

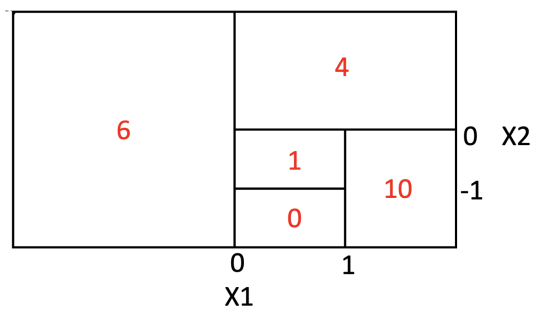
HW 05: Trees

Due Thursday, Nov. 10 at 11:59pm

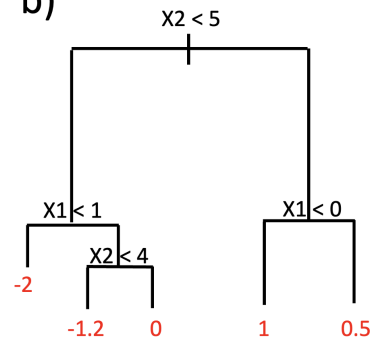
Exercise 1

- Sketch the tree corresponding to the partition of the predictor space illustrated in Figure a. The red numbers inside the boxes indicate the mean of the training Y within each region.
- Create a diagram similar to Figure a using the tree illustrated in Figure b. You should divide up the predictor space into the correct regions, and indicate the mean for each region.
- Suppose I have new a observation with predictors ($X1 = 1.5, X2 = 4.5$). What is your estimate \hat{y} using the tree in Figure a? What about when using the tree in Figure b?

a)



b)



Exercise 2

In this exercise, we will step through an example of the greedy approach for classification tree using Gini index. Suppose we are trying to classify a person's undergraduate **major** into either "CS", "Econ", or "Math", based on two predictors: their preferred programming **language** and their **salary** two years post-graduation (in \$10,000s). Our data is as follows:

salary	language	major
9	Python	Math
10	Python	Math
11	R	Econ
12	R	Econ
12	Python	CS
13	R	Econ
15	Python	CS
17	Python	CS

We will perform recursive binary splitting using a greedy approach, where the splits are chosen based on which split yields the best node purity at each step. We will use the Gini index to define node purity:

$$G = \sum_{k=1}^K \hat{p}_{mk}(1 - \hat{p}_{mk})$$

where \hat{p}_{mk} represents the proportion of training observations in the m -th region that are from the k th response class. Region m corresponds to the segment of predictor space under the given split. At every step, we choose the split that yields the lowest impurity, averaged across the child nodes.

For a given split on predictor X_j , we calculate the Gini indices for both the left and right child nodes. Call these G_l and G_r , respectively. Then take a *weighted average* of the Gini indices to determine the overall quality of this particular split. The weighted average is $w_l G_l + w_r G_r$, where w_l is the proportion of observations in the parent node that fall into the left child node, and w_l is the proportion of observations that fall into the right child node (note: for any given split, $w_l + w_r = 1$). We will choose the split that yields the *lowest weighted average*.

If a given node is completely pure (i.e. the Gini index of that node is 0), set it to be a terminal node.

For quantitative X_j , we typically consider a range of s values to segment the predictor and choose the s that yields the lowest impurity. For the purposes of this homework, we will not take this approach. Instead, let s be the median value in the current segment of the predictor space. If you have an even number of observations n , let the s be the average of the two middle values. Example: if the predictor values for X_j are 1, 4, 3, 2, then $s = 2.5$. If the predictor values for X_j are 3, 2, 5, 4, 1, then $s = 3$.

- We will begin by constructing the root node. Calculate the weighted Gini indices for each of the candidate splits **salary** < s (where s is defined above) and **language** = "Python".
- Based on your weighted Gini indices in (a), what is the first split (root node) in your decision tree?
- Finish building your tree. Once you are finished, draw your resulting tree. Don't forget to add the predicted classes for each of the terminal nodes.
- Why do you think we took a weighted average of the Gini indices of child nodes here, but we didn't take a weighted average of the RSS of the child nodes in the regression tree?