ML project

January 6, 2023

[1]:
```
import pandas as pd
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
```

[2]:
```
music_data = pd.read_csv('music.csv')
music_data
```

[2]:

	age	gender	genre
0	20	1	HipHop
1	23	1	HipHop
2	25	1	HipHop
3	26	1	Jazz
4	29	1	Jazz
5	30	1	Jazz
6	31	1	Classical
7	33	1	Classical
8	37	1	Classical
9	20	0	Dance
10	21	0	Dance
11	25	0	Dance
12	26	0	Acoustic
13	27	0	Acoustic
14	30	0	Acoustic
15	31	0	Classical
16	34	0	Classical
17	35	0	Classical

[3]:
```
X = music_data.drop(columns=['genre'])
X
```

[3]:

	age	gender
0	20	1
1	23	1
2	25	1
3	26	1
4	29	1

```
5    30       1
6    31       1
7    33       1
8    37       1
9    20       0
10   21       0
11   25       0
12   26       0
13   27       0
14   30       0
15   31       0
16   34       0
17   35       0
```

[4]:
```
Y = music_data['genre']
Y
```

[4]:
```
0        HipHop
1        HipHop
2        HipHop
3          Jazz
4          Jazz
5          Jazz
6     Classical
7     Classical
8     Classical
9         Dance
10        Dance
11        Dance
12     Acoustic
13     Acoustic
14     Acoustic
15    Classical
16    Classical
17    Classical
Name: genre, dtype: object
```

[5]:
```
model = DecisionTreeClassifier()
model.fit(X,Y)
music_data
```

[5]:
```
    age  gender      genre
0    20       1     HipHop
1    23       1     HipHop
2    25       1     HipHop
3    26       1       Jazz
4    29       1       Jazz
```

```
5    30       1       Jazz
6    31       1  Classical
7    33       1  Classical
8    37       1  Classical
9    20       0      Dance
10   21       0      Dance
11   25       0      Dance
12   26       0   Acoustic
13   27       0   Acoustic
14   30       0   Acoustic
15   31       0  Classical
16   34       0  Classical
17   35       0  Classical
```

[6]:
```
predictions = model.predict([[21,1],[22,0]])
predictions
```

[6]:
```
array(['HipHop', 'Dance'], dtype=object)
```

[7]:
```
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,train_size = 0.2)
model.fit(X_train,Y_train)
predictions = model.predict(X_test)
score = accuracy_score = (Y_test,predictions)
score
```

[7]:
```
(14     Acoustic
 7     Classical
 17    Classical
 1        HipHop
 4          Jazz
 5          Jazz
 12     Acoustic
 9         Dance
 3          Jazz
 15    Classical
 13     Acoustic
 8     Classical
 2        HipHop
 6     Classical
 11        Dance
 Name: genre, dtype: object,
 array(['Classical', 'Classical', 'Classical', 'Dance', 'Classical',
        'Classical', 'Dance', 'HipHop', 'Dance', 'Classical', 'Dance',
        'Classical', 'Dance', 'Classical', 'Dance'], dtype=object))
```

[8]:
```
tree.export_graphviz(model,out_file = 'music-recommender.dot',
                     feature_names = ['age' , 'gender'],
```

```
                        class_names= sorted(Y.unique()),
                        label='all',
                        rounded='True',
                        filled='True')
```

[ ]: