how you can use decision tree and random forest in sklearn on your own datasets

Decision TREE= supervised learning algorithm, but regression and classification problems

Note: The most popular attribute selection methods that we'll use in this course are information **gain** and **gini** index.

Overfitting is a main disadvantage of using Decision-Tree model

Car\_data . columns = ['buying', 'maint', 'doors', 'persons', 'lug\_boot', 'safety', 'class'] -> it is to add columns to our data

Notice that all of our variables are **ordinal categorical variables**

we can use sklearn's ordinal encoder, or manually map each unique value in each column to some number [0, n\_classes-1].

***Note: We can't use one hot encoding / get\_dummies here because that won't preserve the order.***

***train\_test\_split(X, y, test\_size)*** function. ->split our data into training and testing sets

Here, the training-set accuracy score is **0.7965** while the test-set accuracy is **0.7803.** These two values are quite comparable, so there is no sign of **overfitting**.

Why do you think the random forest performed better? **RANDOM FOREST is better**