Decision Trees for Classification and Ensemble Methods Activity (30-40min)

Classes Used:

- Data 221: Introduction to Machine Learning: Concepts and Applications (for minors)

Learning Objectives:

- Construct a decision tree classifier from labeled training data.
- Select informative features for node splits based on class separation.
- Evaluate classifier performance using accuracy and, optionally, precision and recall.
- Compare training and testing performance to recognize the effects of overfitting and generalization.
- Explain how variability in training data and available features leads to different decision trees.
- Describe how ensemble methods improve predictive performance by combining multiple diverse models.

Materials Used:

- I use the Animal Cards set from this activity created for use at Carnegie Mellon University (Wong and Rosenthal, 2024). But this can be adjusted for other sets of data, or the instructor can create their own.
    - I printed, laminated, and cut out 5 sets of these cards for use with 5 groups of students. The number can be adjusted based on class size.
    - I split each of the 5 sets into training and testing cards in two different pouches (10 total) as follows:
        - Group 1: random split (10 in test set)
        - Group 2: all 'abnormal animals' in test set plus random animals to make 10, rest in training set
            - 'Abnormal animals' are those whose feature combinations make them difficult to classify using simple rules (e.g., dolphins have fins but are mammals rather than fish).
        - Group 3: all 'abnormal animals' in test set plus random animals to make 10, rest in training set
        - Group 4: random split (10 in test set)
        - Group 5: 4 or 5 'abnormal animals' in test set plus random animals to make 10, rest in training set

- Each of the 5 training pouches included the training cards, all arrows, X's and thumbs-up and selected features to split on. Features were assigned to groups as follows:
    - Group 1: all
    - Group 2: makes milk, flying, tail
    - Group 3: fins, feathers, legs
    - Group 4: legs, tail, feathers
    - Group 5: flying, tail, legs

Directions:

1. After giving some basic instruction on the format of decision trees, I use this activity to help them internalize the concepts of entropy, splitting criteria, and basic classification metrics by building a decision tree that predicts the animal class (e.g., mammal, bird, or fish)
2. I split the students into 5 groups and give each group an area to work. This works best if they have access to tables, but I've also had students move to the floor for more room.
3. I give each group a training packet and ask them to build the best tree possible for the training data they were provided. I tell them that they do not need to explicitly calculate entropy or information gain to build their tree and they should stop splitting when no remaining feature improves classification or when all leaves are pure. I also ask them to calculate and write on the chalk/white board the training accuracy that they achieve. Sometimes, depending on the amount of time we have, I also ask them to calculate precision and recall for the mammal class.
4. I walk around the room, keeping track of student progress. As I notice groups beginning to finish their trees, I give them the testing packet and ask them to, keeping training and testing data separate, calculate the testing accuracy (and sometimes precision and recall). They are asked to write this on the chalk/white board as well. I also remind them that they should not change their trees at all based on the testing data.
5. We have a class discussion about the activity. Often this discussion is student prompted as some groups will notice that they had a more restrictive set of features, more challenging train/test split, etc. than others and note that as they write down their metrics. This discussion naturally introduces concepts of model variance, feature importance, train/test generalization, and overfitting.
6. The next class period, I typically continue my lesson on decision trees which leads into the topic of ensemble methods. I find that this is a great time to remind students of the activity with some of the following prompts:

- If we randomly selected one group's decision tree, how would you expect its performance to compare with combining predictions from all five trees using majority vote?
- How did the available feature set influence the structure of your tree? Which features appeared most informative?
- Some groups achieved higher testing accuracy than others. What factors contributed to these differences?
- Did any group appear to overfit the training data? How can you tell?
- Why might combining many imperfect trees produce a better classifier than relying on a single "best" tree?
- Which ideas from this activity resemble the behavior of a random forest?
- Would you expect the majority vote to help if all five groups had built identical trees? Why or why not?

Instructor Notes:

- I find that this works best with 5 or fewer students per group.
- Each group receives a different subset of available features (and optionally a different train/test split) so that the resulting decision trees differ, setting up a discussion of model variance and ensemble methods.
- Students sometimes worry they found the "wrong" tree. It is helpful to remind them that multiple decision trees may correctly classify the training data; students should focus on producing a reasonable tree rather than matching another group's structure.
- This activity can be done before full introduction of decision trees (though students will need some brief introduction) or right before discussion of ensemble methods.
- You could extend this activity by asking students to calculate entropy or information gain, asking them to calculate more classification metrics, or asking them to calculate the accuracy of a random forest build using their trees.
- You could simplify the activity by removing one of the three classes and focusing on binary classification only.
- Though each feature will only be used once per tree, I include extras in the packets as there are extras included in the printout and small things tend to go missing over time.