Exercise 1. A regression tree leaf: what value should it predict?

Consider the following one-dimensional regression dataset.

x	1	2	3	4	5	6
y	2	2	3	8	9	9

Suppose a regression tree has *no split*, so all six observations belong to a single leaf. The leaf predicts the same constant value c for every observation.

1. Compute the mean of the six target values.
2. Using the mean as the prediction, compute
$$\text{RSS} = \sum_{i=1}^{6} (y_i - c)^2.$$
3. Try the alternative constant predictions $c = 4$ and $c = 7$. Compute the RSS in each case.
4. Which of the three values gives the smallest RSS?
5. Based on this experiment, formulate a rule for the value predicted by a regression-tree leaf when squared error is used.

Extension. Replace the last target value 9 by 15. Recompute the mean. Which observation has the strongest influence on the leaf prediction?

Exercise 2. Choosing the best split in a regression tree

Use the same data:

x	1	2	3	4	5	6
y	2	2	3	8	9	9

A split at threshold t sends observations with $x < t$ to the left child and the others to the right child. For each child, use the mean target value in that child as its prediction.

Complete the following table for all possible thresholds.

t	left mean	right mean	RSS left	RSS right	total RSS
1.5					
2.5					
3.5					
4.5					
5.5					

1. Which threshold gives the smallest total RSS?
2. Draw the resulting depth-1 regression tree.
3. Write the prediction rule $\hat{y}(x)$ produced by this tree.
4. Compare the total RSS after the split with the RSS of the single-leaf model from Exercise 1.

Hint. For a given threshold, first compute the mean in each child, then compute squared deviations from that local mean.

Exercise 3. Growing a depth-2 regression tree

Start from the best first split found in Exercise 2.

x	1	2	3	4	5	6
y	2	2	3	8	9	9

Now allow *one additional split inside each child*, whenever possible.

1. For the left child, list all admissible thresholds and compute the best one.
2. For the right child, list all admissible thresholds and compute the best one.
3. Draw the resulting depth-2 tree.
4. For each terminal leaf, write its predicted value.
5. Compute the training RSS of the depth-2 tree.
6. Compare it with the depth-1 tree. Did the training error decrease?
7. Suppose the tree is allowed to keep splitting until each leaf contains one observation. What will the training RSS become?

Discussion. Why can a tree with extremely small training error still perform poorly on new data?

Exercise 4. Classification leaves and class probabilities

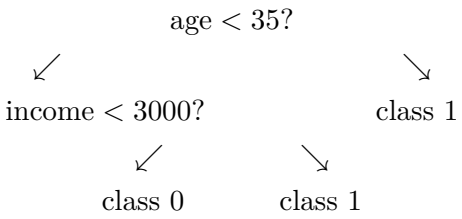
Consider the binary classification dataset

x	1	2	3	4	5	6	7	8
class	0	0	0	1	0	1	1	1

Suppose all eight observations belong to a single leaf.

1. Count the number of class-0 and class-1 observations.
2. Estimate $P(Y = 1)$ using the empirical class proportion in the leaf.
3. If the classification rule is "predict the majority class", what happens here?
4. Suppose the last class label is changed from 1 to 0. What class would the leaf predict now? What would the estimated probability $P(Y = 1)$ be?
5. Suppose instead that the first class label is changed from 0 to 1. Answer the same questions.

Small prediction exercise. Consider this tree:



Predict the class for: $(30, 2500)$, $(30, 4200)$, $(50, 2400)$, $(40, 6000)$, where each pair is (age, income).

Exercise 5. Gini impurity and the best classification split

For a binary node containing proportions p_0 and p_1, define the Gini impurity

$$G = 1 - p_0^2 - p_1^2.$$

First compute the Gini impurity of the following three nodes:

Node	class labels
A	$0, 0, 0, 0$
B	$0, 0, 1, 1$
C	$0, 0, 0, 1$

Now return to the dataset

$$(0, 0, 0, 1, 0, 1, 1, 1)$$

for $x = 1, 2, \ldots, 8$.

For a split at threshold t, compute the weighted impurity

$$G_{\text{split}} = \frac{n_L}{n} G_L + \frac{n_R}{n} G_R.$$

1. Compute G_{split} for $t = 3.5$.
2. Compute it for $t = 4.5$.
3. Compute it for $t = 5.5$.
4. Which of these three splits is best?
5. Why should a classification tree prefer nodes with low Gini impurity?

Exercise 6. Tree depth, perfect fitting, and overfitting

Consider the training dataset

x	1	2	3	4	5	6	7	8	9	10
class	0	0	0	0	1	0	1	1	1	1

The point $x = 6$ looks unusual compared with the general pattern.

1. Find a simple depth-1 split that classifies most observations correctly.
2. Compute its training accuracy.
3. Can a deeper tree isolate the observation at $x = 6$ and achieve 100% training accuracy? Sketch such a tree.
4. Suppose new observations near $x = 6$ usually belong to class 1. Which model would you expect to generalize better: the simple tree or the perfectly fitting deep tree?
5. Explain, in your own words, the difference between *training error* and *generalization error*.

Extension. Imagine that the class label at $x = 6$ was caused by a measurement error. Which tree reacts more strongly to this single bad label?

Exercise 7. Bootstrap samples and bagging

A training set contains five observations

$$\{A, B, C, D, E\}.$$

A bootstrap sample has the same size as the original dataset, but observations are drawn *with replacement*.

1. Which of the following are valid bootstrap samples of size 5?
   - A, B, C, D, E
   - B, B, E, A, E
   - A, C, D, E
   - C, C, C, C, C
2. In the sample B, B, E, A, E, which original observations are absent?
3. Construct two different bootstrap samples of size 5.
4. Suppose three regression trees trained on three bootstrap samples predict

   $$7, \qquad 10, \qquad 8$$

   for the same new observation. Compute the bagged prediction.
5. Suppose five classification trees predict

   $$0, \quad 1, \quad 1, \quad 1, \quad 0.$$

   What class is predicted by majority vote?
6. Why might averaging or voting be more stable than relying on a single deep tree?

Concept question. Bagging does not primarily try to make every individual tree better. What property of the *combined* prediction is it trying to improve?

Exercise 8. From bagging to random forests

Suppose a classification problem has three input variables:

$$X_1 = \text{age}, \qquad X_2 = \text{income}, \qquad X_3 = \text{debt}.$$

Assume `income` is the strongest predictor and therefore tends to be selected near the root of many ordinary decision trees.

A random forest restricts the candidate variables considered at a split. For example:

Tree	variables allowed at a particular split
1	age, income
2	income, debt
3	age, debt
4	age, income
5	income, debt

1. Which tree above is forced to ignore income at this split?
2. Why might this force some trees to choose different splitting rules?
3. If all bagged trees always choose nearly the same first splits, are their prediction errors likely to be strongly correlated or weakly correlated?
4. Why can deliberately making individual trees more different improve the average prediction?
5. Complete the statements:
   - Bagging = bootstrap samples + _____ + averaging/voting.
   - Random forest = bagging + _____ at each split.
6. In one or two sentences, explain the conceptual difference between a single decision tree, bagging, and a random forest.

Final challenge. Which model is easiest to interpret directly: one tree, a bagged ensemble, or a random forest? Which would you expect to be most stable?