Day 39

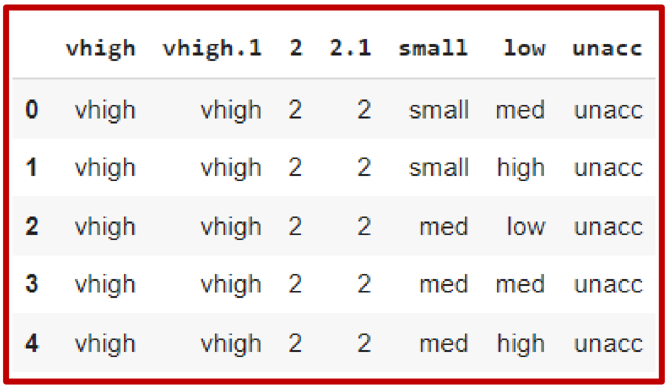
DIY

Q1. Problem Statement: Decision Tree Using the CART Algorithm

You are given a dataset, "`car_evaluation.csv.`" Load the dataset into a DataFrame without the header and rename the columns as the list given here - `['buying', 'maint', 'doors', 'persons', 'lug_boot', 'safety', 'class'].` Considering the `class` column as the target variable, perform the following tasks:

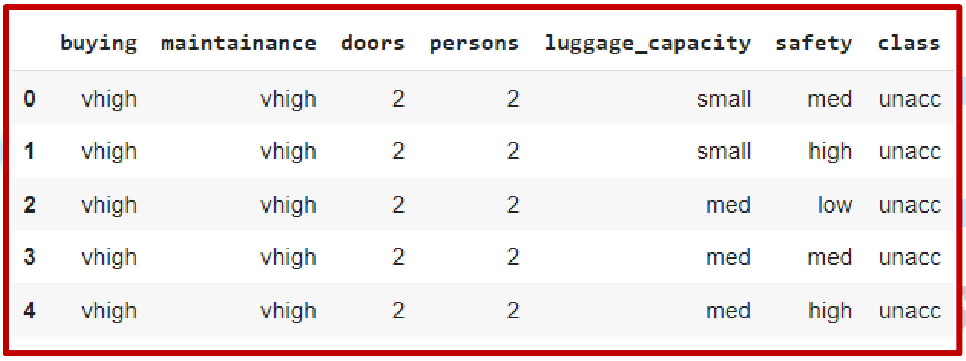
1. Explore the target column, `class` (our task is to predict how the car features affect the class of car as Very good, Good, Acceptable, or Unacceptable, which is why we have considered this column as the target column)
2. Declare feature vectors and the target variable
3. Split the data into test and train fragments using the `train_test_split()` function in an 80:20 ratio (80% train and 20% test)
4. Encode all the ordinal data into numeric values using the `category_encoders` library
5. Predicting the test results using a Decision Tree Classifier based on Gini Index criteria
6. Check the accuracy score of the model based on the Gini Index
7. Visualize the decision tree using Graphviz
8. Show how the importance of features affects the target variable
9. State the results and conclusion

Dataset:



	vhigh	vhigh.1	2	2.1	small	low	unacc
0	vhigh	vhigh	2	2	small	med	unacc
1	vhigh	vhigh	2	2	small	high	unacc
2	vhigh	vhigh	2	2	med	low	unacc
3	vhigh	vhigh	2	2	med	med	unacc
4	vhigh	vhigh	2	2	med	high	unacc

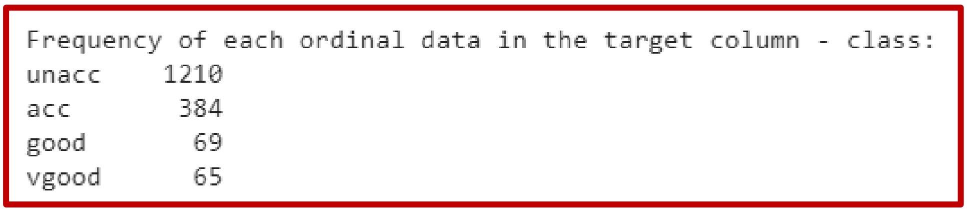
After renaming the columns with the list - `['buying', 'maint', 'doors', 'persons', 'lug_boot', 'safety', 'class']`



	buying	maintainance	doors	persons	luggage_capacity	safety	class
0	vhigh	vhigh	2	2	small	med	unacc
1	vhigh	vhigh	2	2	small	high	unacc
2	vhigh	vhigh	2	2	med	low	unacc
3	vhigh	vhigh	2	2	med	med	unacc
4	vhigh	vhigh	2	2	med	high	unacc

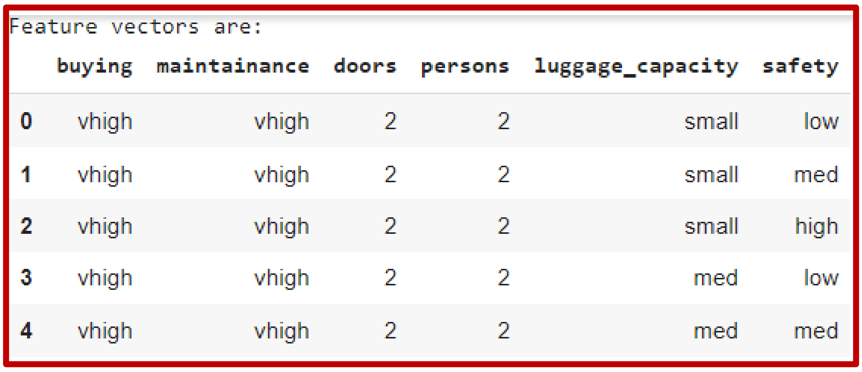
Sample Output:

1. Explore the target column, `Class` (our task is to predict how the car features affect the class of car as - Very good, Good, Acceptable, or Unacceptable, that is why we have considered this column as the target column)



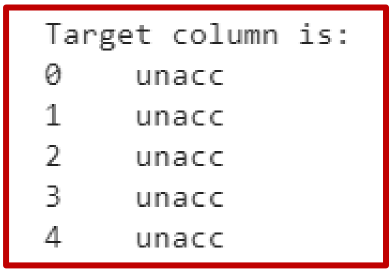
```
Frequency of each ordinal data in the target column - class:
unacc    1210
acc       384
good       69
vgood      65
```

2. Declare feature vectors and the target variable



Feature vectors are:

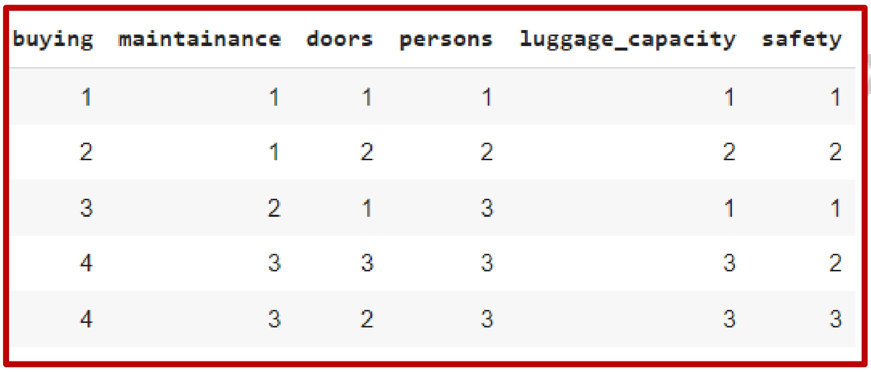
	buying	maintainance	doors	persons	luggage_capacity	safety
0	vhigh	vhigh	2	2	small	low
1	vhigh	vhigh	2	2	small	med
2	vhigh	vhigh	2	2	small	high
3	vhigh	vhigh	2	2	med	low
4	vhigh	vhigh	2	2	med	med



```
Target column is:
0     unacc
1     unacc
2     unacc
3     unacc
4     unacc
```

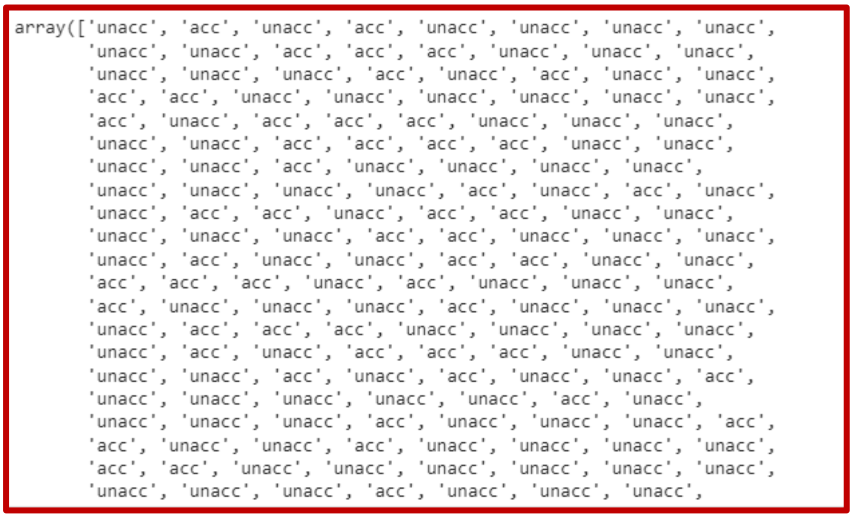
3. Split the data into test and train fragments using the `train_test_split()` function in an 80:20 ratio (80% train and 20% test)

4. Encode all the ordinal data into numeric values using the `category_encoders` library



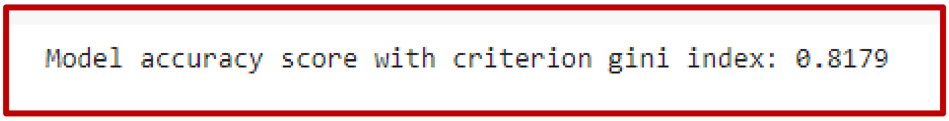
buying	maintainance	doors	persons	luggage_capacity	safety
1	1	1	1	1	1
2	1	2	2	2	2
3	2	1	3	1	1
4	3	3	3	3	2
4	3	2	3	3	3

10.Predicting the test results using a Decision Tree Classifier based on Gini Index criteria



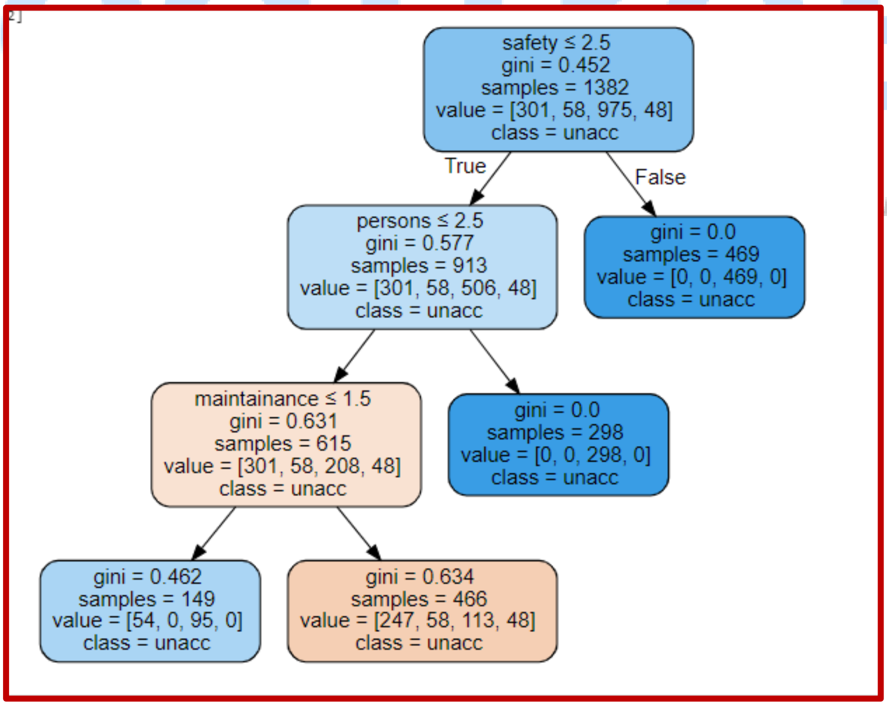
```
array(['unacc', 'acc', 'unacc', 'acc', 'unacc', 'unacc', 'unacc', 'unacc',
       'unacc', 'unacc', 'acc', 'acc', 'acc', 'unacc', 'unacc', 'unacc',
       'unacc', 'unacc', 'unacc', 'acc', 'unacc', 'acc', 'unacc', 'unacc',
       'acc', 'acc', 'unacc', 'unacc', 'unacc', 'unacc', 'unacc', 'unacc',
       'acc', 'unacc', 'acc', 'acc', 'acc', 'unacc', 'unacc', 'unacc',
       'unacc', 'unacc', 'acc', 'acc', 'acc', 'acc', 'unacc', 'unacc',
       'unacc', 'unacc', 'acc', 'unacc', 'unacc', 'unacc', 'unacc',
       'unacc', 'unacc', 'unacc', 'unacc', 'acc', 'unacc', 'acc', 'unacc',
       'unacc', 'acc', 'acc', 'unacc', 'acc', 'acc', 'unacc', 'unacc',
       'unacc', 'unacc', 'unacc', 'acc', 'acc', 'unacc', 'unacc', 'unacc',
       'unacc', 'acc', 'unacc', 'unacc', 'acc', 'acc', 'unacc', 'unacc',
       'acc', 'acc', 'acc', 'unacc', 'acc', 'unacc', 'unacc', 'unacc',
       'acc', 'unacc', 'unacc', 'unacc', 'acc', 'unacc', 'unacc', 'unacc',
       'unacc', 'acc', 'acc', 'acc', 'unacc', 'unacc', 'unacc', 'unacc',
       'unacc', 'acc', 'unacc', 'acc', 'acc', 'acc', 'unacc', 'unacc',
       'unacc', 'unacc', 'acc', 'unacc', 'acc', 'unacc', 'unacc', 'acc',
       'unacc', 'unacc', 'unacc', 'unacc', 'unacc', 'acc', 'unacc',
       'unacc', 'unacc', 'unacc', 'acc', 'unacc', 'unacc', 'unacc', 'acc',
       'acc', 'unacc', 'unacc', 'acc', 'unacc', 'unacc', 'unacc', 'unacc',
       'acc', 'acc', 'unacc', 'unacc', 'unacc', 'unacc', 'unacc', 'unacc',
       'unacc', 'unacc', 'unacc', 'acc', 'unacc', 'unacc', 'unacc',
```

5. Check the accuracy score of the model based on the Gini Index

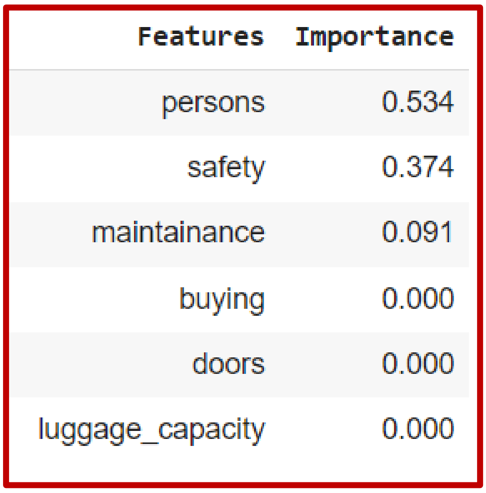


```
Model accuracy score with criterion gini index: 0.8179
```

6. Visualize the decision tree using Graphviz



7. Show how the importance of features affects the target variable



Features	Importance
persons	0.534
safety	0.374
maintainance	0.091
buying	0.000
doors	0.000
luggage_capacity	0.000