CSci 4270 and 6270
Computational Vision,
Spring Semester, 2021
Lecture 13 Exercise — Making the Random Forest Decision
Due: Wednesday, March 17 at 11:59pm EST

As discussed in class, a random forest is a set of random decision trees. Each tree evaluates a data vector $\mathbf{x}$ by applying the classifier associated with each node and moving left or right down the tree until reaching a leaf node. Each leaf node is the count of the number of training data vectors for each class that reached that particular leaf node. Suppose for tree i that the leaf node reached has counts $n_{i,0}, \ldots, n_{i,C-1}$, where C is the number of classes. For example, if $C = 3$ these values might be

$$3, 4, 13$$

meaning that 3 training examples from class 0 reached the leaf, 4 from class 1 reached the leaf and 13 from class 2 reached the leaf. Clearly, class 2 is the most likely based on these values. Interpreted probabilistically, it has a $13/(3 + 4 + 13) = 0.65$ probability of being the right class, whereas the other two classes have 0.15 and 0.2 probability of being correct.

When combining across multiple trees, there are many possible schemes for making a decision. Here are three possible schemes:

1. Sum the count for each class across all trees, and convert to probabilities.

2. Convert to probabilities at each tree (one leaf for each tree!), as we just did, and average the probabilities.

3. Make a decision at each tree about the most likely class, and then choose the class with the highest number of these "votes". (If there are ties, split the vote across the classes that tied.) Probabilities are computed from the votes.

Here's an example to illustrate. If the votes for the selected leaves at $N = 4$ trees are

```
3  4 13
4  2  0
6  0  4
5  5  5
```

Then

- For Scheme 1, we have 18 votes for Class 0, 11 votes for Class 1, and 22 votes for Class 2, so Class 2 is the decision with probability $22/(22 + 18 + 11) \approx 0.43$.

- For Scheme 2, Class 0 is the decision because its probabilities are 0.15, 0.67, 0.6 and 0.33, giving an average of ≈ 0.44. The other classes are lower with values averages of ≈ 0.22 for Class 1 and ≈ 0.35 for Class 2.

- For Scheme 3, Class 0 is again the decision because it has 2.33 votes (highest number for two trees, and tied in a third). Class 0 has 0.33 votes and Class 2 has 1.33 votes. Class 0's final probability is $2.33/4 \approx 0.58$.

Your job is to implement each of these voting schemes. You are given a text file that has the count data as shown above, with C votes per line and N lines. (You may assume that the data are correctly read in for you.) You should have three lines of output for each input file giving the class decision and its probability. For the above example, the output should be

```
Class 2: 0.43
Class 0: 0.44
Class 0: 0.58
```

As usual, we have provided starting code.

It is possible to do the entire exercise without for loops, and I encourage you to try. My solution is 12 lines plus the print statements; maybe you can find something shorter. You will not be penalized, however, for using loops.

Finally, note that our schemes are simple instances of a very challenging research topic in AI: combining voter preferences. Several faculty in the CS department are leaders in this area. In work on random forests, as in other machine learning problems, the choice between schemes is often left to experimental evaluation during the "validation" step of training.