CSE 416: Introduction to Machine Learning
Assignment #4
Instructor: Pemi Nguyen
due: Sunday, May 1, 2022, 11:59 pm.

Instructions:

- **Answers:** Please provide detailed yet concise explanations for your work.
- **Turn-in:** Do not write your name on your pages (your Gradescope account will identify you to us) and do not include a copy of the exercise's question in what you turn in. You must use Gradescope to upload your written homework solutions. You will submit a single PDF file containing your solutions to all the exercises in the homework. You must follow the Gradescope prompts that have you link exercise numbers to your pages. **We do not accept handwritten solutions**, so please typeset them, such as Microsoft Word or LaTeX editor.

1. Please determine whether the following practices are good / correct or not. Please explain why in either case.

   (a) *(2 points)* Sasha sees that the training classification error of his decision tree is 10%, but the validation classification error is 40 %. He wants to regularize the decision tree by fine-tuning the number of minimum training examples to split on a leaf node. Currently, in order for a leaf node to proceed with a split, it needs to have at least 30 examples. He thinks that reducing that number from 30 to 10 will improve the predictive performance of his model.

   Note: To understand what the metric minimum examples in a node to split means, let's take a look at an example. If the current leaf node has 20 examples, in the case of 30 minimum examples, the node will stop splitting because $20 < 30$. However, in the case of 10 minimum examples, the node will continue splitting because $20 > 10$.

   (b) *(2 points)* Wuwei works as a data scientist for an education company to assess students' surroundings on their academic performance to decide whether additional support should be given to a student or not. Her dataset contains columns with features such as "Type of high school a student goes to", "Are their parents still together?", "Personality type". She claims that using a decision tree approach might be a good approach since she doesn't have to preprocess the training data.

   (c) *(2 points)* Aric's computer has 8 cores in the CPU, and each core can be responsible for a parallel task. He plans to use all of these for training a random forest classifier. In building it, on each core, he makes the exact copy of the original training dataset and trains a decision tree based on that. In the end, he collects the majority votes from all trees to decide the final output class.

   (d) *(2 points)* Aware of the weak predictive performance with using a single decision tree, Tanmay thinks that an ensemble model might help improve his classifier model. The issue is, he doesn't have a lot of time for training since he's on an urgent deadline, but he

has enough computing resources. In deciding between Random Forest and Adaboost, he chooses Random Forest to save time for training.

(e) *(2 points)* Zeynep is a real estate agent. Instead of predicting house prices, she wants to run a classifier model on the housing price dataset to decide whether she will recommend a house to her clients or not. She's interested in the decision tree classifier due to how interpretable it is. She believes that normalization is an important step in training any machine learning model. For the decision tree classifier, she thinks normalizing the numeric features like "number of bedrooms", "area of a house", etc. will significantly improve its performance because we can put these features on the same scale.

2. *(3 points)* Draw a decision tree (at least 2 levels) that illustrates a guideline for something you're interested in. Try to make it as informative as possible. Briefly describe in 2-3 sentences you formulate the decisions at each step and how it can be used in real life. Try not to use the examples from lectures. This will be graded based on effort.

3. *(2 points)*

   Based on our implementation of the two models from the programming assignment (Decision Tree and Random Forest), which model, when using max depth of 25, would we expect to perform better in the future? Briefly explain.

4. **Extra credit**

   Below is a dataset which outlines different weather features that can affect a football team's decision to host a match or not.

Day	Outlook	Humidity	Temperature	Decision
1	Sunny	High	Hot	No
2	Sunny	High	Hot	No
3	Overcast	High	Hot	Yes
4	Rain	High	Mild	Yes
5	Rain	Normal	Cool	Yes
6	Rain	Normal	Cool	No
7	Overcast	Normal	Cool	Yes
8	Sunny	High	Mild	No
9	Sunny	Normal	Cool	Yes
10	Rain	Normal	Mild	Yes
11	Sunny	Normal	Mild	Yes
12	Overcast	High	Mild	Yes
13	Overcast	Normal	Hot	Yes
14	Rain	High	Mild	No

   (a) *(1 point)* In the decision tree algorithm from lecture, we decide on the best split for a leaf node by computing the classification errors of stumps from different features

and choosing the one that yields the smallest classification error for splitting. What are the classification errors when splitting the dataset on **Outlook**, **Humidity** and **Temperature**? What is interesting about these results?

Note: To earn full credit, you have to show to show detailed mathematical calculations from intermediate steps.

(b) *(2 points)*

An improved way to split on a node is to use a metric called Information Gain, which is used in the ID3 algorithm, one of the most common implementations for Decision Tree. For the best split on a leaf node, we choose the feature that can cause the greatest reduction in the entropy from a dataset, or **maximize the information gain**.

Entropy is a measure of the amount of uncertainty in the dataset, which is defined by:

$$E(S) = -\sum_{i \in C} p_i \log_2(p_i)$$

where:

- C is the set of output classes for a dataset S
- p_i is the probability of randomly picking an element of class i from set S (i.e. the proportion of examples made up of class i in dataset S)

We can define information gain as a measure of how much information a feature provides about a class, which helps to determine the order of attributes in the nodes of a decision tree. It is determined by the difference between the entropy of a parent node with dataset S and each child nodes with subdataset S_V from a split on feature A:

$$IG(S, A) = E(S) - \sum_{V \in A} p_V E(S_V)$$

where:

- A is a feature of the dataset S (that we decide to consider a split on)
- V is a value in feature A
- S_V is a sub-dataset of S that contains all examples which have value V
- p_V is the proportion of examples from S that has value V

Build a full decision tree that aims to maximize the information gain in each split and give a visualization the tree. Your tree must satisfy these stopping requirements:

- The tree has at most 2 levels.
- If a node doesn't have any incorrectly classified examples, we can stop early.

Note: To earn full credit, you have to show to show detailed mathematical calculations from intermediate steps.