In [1]:
```
from sklearn import tree
import pandas as pd
import os
```

In [2]:
```
df = pd.read_csv("main_data_tree2.csv")
df.head(10)
```

Out[2]:

	EOG_quintile	percent_EDS	parent_unemployed	county_poc	median_inc_county
0	4	96.0	11.3	50.3	40433
1	4	96.0	8.3	21.6	53419
2	3	96.0	8.6	56.6	59329
3	4	96.0	19.3	74.2	35407
4	4	4.0	9.5	42.7	50112
5	1	4.0	2.8	11.5	64426
6	1	4.0	12.7	52.5	44015
7	1	4.0	4.0	39.1	80169
8	3	4.0	9.2	35.7	57710
9	1	4.0	4.0	39.1	80169

In [3]:
```
target = df["EOG_quintile"]
target_names = ["1", "2", "3", "4", "5"]
```

In [4]:
```
data = df.drop("EOG_quintile", axis=1)
feature_names = ["percent_EDS","parent_unemployed","county_poc","median_inc_county"]
data.head()
```

Out[4]:

	percent_EDS	parent_unemployed	county_poc	median_inc_county
0	96.0	11.3	50.3	40433
1	96.0	8.3	21.6	53419
2	96.0	8.6	56.6	59329
3	96.0	19.3	74.2	35407
4	4.0	9.5	42.7	50112

In [5]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(data, target, random_state=42)
```

In [6]:
```
clf = tree.DecisionTreeClassifier()
clf = clf.fit(X_train, y_train)
clf.score(X_test, y_test)
```

Out[6]:
```
0.4606741573033708
```

In [7]:
```
from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier(n_estimators=200)
rf = rf.fit(X_train, y_train)
rf.score(X_test, y_test)
```

Out[7]:
```
0.48635634028892455
```

In [8]:
```
sorted(zip(rf.feature_importances_, feature_names), reverse=True)
```

Out[8]:
```
[(0.6805577732473067, 'percent_EDS'),
 (0.12851117447507474, 'county_poc'),
 (0.1009517441954988, 'median_inc_county'),
 (0.08997930808211989, 'parent_unemployed')]
```

In [ ]:
```

```