Random Forest

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

In [2]:
```
df1=pd.read_csv(r"C:\Users\user\Downloads\C9_Data.csv")
df1
```

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]:
```
df1.columns
```

Out[3]:
```
Index(['row_id', 'user_id', 'timestamp', 'gate_id'], dtype='object')
```

In [4]:
```
df1.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 [7]:

```
df=df1.head(10)
df
```

Out[7]:

	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
5	5	18	2022-07-29 09:10:34	10
6	6	18	2022-07-29 09:32:47	11
7	7	18	2022-07-29 09:33:12	4
8	8	18	2022-07-29 09:33:13	4
9	9	1	2022-07-29 09:33:16	7

In [8]:

```
df['user_id'].value_counts()
```

Out[8]:

```
18    9
1     1
Name: user_id, dtype: int64
```

In [9]:

```
x=df[['row_id', 'gate_id']]
y=df['user_id']
```

In [10]:

```
g1={"g":{'g':1,'g':2}}
df=df.replace(g1)
print(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
5       5       18  2022-07-29 09:10:34       10
6       6       18  2022-07-29 09:32:47       11
7       7       18  2022-07-29 09:33:12        4
8       8       18  2022-07-29 09:33:13        4
9       9        1  2022-07-29 09:33:16        7
```

In [11]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,train_size=0.70)
```

In [12]:
```
from sklearn.ensemble import RandomForestClassifier

rfc = RandomForestClassifier()
rfc.fit(x_train,y_train)
```

Out[12]:
```
RandomForestClassifier()
```

In [13]:
```
parameters = { 'max_depth':[1,2,3,4,5],
    'min_samples_leaf':[5,10,15,20,25],
              'n_estimators':[10,20,30,40,50]
}
```

In [14]:
```
from sklearn.model_selection import GridSearchCV

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

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\model_selection\_split.py:
666: 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[14]:
```
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')
```

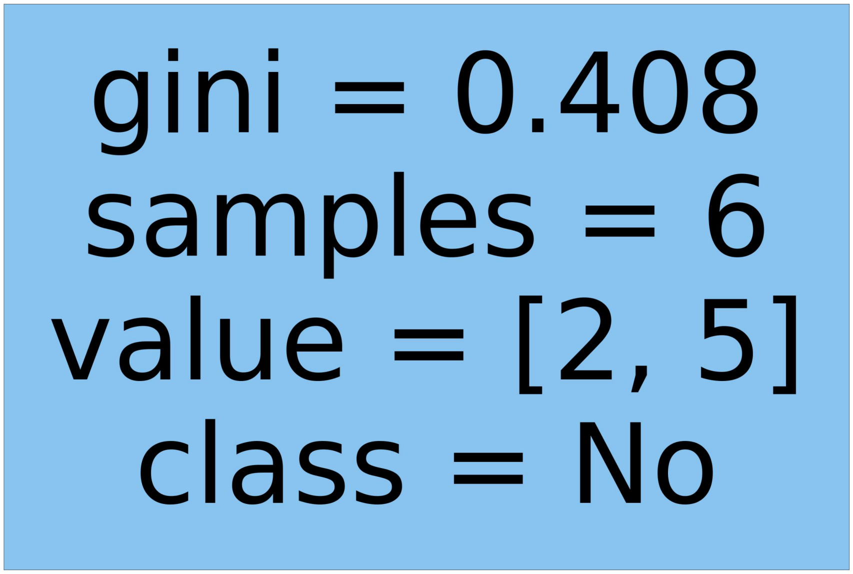
In [15]:
```
rf_best=grid_search.best_estimator_
print(rf_best)
```

```
RandomForestClassifier(max_depth=1, min_samples_leaf=5, n_estimators=10)
```

```
In [16]: from sklearn.tree import plot_tree

         plt.figure(figsize=(80,40))
         plot_tree(rf_best.estimators_[5],feature_names=x.columns,class_names=['Yes','N
```

```
Out[16]: [Text(2232.0, 1087.2, 'gini = 0.408\nsamples = 6\nvalue = [2, 5]\nclass = N
         o')]
```


gini = 0.408
samples = 6
value = [2, 5]
class = No