

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [2]: df=pd.read_csv(r"C:\Users\user\Downloads\C9_Data - C9_Data.csv")
df
```

Out[2]:

	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 [3]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 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: 1.1+ MB
```

```
In [6]: df.columns
```

Out[6]: Index(['row\_id', 'user\_id', 'timestamp', 'gate\_id'], dtype='object')

```
In [5]: df['user_id'].value_counts()
df['gate_id'].value_counts()
```

```
Out[5]: 4      8170
        3      5351
        10     4767
        5      4619
        11     4090
        9      3390
        7      3026
        6      1800
        13     1201
        12      698
        15      298
        -1      48
        8      48
        1       5
        16      4
        0       2
        14      1
        Name: gate_id, dtype: int64
```

```
In [8]: df1=df[['row_id', 'user_id', 'timestamp', 'gate_id']]
```

```
In [9]: x=df1[['row_id', 'user_id', 'gate_id']]
y=df1['gate_id']
```

```
In [19]: g1={'gate_id':{'15':0,'14':1,'13':2,'12':3,'11':4,'10':5,'9':6,'8':7,'7':8,'6':9,
:12,'1':13,'0':14,'-1':15}}
df1=df1.replace(g1)
print(df1)
```

	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 x 4 columns]

```
In [20]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=45)
```

```
In [21]: from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[21]: RandomForestClassifier()

```
In [22]: parameters = {'max_depth':[1,2,3,4,5],
                        'min_samples_leaf':[5,10,15,20,25],
                        'n_estimators':[10,20,30,40,50]}
```

```
In [23]: from sklearn.model_selection import GridSearchCV

grid_search = GridSearchCV(estimator=rfc,param_grid=parameters,cv=2,scoring='acc
grid_search.fit(x_train,y_train)
```

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\model\_selection\\_split.py:66  
6: UserWarning: The least populated class in y has only 1 members, which is less than n\_splits=2.

warnings.warn("The least populated class in y has only %d"

Out[23]: GridSearchCV(cv=2, estimator=RandomForestClassifier(),  
param\_grid={'max\_depth': [1, 2, 3, 4, 5],  
'min\_samples\_leaf': [5, 10, 15, 20, 25],  
'n\_estimators': [10, 20, 30, 40, 50]},  
scoring='accuracy')

```
In [24]: grid_search.best_score_
```

Out[24]: 0.8593921888319084

```
In [25]: rfc_best = grid_search.best_estimator_
```

```
In [26]: from sklearn.tree import plot_tree

plt.figure(figsize=(80,40))
plot_tree(rfc_best.estimators_[4],feature_names=x.columns,filled=True)
text(4528.727272727273, 181.19999999999982, gini = 0.441\nsamples = 74\nvalue = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 79, 0, 1\n0, 32, 2]')
```