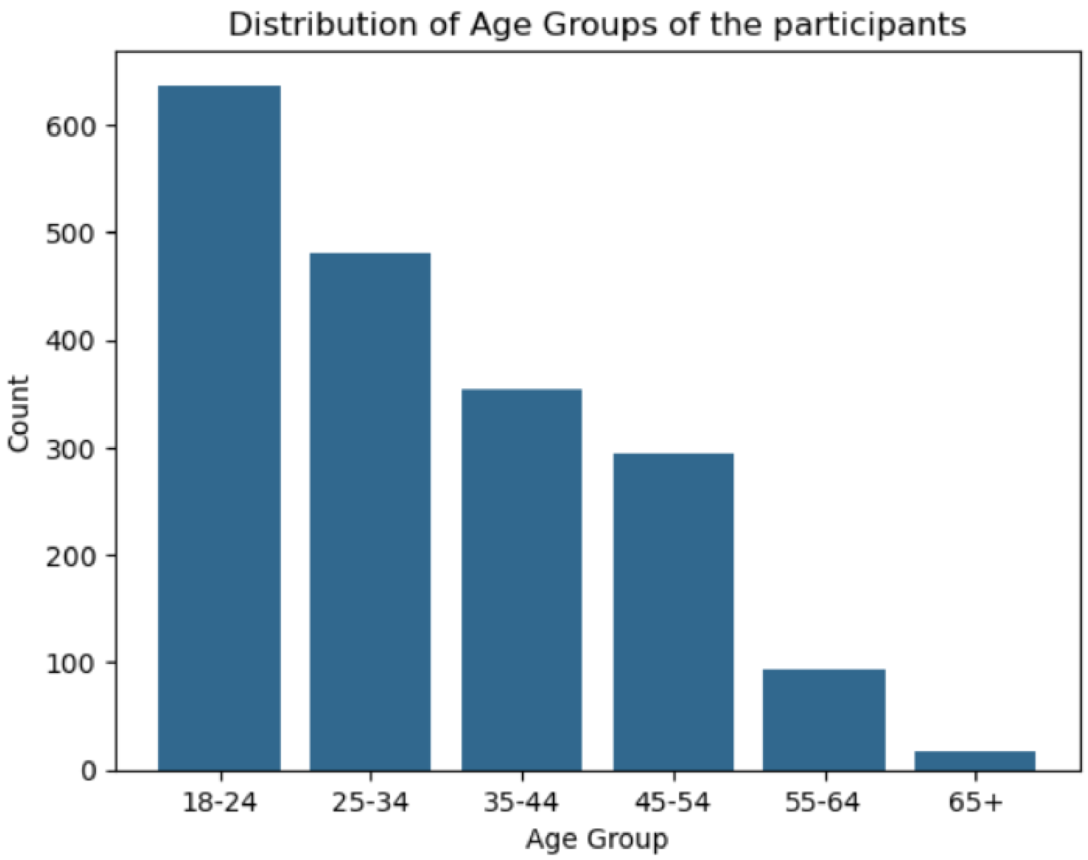
However, some columns contain some **nominal values,**

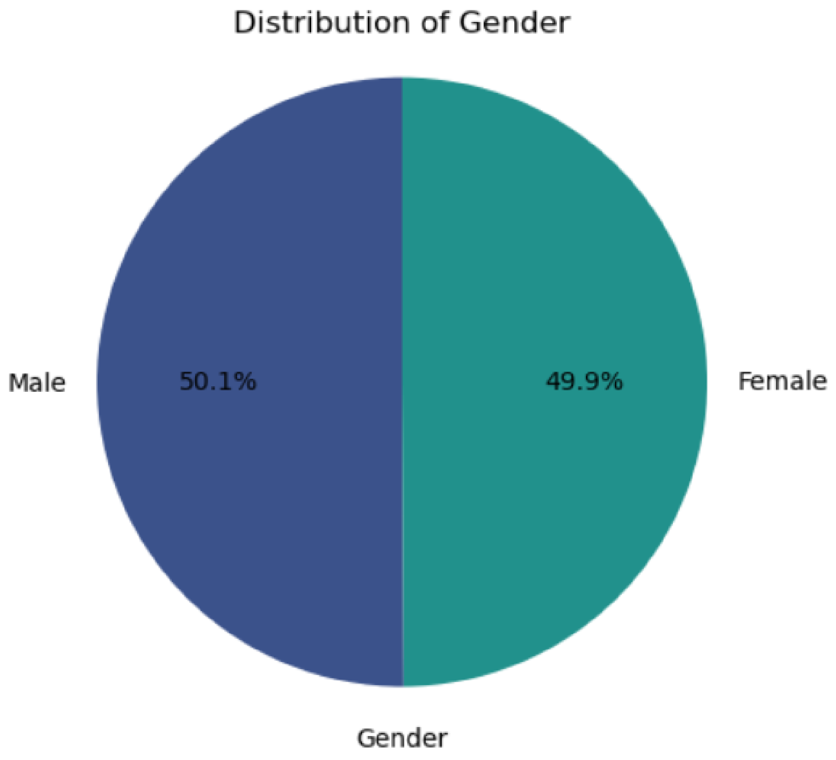
But the models can only allow numerical values in input.

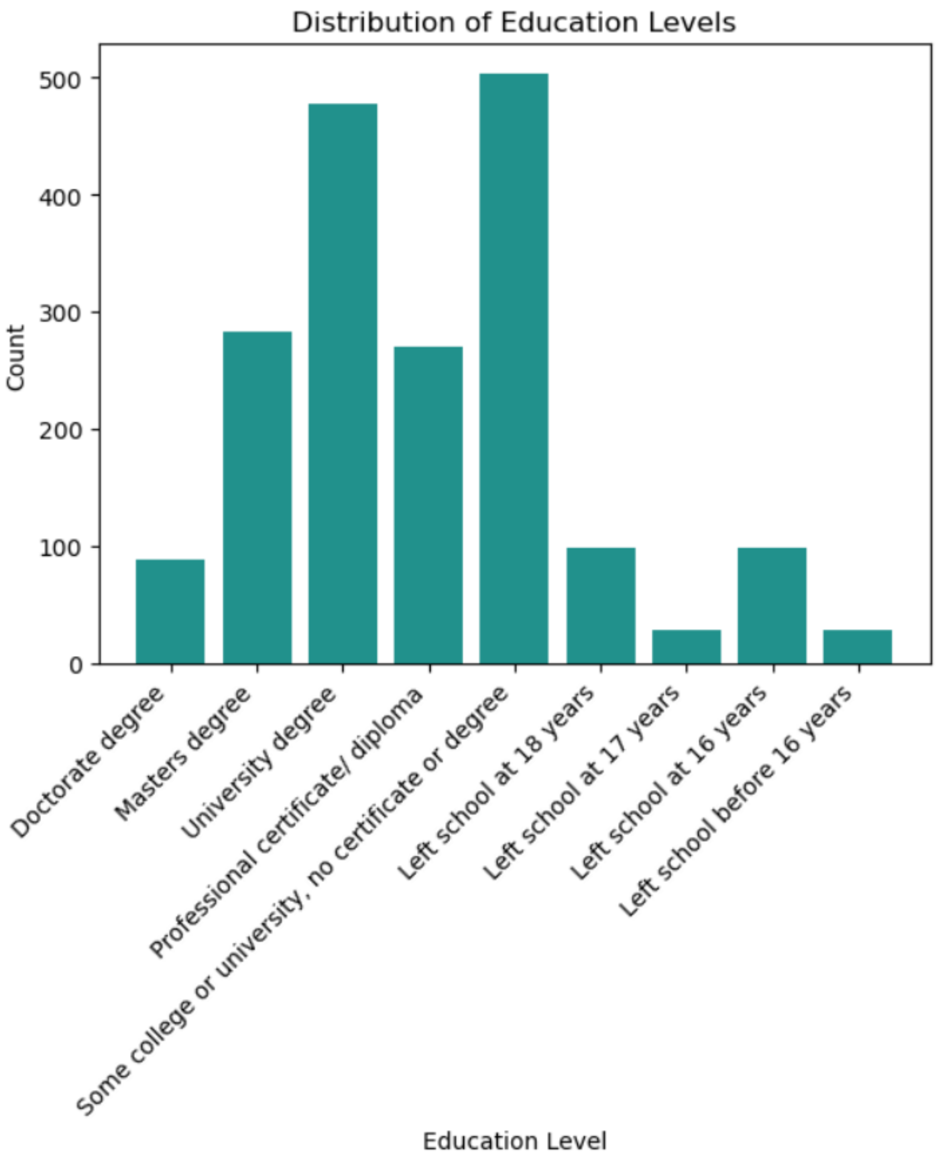
Gender
Female
Male
Male

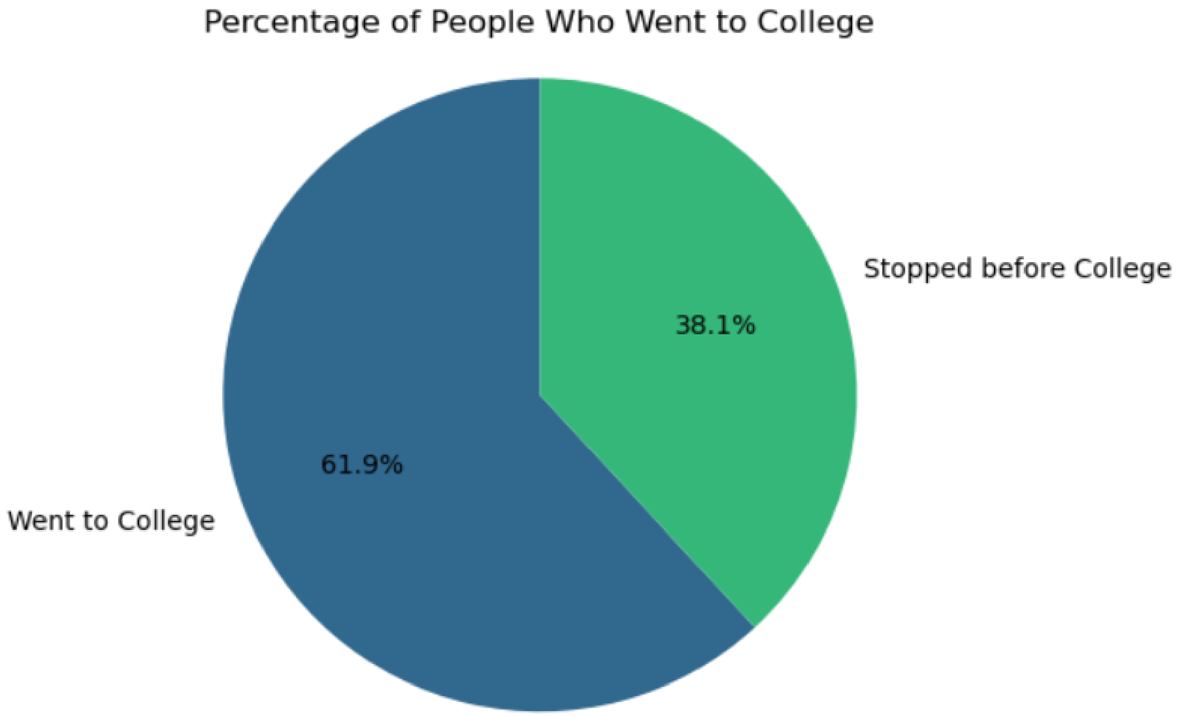
Example of nominal values in the DataFrame

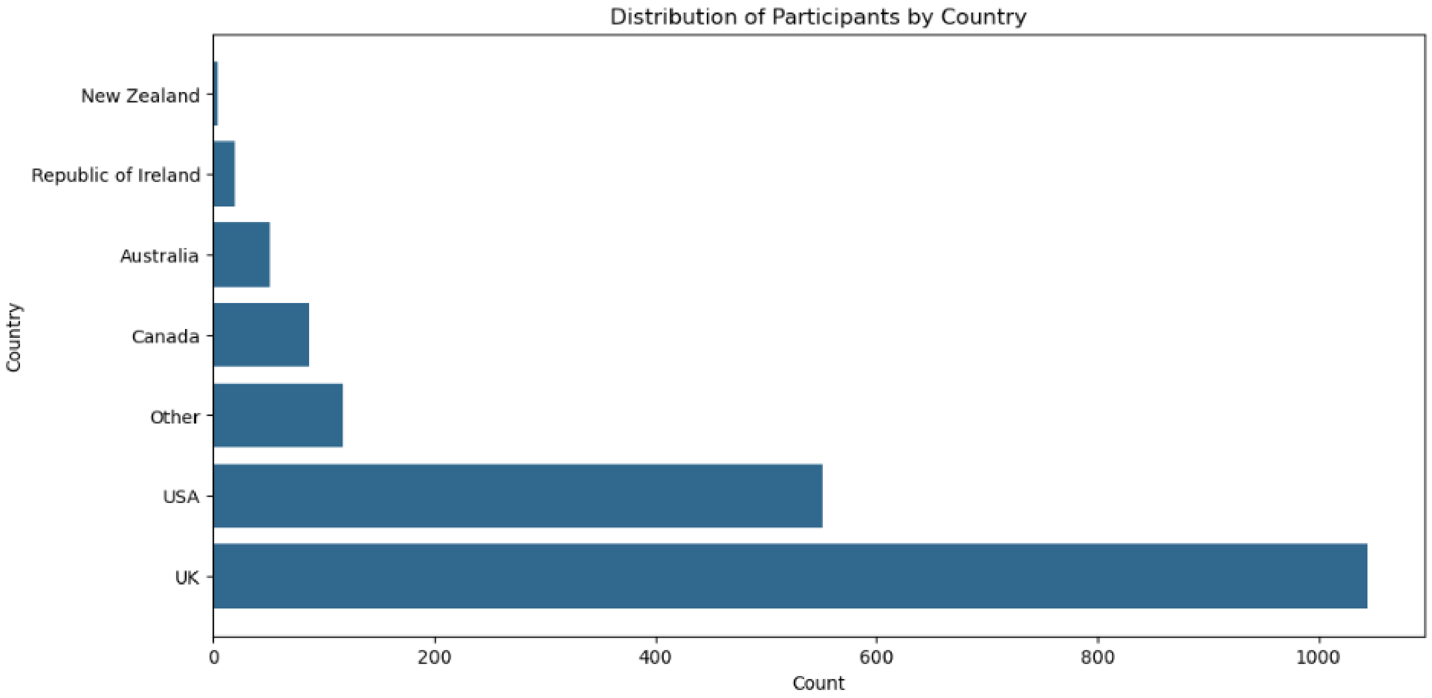
Data Vizualisation : Presentation of the data distribution:



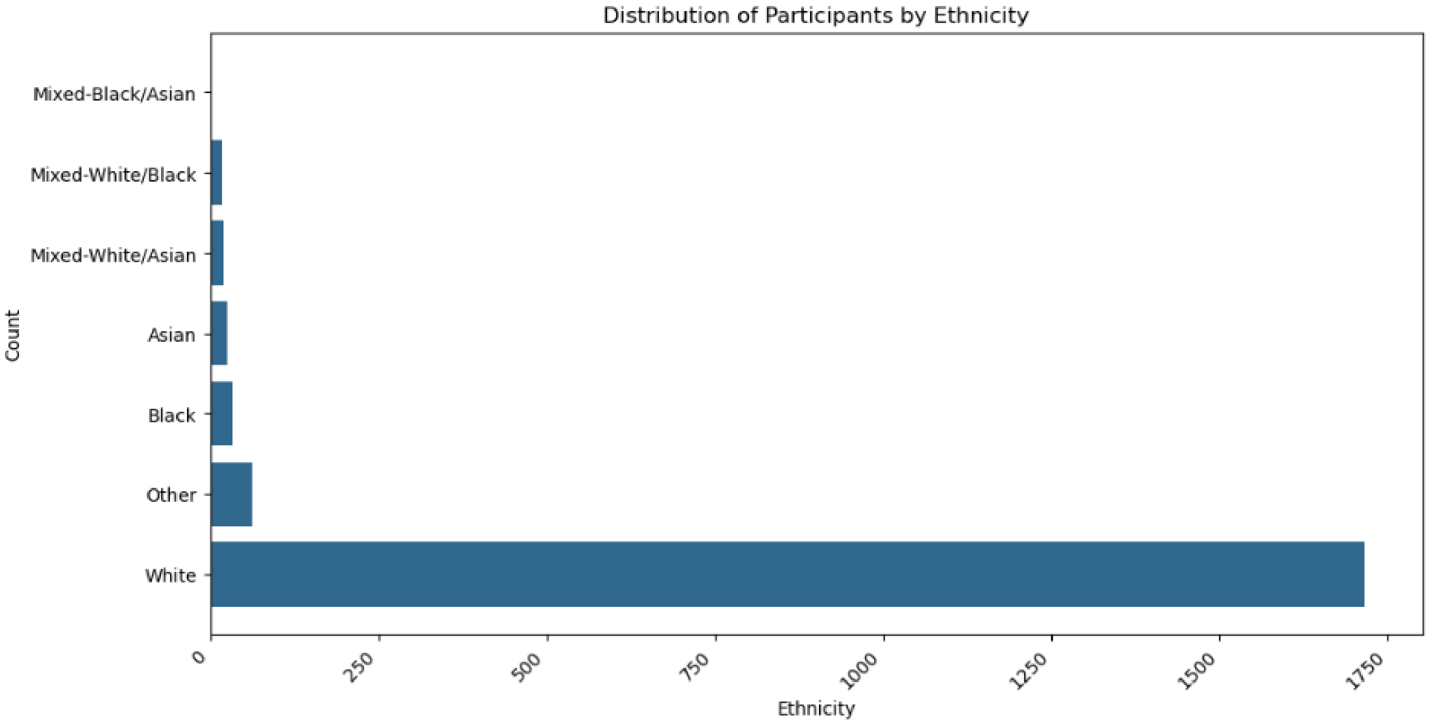



Distribution of Education Levels
500
400
300
200
100
0
Count
Doctorate degree
Masters degree
University degree
Professional certificate/ diploma
Some college or university, no certificate or degree
Left school at 18 years
Left school at 17 years
Left school at 16 years
Left school before 16 years
Education Level


Percentage of People Who Went to College
Stopped before College
38.1%
61.9%
Went to College



Distribution of Participants by Country
New Zealand
Republic of Ireland
Australia
Canada
Other
USA
UK
Country
0
200
400
600
800
1000
Count



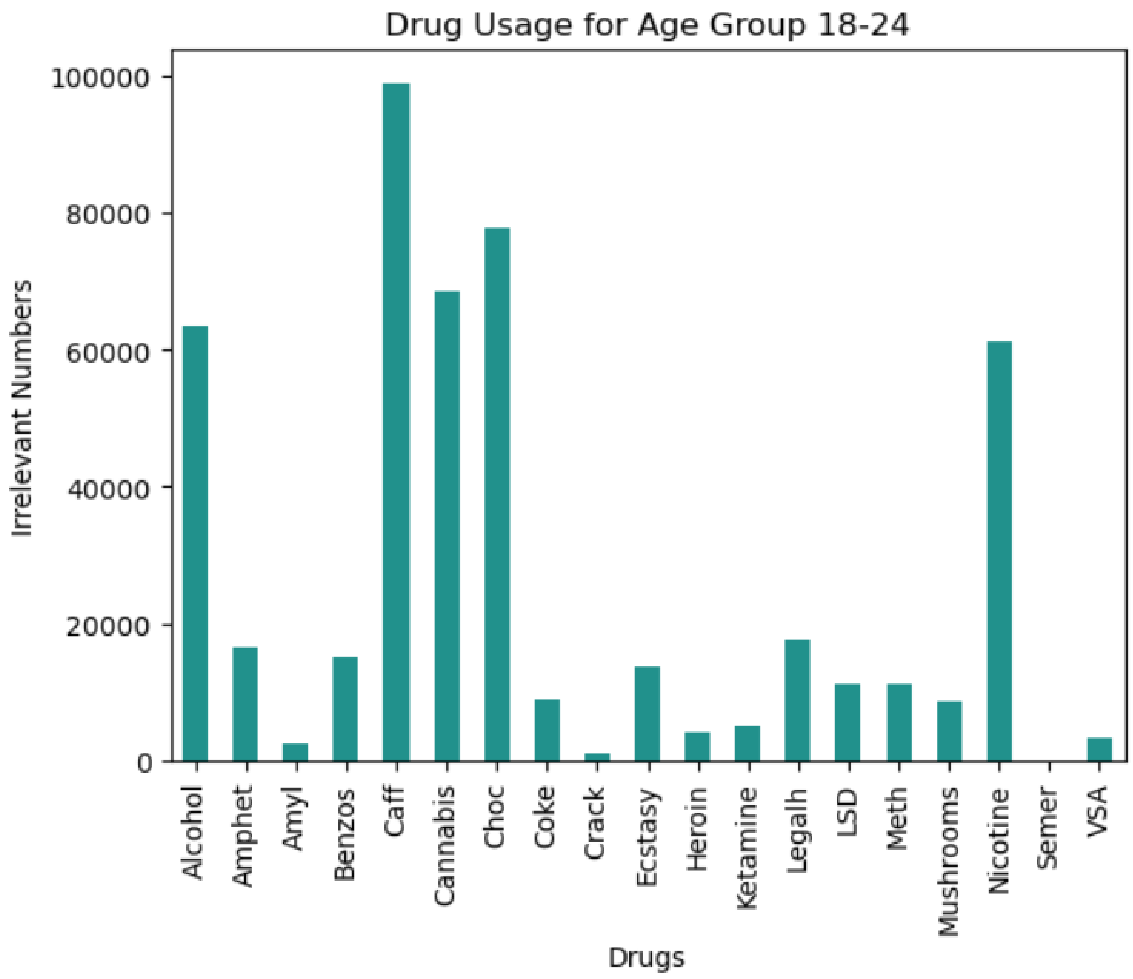
Distribution of Participants by Ethnicity
Mixed-Black/Asian
Mixed-White/Black
Mixed-White/Asian
Asian
Black
Other
White
Count
0
250
500
750
1000
1250
1500
1750
Ethnicity

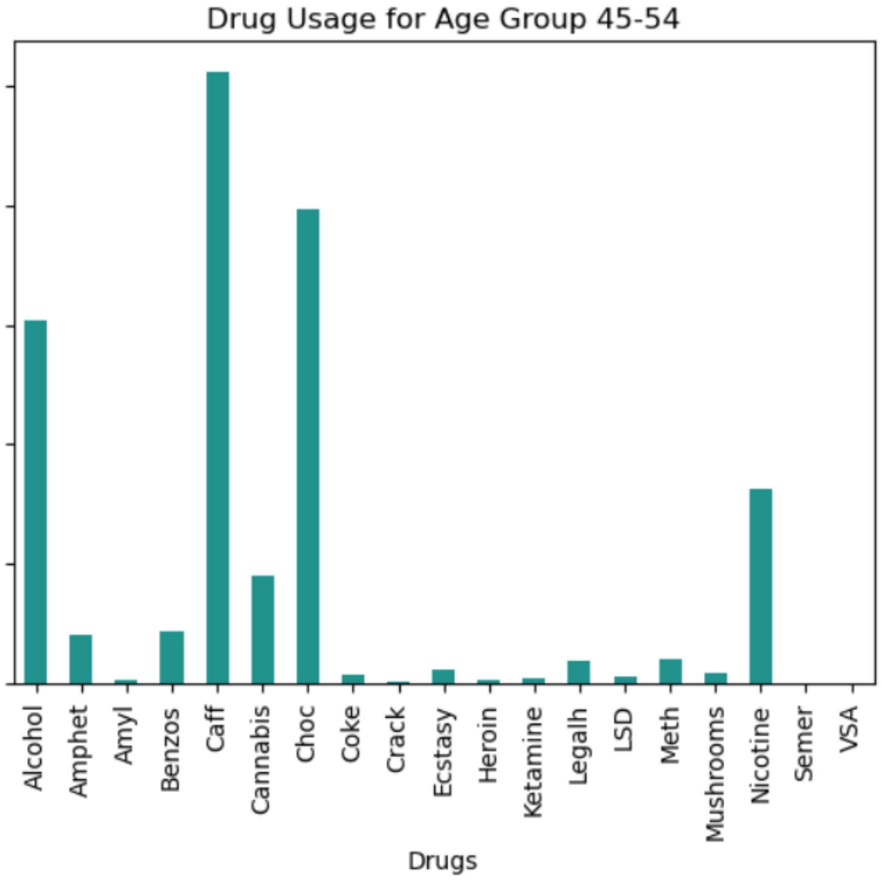
Data Vizualisation : Analysis steps

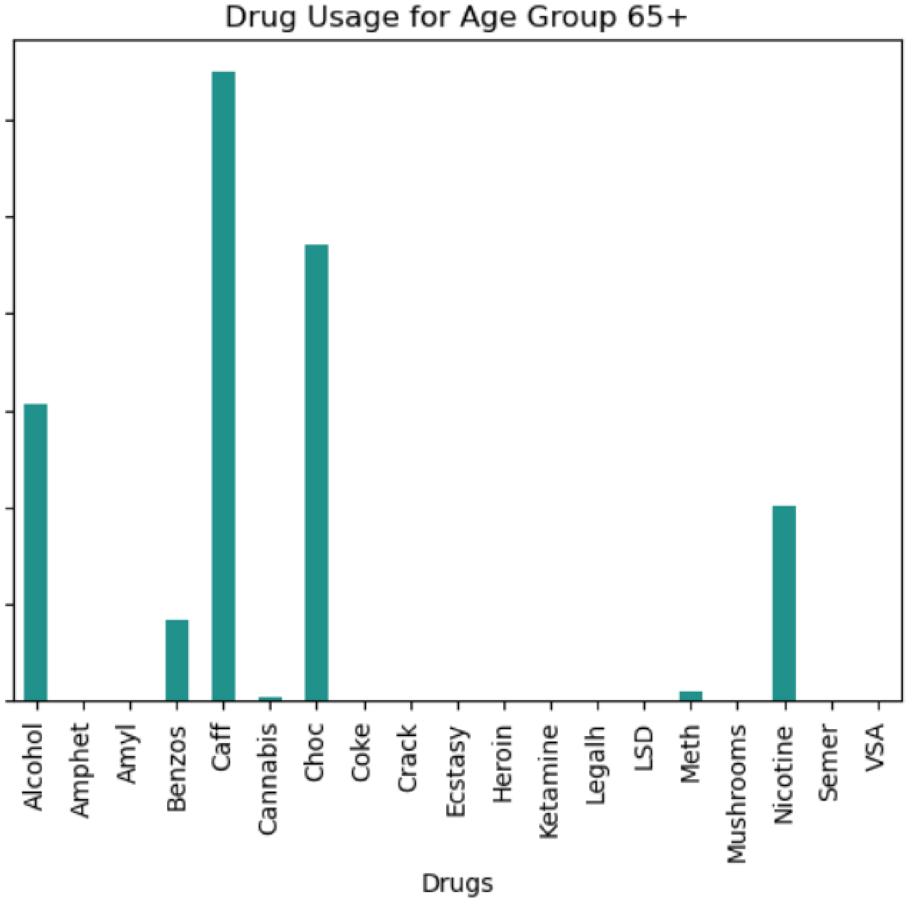
- Analyze how age correlates with the use of different drugs.
- Search for differences in drug consumption based on the gender.
- Analyze drug use patterns based on the country of residence.
- Explore correlations between the use of different drugs. For example, do individuals who use one type of drug tend to use another?
- Explore how different personality traits (NEO-FFI- R) correlate with drug use.

Age and Drug Use

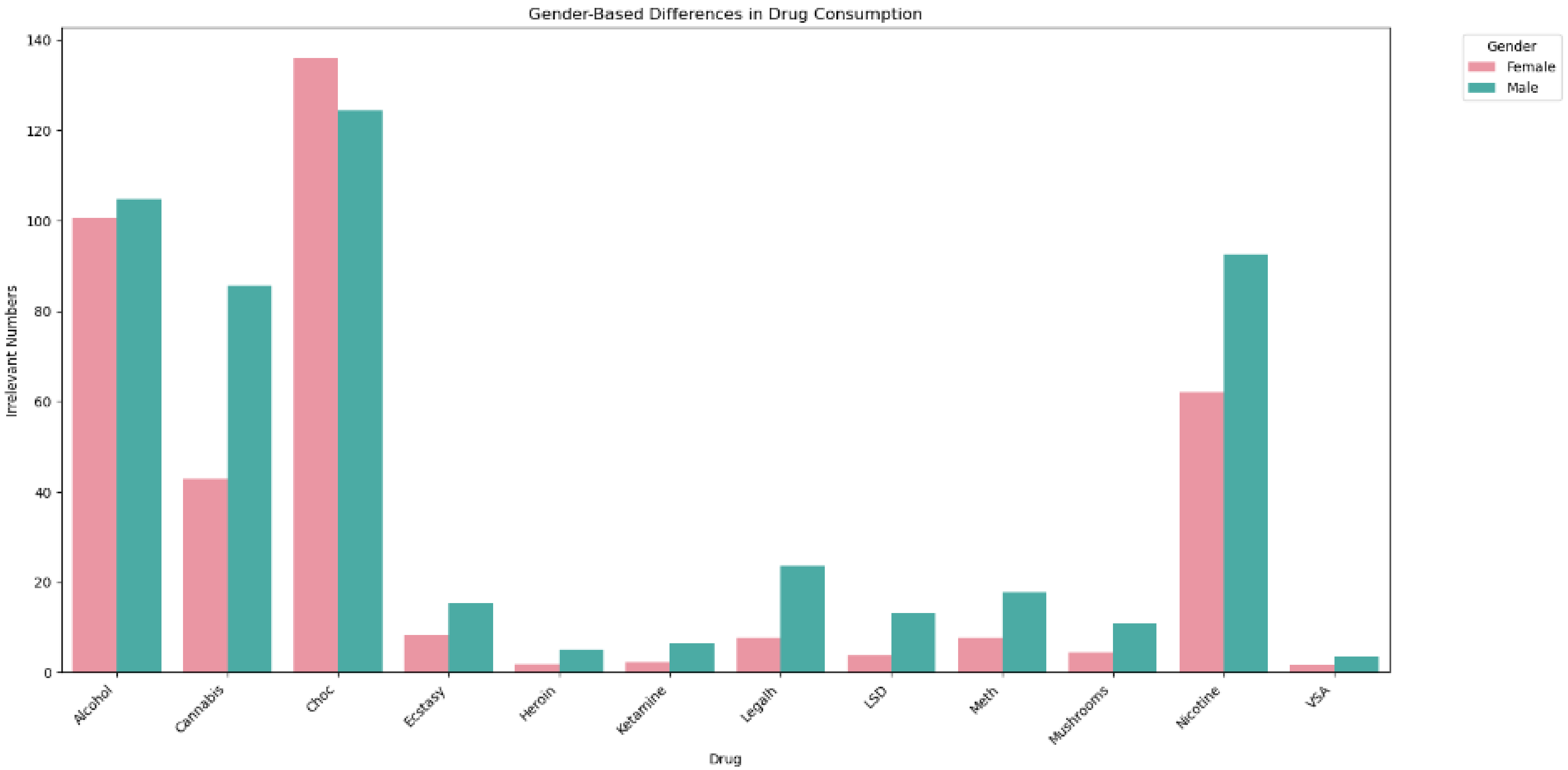
```python
usage_mapping = {
    'Never Used': 0,
    'Used over a Decade Ago': 1,
    'Used in Last Decade': 5,
    'Used in Last Year': 10,
    'Used in Last Month': 50,
    'Used in Last Week': 100,
    'Used in Last Day': 200
}
```

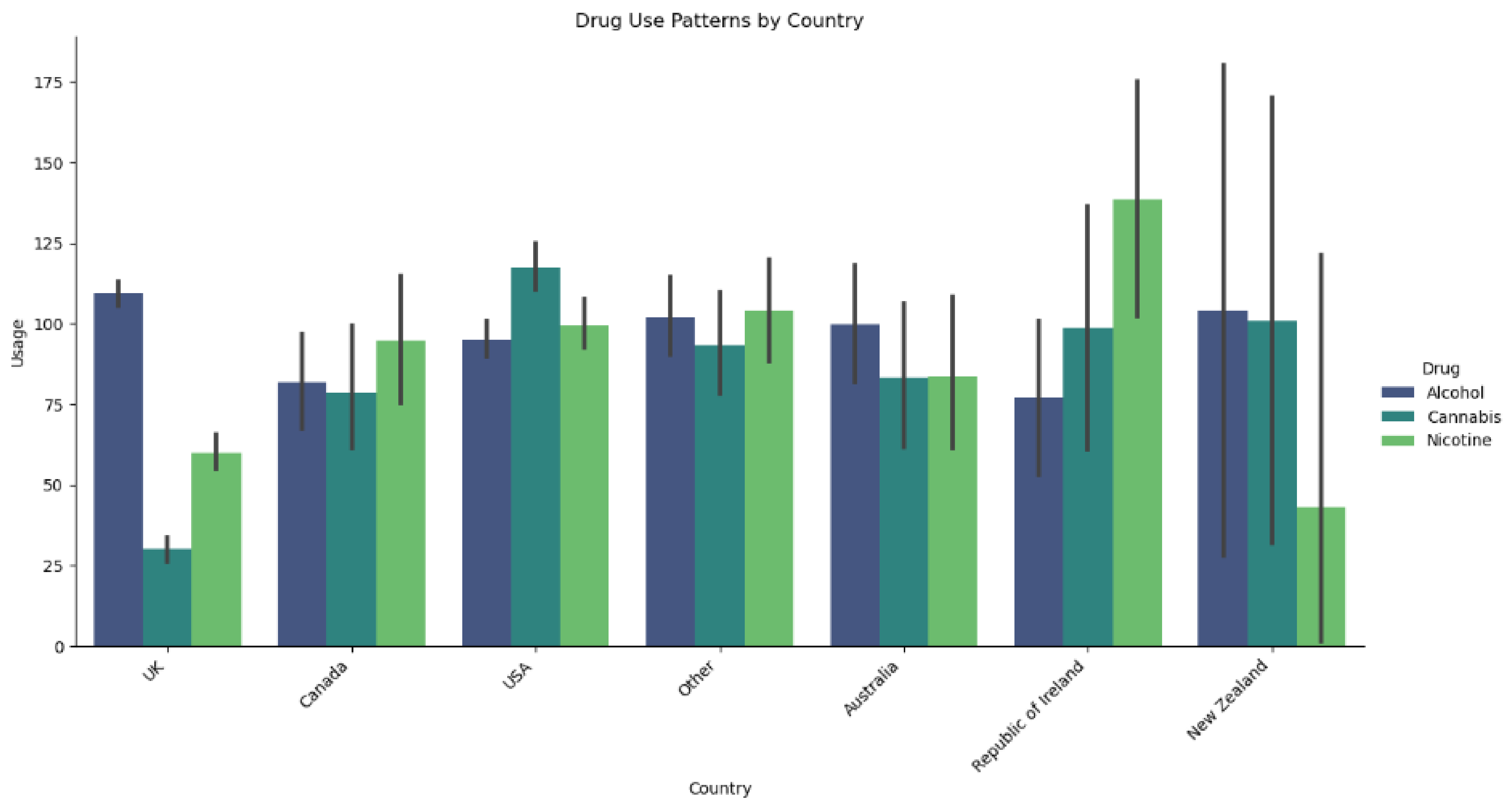


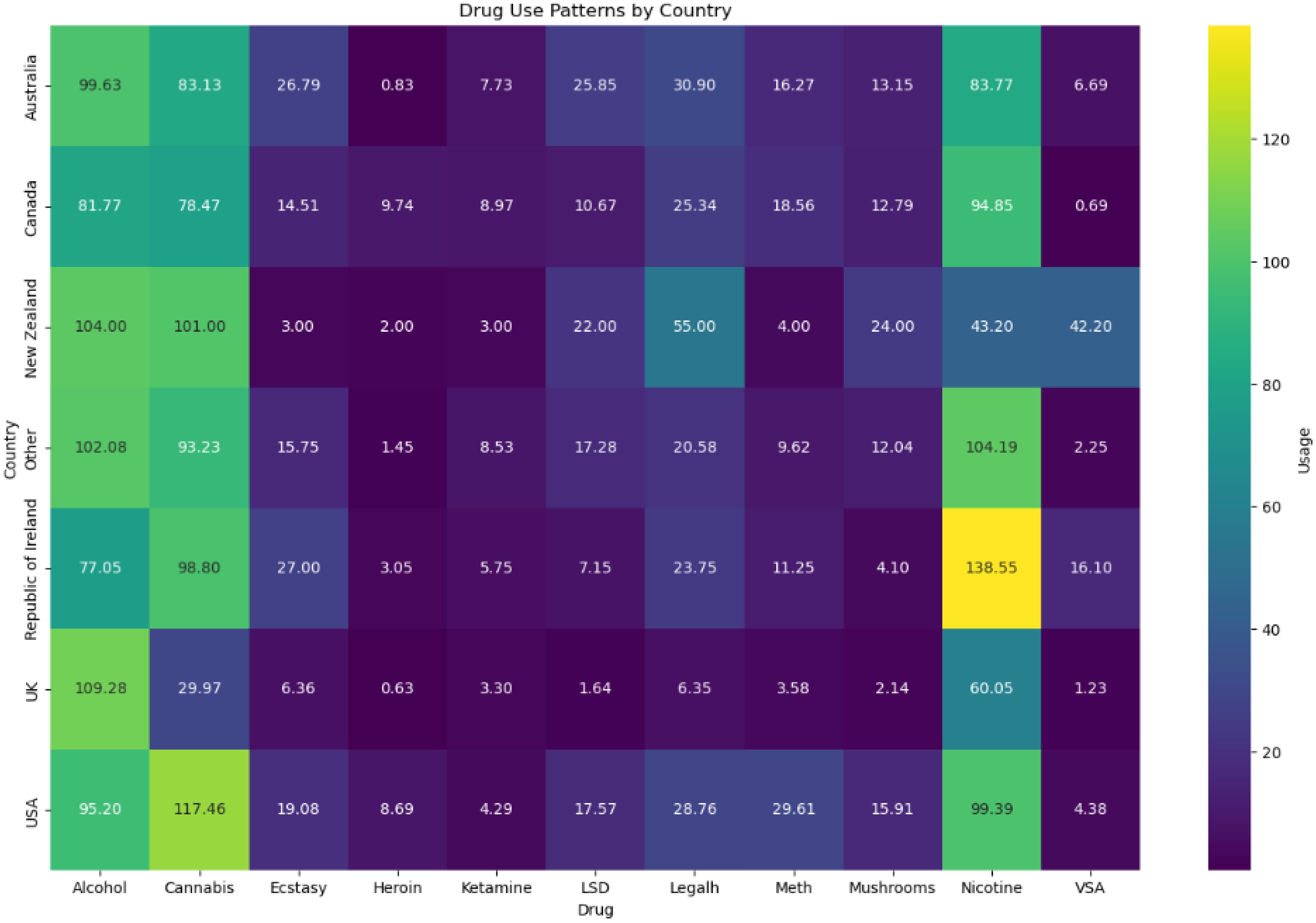

Drug Usage for Age Group 45-54
Alcohol
Amphet
Amyl
Benzos
Caff
Cannabis
Choc
Coke
Crack
Ecstasy
Heroin
Ketamine
Legalh
LSD
Meth
Mushrooms
Nicotine
Semer
VSA
Drugs

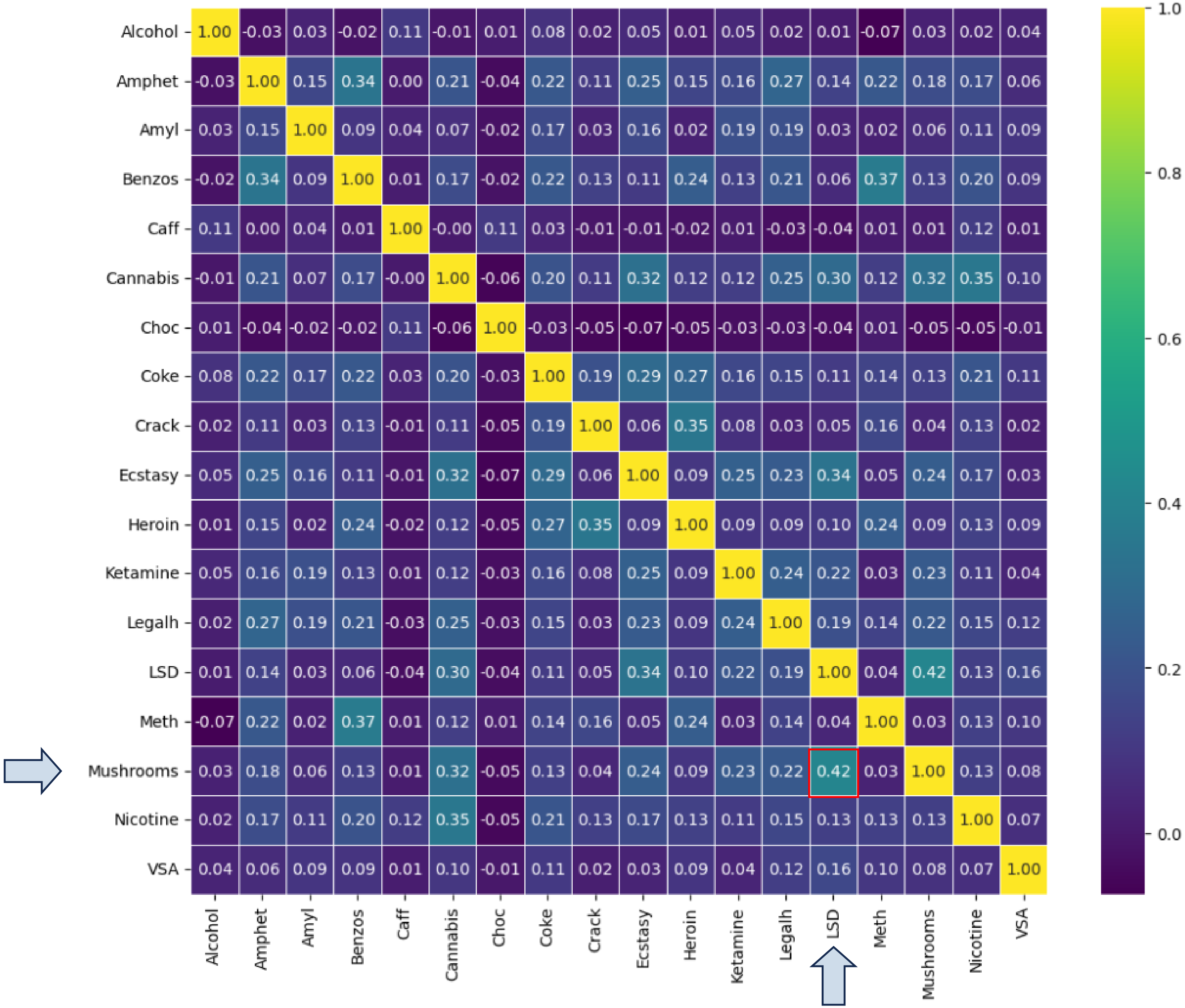

Drug Usage for Age Group 65+
Alcohol
Amphet
Amyl
Benzos
Caff
Cannabis
Choc
Coke
Crack
Ecstasy
Heroin
Ketamine
Legalh
LSD
Meth
Mushrooms
Nicotine
Semer
VSA
Drugs

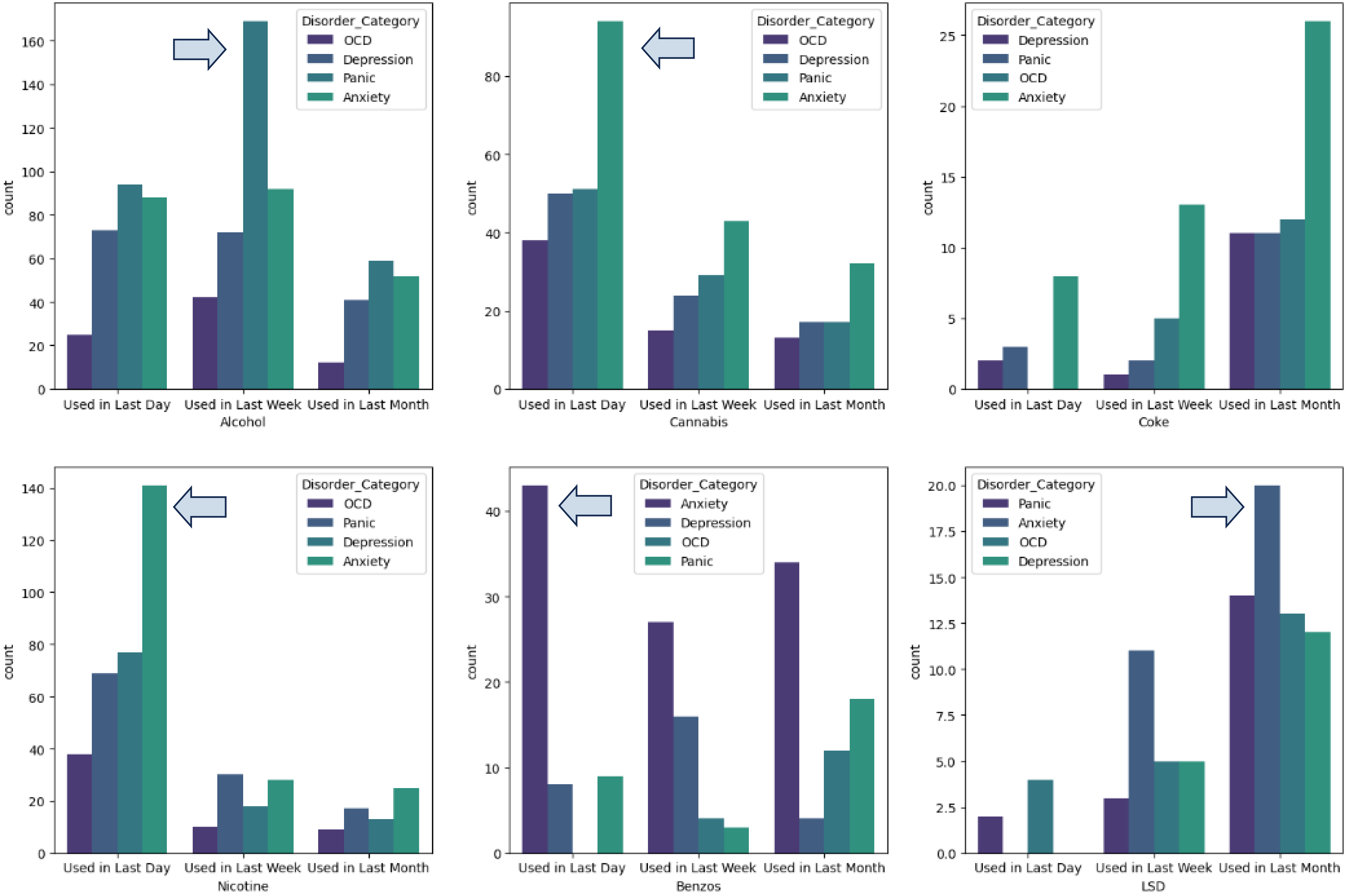
Gender and Drug Use




Drug Use Patterns by Country
Usage
0
25
50
75
100
125
150
175
UK
Canada
USA
Other
Australia
Republic of Ireland
New Zealand
Country
Drug
Alcohol
Cannabis
Nicotine


Drug Use Patterns by Country
Country
Australia
Canada
New Zealand
Other
Republic of Ireland
UK
USA
Drug
Alcohol
Cannabis
Ecstasy
Heroin
Ketamine
LSD
Legalh
Meth
Mushrooms
Nicotine
VSA
Usage
20
40
60
80
100
120
99.63 83.13 26.79 0.83 7.73 25.85 30.90 16.27 13.15 83.77 6.69
81.77 78.47 14.51 9.74 8.97 10.67 25.34 18.56 12.79 94.85 0.69
104.00 101.00 3.00 2.00 3.00 22.00 55.00 4.00 24.00 43.20 42.20
102.08 93.23 15.75 1.45 8.53 17.28 20.58 9.62 12.04 104.19 2.25
77.05 98.80 27.00 3.05 5.75 7.15 23.75 11.25 4.10 138.55 16.10
109.28 29.97 6.36 0.63 3.30 1.64 6.35 3.58 2.14 60.05 1.23
95.20 117.46 19.08 8.69 4.29 17.57 28.76 29.61 15.91 99.39 4.38


Alcohol 1.00 -0.03 0.03 -0.02 0.11 -0.01 0.01 0.08 0.02 0.05 0.01 0.05 0.02 0.01 -0.07 0.03 0.02 0.04
Amphet -0.03 1.00 0.15 0.34 0.00 0.21 -0.04 0.22 0.11 0.25 0.15 0.16 0.27 0.14 0.22 0.18 0.17 0.06
Amyl 0.03 0.15 1.00 0.09 0.04 0.07 -0.02 0.17 0.03 0.16 0.02 0.19 0.19 0.03 0.02 0.06 0.11 0.09
Benzos -0.02 0.34 0.09 1.00 0.01 0.17 -0.02 0.22 0.13 0.11 0.24 0.13 0.21 0.06 0.37 0.13 0.20 0.09
Caff 0.11 0.00 0.04 0.01 1.00 -0.00 0.11 0.03 -0.01 -0.01 -0.02 0.01 -0.03 -0.04 0.01 0.01 0.12 0.01
Cannabis -0.01 0.21 0.07 0.17 -0.00 1.00 -0.06 0.20 0.11 0.32 0.12 0.12 0.25 0.30 0.12 0.32 0.35 0.10
Choc 0.01 -0.04 -0.02 -0.02 0.11 -0.06 1.00 -0.03 -0.05 -0.07 -0.05 -0.03 -0.03 -0.04 0.01 -0.05 -0.05 -0.01
Coke 0.08 0.22 0.17 0.22 0.03 0.20 -0.03 1.00 0.19 0.29 0.27 0.16 0.15 0.11 0.14 0.13 0.21 0.11
Crack 0.02 0.11 0.03 0.13 -0.01 0.11 -0.05 0.19 1.00 0.06 0.35 0.08 0.03 0.05 0.16 0.04 0.13 0.02
Ecstasy 0.05 0.25 0.16 0.11 -0.01 0.32 -0.07 0.29 0.06 1.00 0.09 0.25 0.23 0.34 0.05 0.24 0.17 0.03
Heroin 0.01 0.15 0.02 0.24 -0.02 0.12 -0.05 0.27 0.35 0.09 1.00 0.09 0.09 0.10 0.24 0.09 0.13 0.09
Ketamine 0.05 0.16 0.19 0.13 0.01 0.12 -0.03 0.16 0.08 0.25 0.09 1.00 0.24 0.22 0.03 0.23 0.11 0.04
Legalh 0.02 0.27 0.19 0.21 -0.03 0.25 -0.03 0.15 0.03 0.23 0.09 0.24 1.00 0.19 0.14 0.22 0.15 0.12
LSD 0.01 0.14 0.03 0.06 -0.04 0.30 -0.04 0.11 0.05 0.34 0.10 0.22 0.19 1.00 0.04 0.42 0.13 0.16
Meth -0.07 0.22 0.02 0.37 0.01 0.12 0.01 0.14 0.16 0.05 0.24 0.03 0.14 0.04 1.00 0.03 0.13 0.10
Mushrooms 0.03 0.18 0.06 0.13 0.01 0.32 -0.05 0.13 0.04 0.24 0.09 0.23 0.22 0.42 0.03 1.00 0.13 0.08
Nicotine 0.02 0.17 0.11 0.20 0.12 0.35 -0.05 0.21 0.13 0.17 0.13 0.11 0.15 0.13 0.13 0.13 1.00 0.07
VSA 0.04 0.06 0.09 0.09 0.01 0.10 -0.01 0.11 0.02 0.03 0.09 0.04 0.12 0.16 0.10 0.08 0.07 1.00
Alcohol Amphet Amyl Benzos Caff Cannabis Choc Coke Crack Ecstasy Heroin Ketamine Legalh LSD Meth Mushrooms Nicotine VSA
1.0
0.8
0.6
0.4
0.2
0.0


Disorder_Category
OCD
Depression
Panic
Anxiety
count
Used in Last Day
Used in Last Week
Used in Last Month
Alcohol
Disorder_Category
OCD
Depression
Panic
Anxiety
count
Used in Last Day
Used in Last Week
Used in Last Month
Cannabis
Disorder_Category
Depression
Panic
OCD
Anxiety
count
Used in Last Day
Used in Last Week
Used in Last Month
Coke
Disorder_Category
OCD
Panic
Depression
Anxiety
count
Used in Last Day
Used in Last Week
Used in Last Month
Nicotine
Disorder_Category
Anxiety
Depression
OCD
Panic
count
Used in Last Day
Used in Last Week
Used in Last Month
Benzos
Disorder_Category
Panic
Anxiety
OCD
Depression
count
Used in Last Day
Used in Last Week
Used in Last Month
LSD

Strategies proposed

We first wanted to arbitrarily attribue numbers to the categories

(example for Age : 18-24 = 0, 25-34 = 1…)

But this would create a bias in the model.

Hence we proposed to represent the categories as the following :

Individual	18-24	25-34	35-44	45-54	55-64
Indiv 1	1	0	0	0	0
Indiv 2	0	0	0	1	0

In this case, an individual with zeros in all categories would belong to the 65+ category.

Application on the DataFrame

```
age_d = pd.get_dummies(df_v.Age)
age_d.drop([age_d.columns[len(age_d.columns)-1]], axis=1, inplace=True)

df_m = pd.concat([df_m, age_d], axis=1, join='inner')
```

	Age
ID	
1	35-44
2	25-34
3	35-44

	18-24	25-34	35-44	45-54	55-64
ID					
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	0	0
4	1	0	0	0	0
5	0	0	1	0	0

DataFrame ready for Modeling

ID	18-24	25-34	35-44	45-54	55-64	Female	Doctorate degree	Left school at 16 years	Left school at 17 years	Left school at 18 years	...	Semer Never Used	Semer Used in Last Decade	Semer Used in Last Month	Semer Used in Last Year	VSA Never Used	VSA Used in Last Day	VSA Used in Last Decade	VSA Used in Last Month	VSA Used in Last Week	VSA Used in Last Year
1	0	0	1	0	0	1	0	0	0	0	...	1	0	0	0	1	0	0	0	0	0
2	0	1	0	0	0	0	1	0	0	0	...	1	0	0	0	1	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0	0	...	1	0	0	0	1	0	0	0	0	0
4	1	0	0	0	0	1	0	0	0	0	...	1	0	0	0	1	0	0	0	0	0
5	0	0	1	0	0	1	1	0	0	0	...	1	0	0	0	1	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1884	1	0	0	0	0	1	0	0	0	0	...	1	0	0	0	0	0	0	0	1	0
1885	1	0	0	0	0	0	0	0	0	0	...	1	0	0	0	1	0	0	0	0	0
1886	0	1	0	0	0	1	0	0	0	0	...	1	0	0	0	1	0	0	0	0	0
1887	1	0	0	0	0	1	0	0	0	0	...	1	0	0	0	1	0	0	0	0	0
1888	1	0	0	0	0	0	0	0	0	0	...	1	0	0	0	0	0	1	0	0	0

Modeling

Our entry Values :

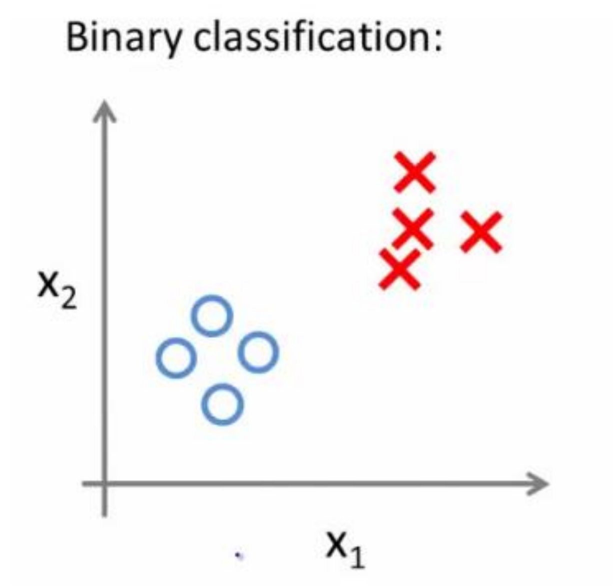
- Age
- Gender
- Education level
- Country of residence
- Ethnicity
- Personality Test Scores NEO-FFI-R et BIS-11

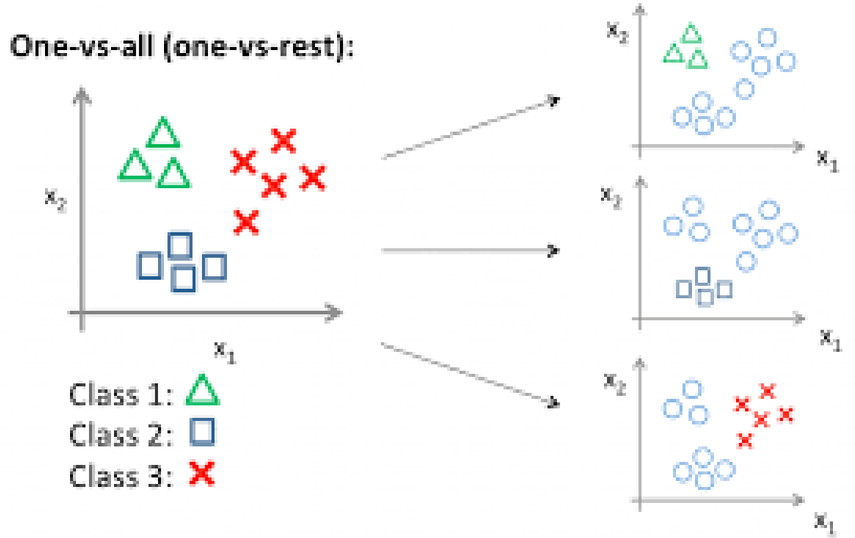
Values to predict :

The frequency of consuption of each drug for each individual

Classification multi-label

- Classification models -> binary classification
- One vs All -> multi-class classification
- MultiLabelBinarizer -> multi-label classification





Models and metrics used

Models:

- Linear SVC
- KNN
- Naive Bayes
- SVC
- SGD

Metrics:

- Precision
- Rappel
- F1
- Hamming loss

Performance des modèles sur dataset initial

```
Precision_ LinearSVC() :  0.4002654080681768
Rappel_ LinearSVC() :  0.4618794326241135
f1_ LinearSVC() :  0.42837663992171576
Hamming Loss_ LinearSVC() :  0.1743498817966903


Precision_ KNeighborsClassifier() :  0.6525115338006452
Rappel_ KNeighborsClassifier() :  0.4664598108747045
f1_ KNeighborsClassifier() :  0.5363100877762321
Hamming Loss_ KNeighborsClassifier() :  0.10887369132049983


Precision_ GaussianNB() :  0.1447261027964733
Rappel_ GaussianNB() :  0.6529255319148937
f1_ GaussianNB() :  0.23386626468891622
Hamming Loss_ GaussianNB() :  0.6340341100979399


Precision_ SVC() :  0.7484210824636358
Rappel_ SVC() :  0.46365248226950356
f1_ SVC() :  0.5621796778390171
Hamming Loss_ SVC() :  0.09747551502870652


Precision_ SGDClassifier() :  0.502076425622986
Rappel_ SGDClassifier() :  0.5323581560283688
f1_ SGDClassifier() :  0.515912582173863
Hamming Loss_ SGDClassifier() :  0.14163289429246875
```

Simplifiying the DataSet

Changes :

Never Used -> Never

Used over a Decade Ago -> Tryed

Used in Last Decade -> Tryed

Used in Last Year -> Tryed

Used in Last Month -> Frequently

Used in Last Week -> Frequently

Used in Last Day -> Frequently

New results :

```
Precision_ LinearSVC() :  0.6065315137813244
Rappel_ LinearSVC() :  0.618646572104019
f1_ LinearSVC() :  0.6120200756381468
Hamming Loss_ LinearSVC() :  0.259801024428684
```

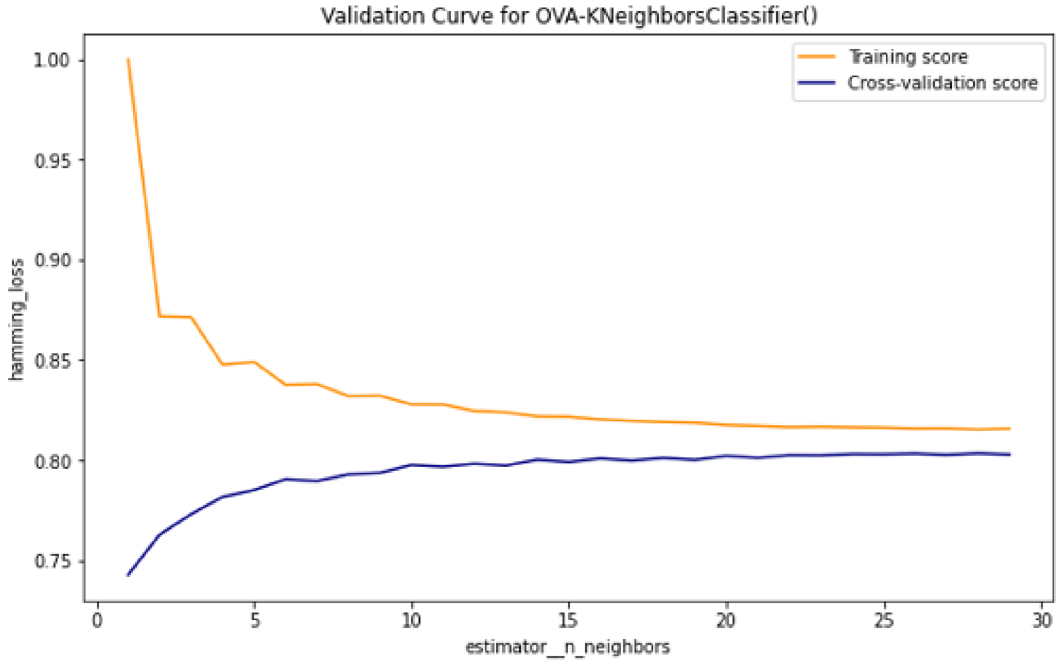
```
Precision_ KNeighborsClassifier() :  0.6991531498591105
Rappel_ KNeighborsClassifier() :  0.6326832151300237
f1_ KNeighborsClassifier() :  0.6626386295901975
Hamming Loss_ KNeighborsClassifier() :  0.2119286840031521
```

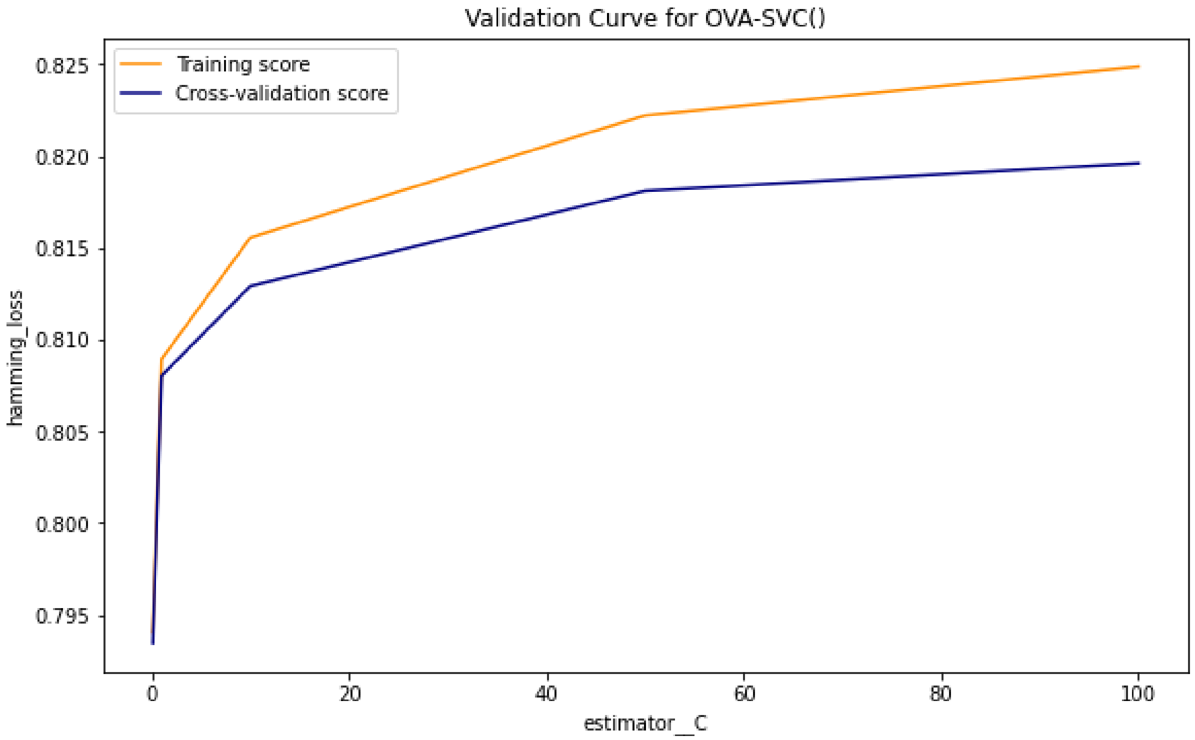
```
Precision_ GaussianNB() :  0.31729246558354324
Rappel_ GaussianNB() :  0.5821513002364066
f1_ GaussianNB() :  0.40842516965953446
Hamming Loss_ GaussianNB() :  0.5701832151300237
```

```
Precision_ SVC() :  0.7804863465435273
Rappel_ SVC() :  0.5992907801418441
f1_ SVC() :  0.6735699947301089
Hamming Loss_ SVC() :  0.18799251379038612
```

```
Precision_ SGDClassifier() :  0.6054365936418785
Rappel_ SGDClassifier() :  0.6385933806146572
f1_ SGDClassifier() :  0.6206683752769722
Hamming Loss_ SGDClassifier() :  0.25950551615445233
```

Validation Curve





Choosing hyperparameters

KNN:

- estimator__n_neighbors: [5, 7,10]
- estimator__weights: ['uniform', 'distance’]

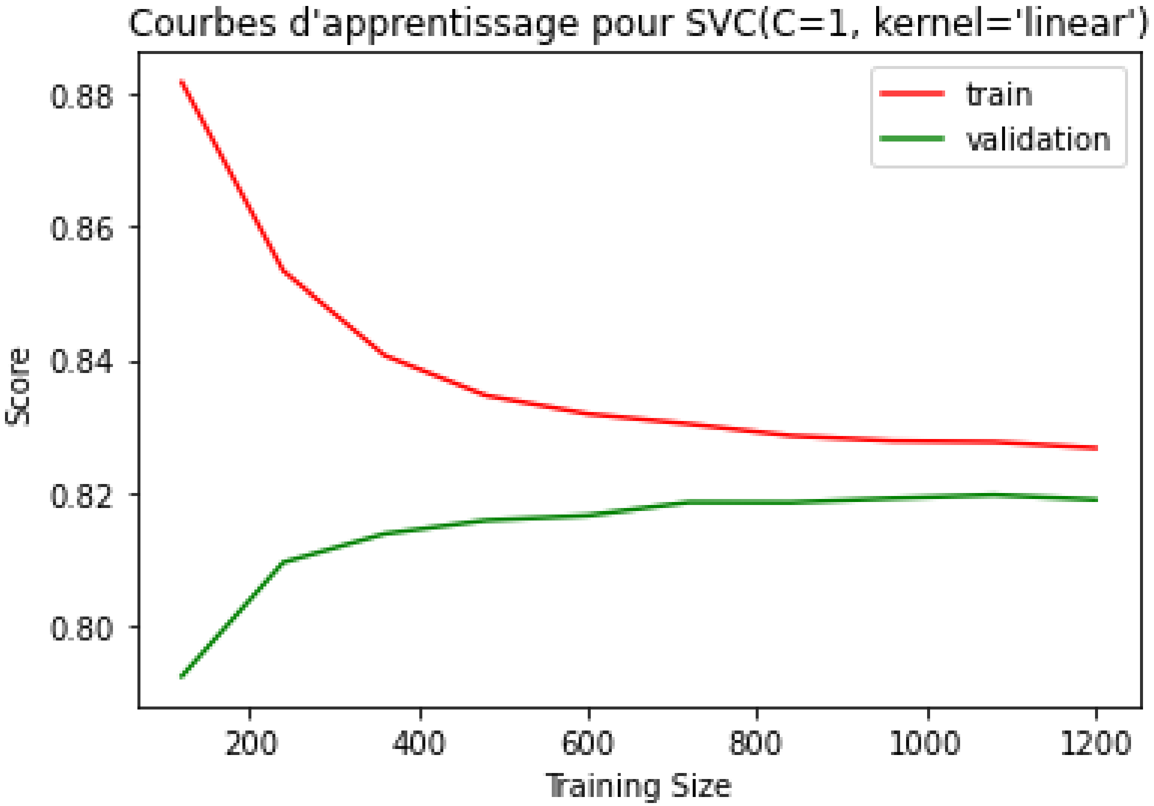
SVC:

- estimator__C: [1, 5, 10]
- estimator__kernel: ['linear', 'rbf']

Gridsearch Results

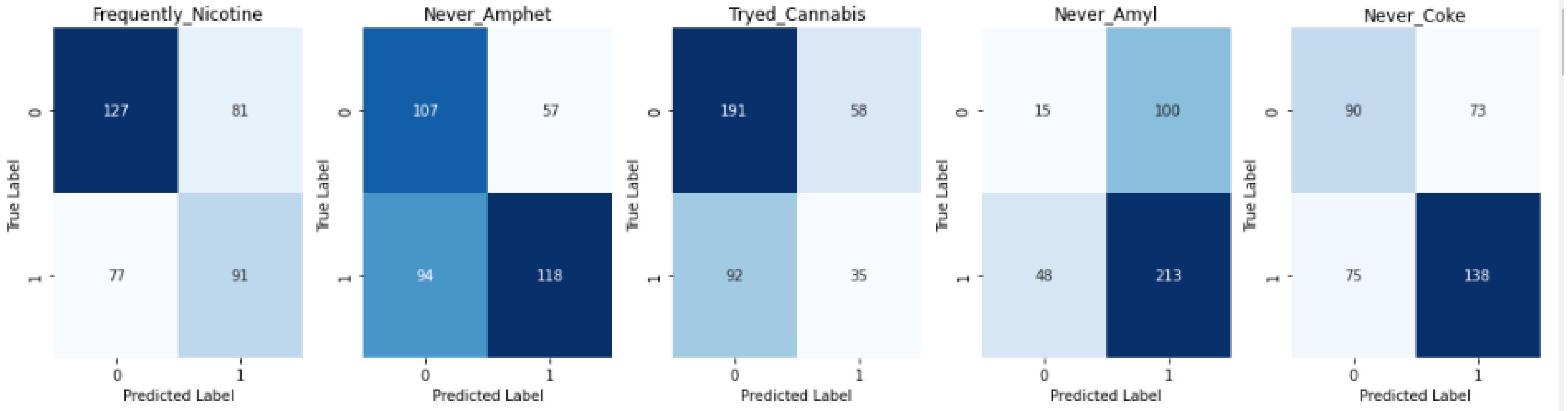
```
Meilleurs paramètres : {'estimator__n_neighbors': 7, 'estimator__weights': 'distance'}
Precision_ KNeighborsClassifier() :  0.6966359164927688
Rappel_ KNeighborsClassifier() :  0.6182033096926713
f1_ KNeighborsClassifier() :  0.653291343882303
Hamming Loss_ KNeighborsClassifier() :  0.2157702915681639
Meilleurs paramètres : {'estimator__C': 1, 'estimator__kernel': 'linear'}
Precision_ SVC() :  0.7687046302652364
Rappel_ SVC() :  0.6354905437352246
f1_ SVC() :  0.6921714562164204
Hamming Loss_ SVC() :  0.1828703703703703
```

Courbe d’apprentissage / Training Curve



The weaknesses

KNN:



SVC: