In [57]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [58]:
```
from sklearn.linear_model import LogisticRegression
```

In [59]:
```
df=pd.read_csv("Data.csv").dropna()
df
```

	row_id	user_id	timestamp	gate_id
0	0	18	2022-07-29 09:08:54	7
1	1	18	2022-07-29 09:09:54	9
2	2	18	2022-07-29 09:09:54	9
3	3	18	2022-07-29 09:10:06	5
4	4	18	2022-07-29 09:10:08	5
...	...	...	...	...
37513	37513	6	2022-12-31 20:38:56	11
37514	37514	6	2022-12-31 20:39:22	6
37515	37515	6	2022-12-31 20:39:23	6
37516	37516	6	2022-12-31 20:39:31	9
37517	37517	6	2022-12-31 20:39:31	9

37518 rows × 4 columns

In [60]:
```
df.dropna(inplace=True)
```

In [61]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 37518 entries, 0 to 37517
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   row_id     37518 non-null  int64 
 1   user_id    37518 non-null  int64 
 2   timestamp  37518 non-null  object
 3   gate_id    37518 non-null  int64 
dtypes: int64(3), object(1)
memory usage: 1.4+ MB
```

In [62]:
```
feature_matrix = df[['row_id','user_id']]
target_vector = df['gate_id']
```

In [63]:
```
feature_matrix.shape
```

Out[63]:
```
(37518, 2)
```

In [64]:
```
target_vector.shape
```

Out[64]:
```
(37518,)
```

In [65]:
```
from sklearn.preprocessing import StandardScaler
```

In [66]:
```
fs = StandardScaler().fit_transform(feature_matrix)
```

In [67]:
```
logr = LogisticRegression()
logr.fit(fs,target_vector)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear_model\_logistic.py:
763: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html (https://scikit-learn.org/stable/modules/preprocessing.html)
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)
  n_iter_i = _check_optimize_result(
```

Out[67]:
```
LogisticRegression()
```

In [68]:
```
feature_matrix.shape
```

Out[68]:
```
(37518, 2)
```

In [69]:
```
target_vector.shape
```

Out[69]:
```
(37518,)
```

In [70]:
```
from sklearn.preprocessing import StandardScaler
```

In [71]:
```
fs = StandardScaler().fit_transform(feature_matrix)
```

In [72]:
```
logr = LogisticRegression()
logr.fit(fs,target_vector)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear_model\_logistic.py:
763: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html (https://scikit-learn.org/stable/modules/preprocessing.html)
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)
  n_iter_i = _check_optimize_result(
```

Out[72]:
```
LogisticRegression()
```

In [73]:
```
observation=df[['row_id','user_id']]
```

In [74]:
```
prediction = logr.predict(observation)
prediction
```

Out[74]:
```
array([-1, -1, -1, ..., 16, 16, 16], dtype=int64)
```

In [75]:
```
logr.classes_
```

Out[75]:
```
array([-1,  0,  1,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16],
      dtype=int64)
```

In [76]:
```
logr.predict_proba(observation)[0][1]
```

Out[76]:
```
1.7263815682078809e-09
```