

4. Classify

Now that we can make different decision trees, it's time to plant a whole forest! Let's say we make different 8 trees using bagging and feature bagging. We can now take a new unlabeled point, give that point to each tree in the forest, and count the number of times different labels are predicted.

The trees give us their votes and the label that is predicted most often will be our final classification! For example, if we gave our random forest of 8 trees a new data point, we might get the following results:

```
["vgood", "vgood", "good", "vgood", "acc", "vgood", "good", "vgood"]
```

Since the most commonly predicted classification was "vgood", this would be the random forest's final classification.

Let's write some code that can classify an unlabeled point!

Instructions

1.

At the top of your code, we've included a new unlabeled car named `unlabeled_point` that we want to classify. We've also created a tree named `subset_tree` that was created using bagging and feature bagging.

Let's see how that tree classifies this point. Print the results of `classify()` using `unlabeled_point` and `subset_tree` as parameters.

2.

That's the prediction using one tree. Let's make 20 trees and record the prediction of each one!

Take all of your code between creating `indices` and the `print` statement you just wrote and put it in a for loop that happens 20 times.

Above your for loop, create a variable named `predictions` and set it equal to an empty list. Inside your for loop, instead of printing the prediction, use `.append()` to add it to `predictions`.

Finally after your for loop, print `predictions`.

3.

We now have a list of 20 predictions — let's find the most common one! You can find the most common element in a list by using this line of code:

```
max(predictions, key=predictions.count)
```

Outside of your for loop, store the most common element in a variable named `final_prediction` and print that variable.

```

from tree import build_tree, print_tree, car_data, car_labels, classify
from collections import Counter
import random
random.seed(4)

# The features are the price of the car, the cost of maintenance, the number of doors, the number of people the car can hold, the size of the trunk, and the safety rating
unlabeled_point = ['high', 'vhigh', '3', 'more', 'med', 'med']

predictions = []
for i in range(20):
    indices = [random.randint(0, 999) for i in range(1000)]
    data_subset = [car_data[index] for index in indices]
    labels_subset = [car_labels[index] for index in indices]
    subset_tree = build_tree(data_subset, labels_subset)
    predictions.append(classify(unlabeled_point, subset_tree))
final_prediction = max(predictions, key=predictions.count)
print(predictions)
print(final_prediction)

```

```

['acc', 'unacc', 'acc', 'unacc', None, 'acc', 'acc', 'unacc',
'unacc', None, 'acc', 'unacc', 'acc', 'acc', 'acc', 'acc',
'acc', 'unacc', None, 'acc']
acc

```