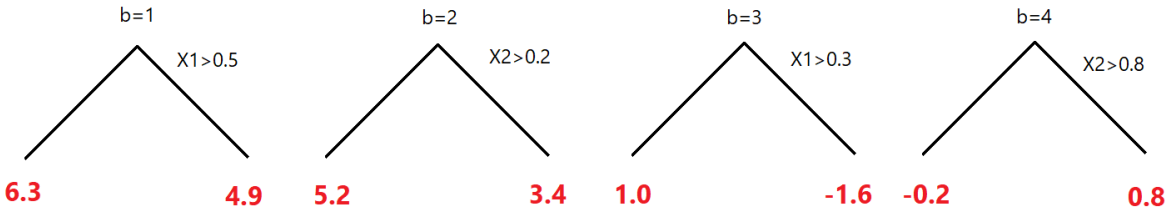
# STA 325: Homework 4

**DUE**: 11:59pm, November 16 (on Sakai)
**COVERAGE**: ISL Chapter 8

1. **[40 points]** Let's investigate some interesting properties of the boosting predictor.

   (a) **[4 points]** Suppose you run boosting on a dataset with two predictors, using an interaction depth of $d = 1$, i.e., only trees of one split (stump trees) are allowed. With a learning rate of $\lambda = 0.1$, the boosting procedure returns the following $B = 4$ decision trees $\{\hat{f}^b\}_{b=1}^4$:




   Draw out each tree as a partition of the prediction space $\mathcal{X} = \{(x_1, x_2) : x_1 \in [0, 1], x_2 \in [0, 1]\}$. Label clearly each split and the predicted value within each partition.

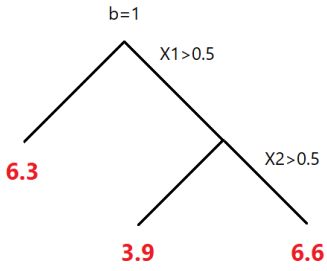
   (b) **[5 points]** Using part (a), draw out the boosted predictor $\hat{f}$ as a partition of the prediction space $\mathcal{X}$. Label clearly each split and the predicted value within each partition.

   (c) **[8 points]** With fixed $x_1 = 0.2$, draw out the boosted predictor function $\hat{f}$ as a function of only $x_2$. Do the same thing with fixed $x_1 = 0.4$. How are these two functions different? Similarly, draw the boosted predictor function $\hat{f}$ as a function of only $x_1$, with $x_2$ fixed at 0.3 and at 0.9. How are these two functions different?

   (d) **[5 points]** From your findings in part (c), does the boosted predictor $\hat{f}$ have any interaction effects? What is one property of $\hat{f}$ which allows for interpretability? Explain.

   (e) **[5 points]** Should more trees be added to the boosted predictor (i.e., should $B$ be increased)? Why? Use parts (a) and (b) to support your answer.

   (f) **[5 points]** Now suppose the first tree in part (a) is allowed to have $d = 2$ splits, yielding the following decision tree:




   Suppose the next three trees are the same as in part (a). Draw out the boosted predictor $\hat{f}$ as a partition of the prediction space $\mathcal{X}$. Label clearly each split and the predicted value within each partition.

(g) **[8 points]** Draw the boosted predictor function $\hat{f}$ as a function of $x_1$, with $x_2$ fixed at 0.3 and 0.9. How do these functions compare to those in part (c)? Does the new boosted predictor enjoy the property identified in part (d)? What does this mean for the interpretability of boosted trees with interaction depth $d = 2$? Explain.

2. **[18 points]** State whether each of the following statements are TRUE or FALSE. Briefly justify why in a couple of sentences.

   (a) In complexity cost pruning, the optimal tree with $\alpha = 0$ reduces to the unpruned tree from recursive binary splitting.

   (b) Out-of-bag error estimation provides an efficient way to estimate test error of a single unpruned decision tree.

   (c) The misclassification error rate of a classification tree always decreases in the tree-building process.

   (d) In bagging, the fitted trees are independent of each other, since the bootstrapped datasets are drawn with replacement from the training data.

   (e) In random forests, a smaller $m$ (the number of candidate predictors for splitting) results in lower bias but higher variance for the tree ensemble.

   (f) In boosting, a larger shrinkage parameter $\lambda$ typically requires a larger number of trees $B$ to reduce bias in the boosted ensemble.

3. **[22 points]** Let's dig deeper into the bootstrapping idea in bagging, where a "new" dataset is obtained by sampling the training data with replacement. Suppose your training data consists of the data points $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$. A *bootstrapped dataset* $\mathcal{D}^{*b} = \{(\mathbf{x}_i^*, y_i^*)\}_{i=1}^n$ consists of $n$ data points, each independently sampled with replacement from $\mathcal{D}$.

   (a) **[2 points]** What is the probability that the first bootstrapped data point $(\mathbf{x}_1^*, y_1^*)$ equals the first data point $(\mathbf{x}_1, y_1)$?

   (b) **[2 points]** What is the probability that the second bootstrapped data point $(\mathbf{x}_2^*, y_2^*)$ does *not* equal the first data point $(\mathbf{x}_1, y_1)$?

   (c) **[4 points]** Show that the probability of the first data point $(\mathbf{x}_1, y_1)$ *not* appearing in the bootstrapped dataset $\mathcal{D}^{*b}$ is $(1 - 1/n)^n$. What is the probability of the fifth data point $(\mathbf{x}_5, y_5)$ appearing?

   (d) **[5 points]** Let $p_j(n)$ be the probability of the $j$-th data point $(\mathbf{x}_j, y_j)$ appearing in the bootstrapped dataset $\mathcal{D}^{*b}$. What does $p_j(n)$ converge to as the number of data points grows large, i.e., as $n \to \infty$?

   (e) **[4 points]** Let's bring this back to bagging. Recall that each bagged tree is trained using a new bootstrapped dataset $\mathcal{D}^{*b}$, $b = 1, \cdots, B$. Suppose your training data is very large, i.e., $n \to \infty$. Using part (d), what % of the original data points are used (on average) to train a bagged tree? What % of the original data are *not* used (on average)?

   (f) **[5 points]** From part (e), it is clear that a bagged tree will not make use of all $n$ data points on average. Are the unused points wasted? If so, explain why. If not, explain how these points may be useful.