Decision Trees

COSC 410: Applied Machine Learning
Fall 2025
Prof. Forrest Davis

September 9, 2025

Warm-up

1. Talk to the person next to you about your favorite season.
2. Consider the decision tree in Figure 1. For a new data point with a petal length of 3.0 cm and petal width of 1.78 cm, what class would you predict?

Logistics

- Codelet 1 is out on PyTorch and KNN
- Office Hours today 3-6PM 331 Bernstein

Learning Objectives

- Utilize decision trees
- Calculate Gini scores
- Apply the CART algorithm

Summary: We cover the basic use and construction of decision trees using the CART algorithm. We conclude with a brief sketch of random forests.

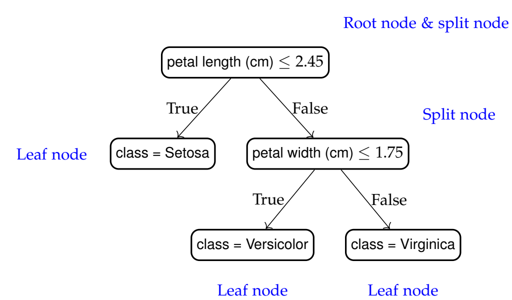

Figure 1: Example decision tree for classifying flowers based on petal dimensions and adapted from Chapter 6 of Géron [2023].

Motivating a Loss

IN FITTING A DECISION TREE, we are trying to find splits of our data that minimize 'impurity' or that minimize the uncertainty in our data (i.e., minimize our system's entropy). Let's understand what we are trying to do with an example.

Question

For the data in Figure 2, quantify the chance of misclassification based on the distribution of class labels. To do this, consider answering questions like if you selected a point that was a blue triangle, what is the likelihood you misclassify this point if you guessed based on the distribution of labels?

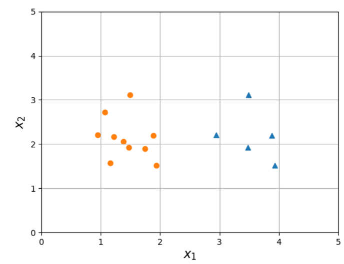

Figure 2: Sample data with to features, x_1 and x_2 and belonging to one of two classes, either orange circles or blue triangles.

Utilizing a Loss: Information Gain

WE ARE MOVING TOWARDS THE USE OF GINI IMPURITY as a loss function. Let's see what makes it a useful loss function for decision trees by considering an example.

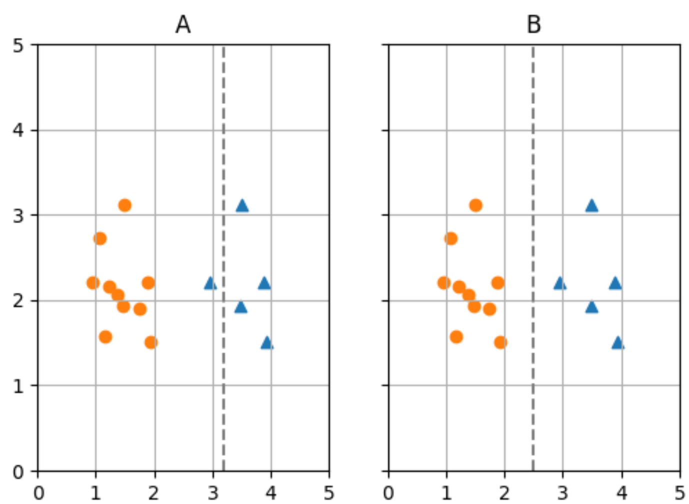

Figure 3: Two possible decision boundaries for the same sample data.

Question

Consider the two potential decision tree boundaries in Figure 3.

1. Calculate the Gini impurity for the left and right regions for each decision boundary.

Question

Consider the two potential decision tree boundaries in Figure 3.

2. Calculate the CART cost function value for both decision boundaries.

Question

Consider the two potential decision tree boundaries in Figure 3.

3. How much information did the system gain with each decision boundary?

Let's make this more abstract to ensure the tendencies of this loss function are clear. Consider a binary classification task. Three decision boundaries with the following properties are generated. Determine their Gini values.

Question

1. All points belong to class 1
2. Half of the points belong to class 1
3. None of the points belong to class 1

CART Algorithm

WITH OUR LOSS FUNCTION IN HAND, let's state the algorithm we will use to optimize our decision tree model. It's called the CART Algorithm. It is comprised of the following steps:

1. Create an initial root node that contains all of the data
2. For each node,
   (a) Calculate the Gini impurity for the current node
   (b) For each input feature, calculate the Ginit impurity for all possible thresholds.
   (c) The feature, threshold pair which has the minimum Gini impurity wins.
   (d) Split the data based on the chosen feature, threshold pair. Create new nodes.
3. Repeat Step 2 until some stopping criteria is met

Question

1. This a greedy algorithm? How do you know?
2. What are some possible stopping criteria
3. Why might we want to stop it early?

Learning a Decision Tree with Continuous Data

WITH YOUR SMALL GROUP WORK, fit a decision tree to the sample data, which has one continuous feature x and a binary label y.

sample	x	y
1	1.1	0
2	1.4	0
3	1.6	0
4	2.0	1
5	2.2	1
6	2.5	1
7	2.9	1
8	3.2	0
9	3.8	0

Learning a Decision Tree with Categorical Data

WITH YOUR SMALL GROUP WORK, fit a decision tree to the sample data, which has one discrete feature x and a binary label y.

sample	C	y
1	A	1
2	A	1
3	A	0
4	B	0
5	B	0
6	B	0
7	C	1
8	C	0
9	C	1

Limitations with Decision Trees

I HOPE IN SOLVING THESE TWO EXAMPLE PROBLEMS you can start to get a feel for possible limitations with decision trees.

Question

What are some limitations with Decision Trees

Random forests is an approach to modeling using decision trees that can overcome some limitations of vanilla decision trees. I'll just sketch the basic idea. Instead of relying on one deep tree, our aim is to build a collection of many trees and combine their predictions (perhaps with majority vote for classification, and averaging for regression). We build these trees by sampling a random subset of the training data, which adds variation across the trees.[1] Additionally, we interject more randomness by only considering a random subset of the input features for each trees.

[1] There are two broad ways to sample. We could sample with replacement, called **bagging**, or without replacement, called **pasting**.

Combining the predictions of a collection of models, of the same type or of different types, is more broadly called **ensemble** learning.

Before Next Class

- Read from Dive into Deep Learning linked on the course website
- No quiz
- Make progress on Codelet 1

References

Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly, Sebastopol, CA, third edition. edition, 2023. ISBN 978-1-09-812247-8.