Bagging

STAT 37710 / CMSC 35300
Rebecca Willett and Yuxin Chen

Recall Decision Trees:

$$\hat{h}_{tree}(x) = \sum_{j=1}^{m} c_j \cdot \mathbb{1}\{x \in R_j\}$$

$$= c_j \quad \text{such that } x \in R_j$$

where c_j is the label for region R_j

In binary classification, $c_j \in \{0,1\}$ $\forall j$.

But for each R_j, we can also compute

$$\hat{p}(R_j) = \underbrace{\frac{1}{n_j} \sum_{x_i \in R_j} \mathbb{1}\{y_i = 1\}}$$

fraction of training samples in R_j with label = 1

(Then $c_j = \begin{cases} 1 & \hat{p}(R_j) > 1/2 \\ 0 & \text{otherwise} \end{cases}$

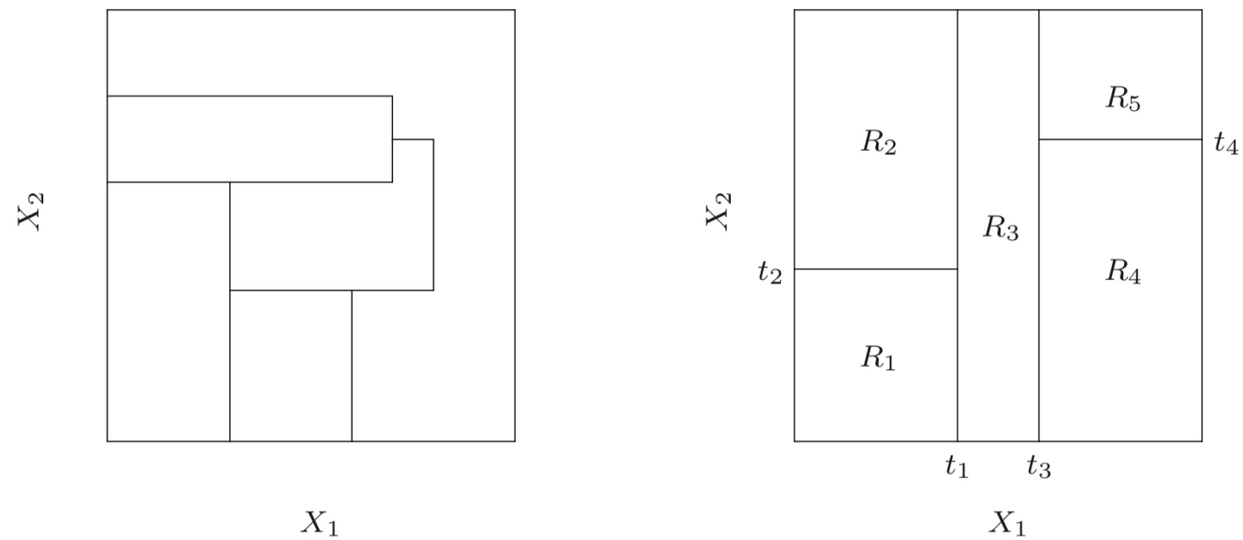


FIGURE 9.2. *Partitions and CART. Top right panel shows a partition of a two-dimensional feature space by recursive binary splitting, as used in CART, applied to some fake data. Top left panel shows a general partition that cannot be obtained from recursive binary splitting. Bottom left panel shows the tree corresponding to the partition in the top right panel, and a perspective plot of the prediction surface appears in the bottom right panel.*

Bagging = Bootstrap Aggregating learns a predictor by aggregating the predictors learned over multiple random draws (bootstrap samples) from the training data

We start with training samples $Z = \{(x_i, y_i),\ i = 1, \ldots, n\}$

A bootstrap sample of size m from Z is $\{(\tilde{x}_i, \tilde{y}_i),\ i = 1, \ldots, m\}$, where each $(\tilde{x}_i, \tilde{y}_i)$ is drawn uniformly at random from Z (with replacement).

Example: $Z = \{(1,1), (2,2), (3,3), (4,4), (5,5)\}$, $m = 4$

1st bootstrap sample of size m = $\{(5,5), (3,3), (5,5), (1,1)\}$

2nd bootstrap sample of size m = $\{(3,3), (5,5), (4,4), (3,3)\}$

3rd bootstrap sample of size m = $\{(4,4), (4,4), (2,2), (2,2)\}$

Imagine we have a model we can fit to the training data (for example, a decision tree or logistic regression) to produce a predictor $\hat{h}(x)$ that we use to predict $\mathbb{E}[Y|X = x]$.

With bagging we compute B different bootstrap samples, $\tilde{Z}_1, \tilde{Z}_2, \ldots, \tilde{Z}_B$, learn a predictor for each one ($\hat{h}_b$ for $b = 1, 2, \ldots, B$), and then aggregate the $\hat{h}_b$'s to form $\hat{h}$.

Example: binary classification, where $y_i \in \{0, 1\}$

case 1: $\hat{h}_b(x)$ outputs label 0 or 1

$$\hat{h}(x) = \begin{cases} 0 & \text{if } \sum_{b=1}^{B} \mathbb{1}\{\hat{h}_b(x)=0\} > \sum_{b=1}^{B} \mathbb{1}\{\hat{h}_b(x)=1\} \\ 1 & \text{otherwise} \end{cases}$$

} majority vote / consensus

case 2: $\hat{p}_b(x)$ is an estimate of $\mathbb{E}[Y|X=x] = \Pr[Y=1|X=x]$ for classifier b

$$\hat{p}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{p}_b(x)$$

from here we can define the predicted class label $\hat{h}(x) = \begin{cases} 1 & \text{if } \hat{p}(x) \geq 1/2 \\ 0 & \text{if } \hat{p}(x) < 1/2 \end{cases}$

} probability aggregation

Bagging is commonly used when the $\hat{h}_b$'s correspond to decision trees — the resulting method is called a RANDOM FOREST

Example from Elements of Statistical Learning.

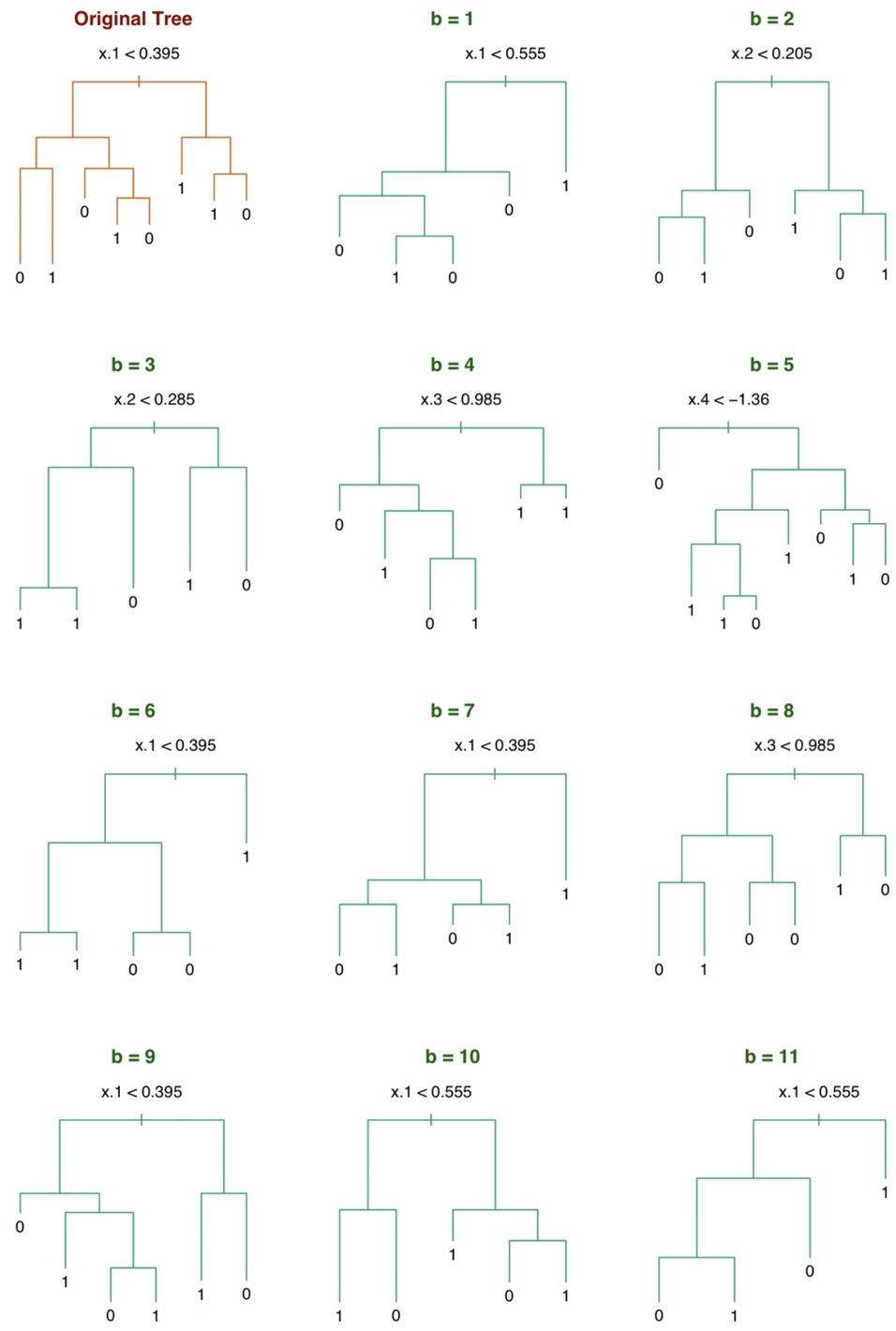


FIGURE 8.9. *Bagging trees on simulated dataset. The top left panel shows the original tree. Eleven trees grown on bootstrap samples are shown. For each tree, the top split is annotated.*

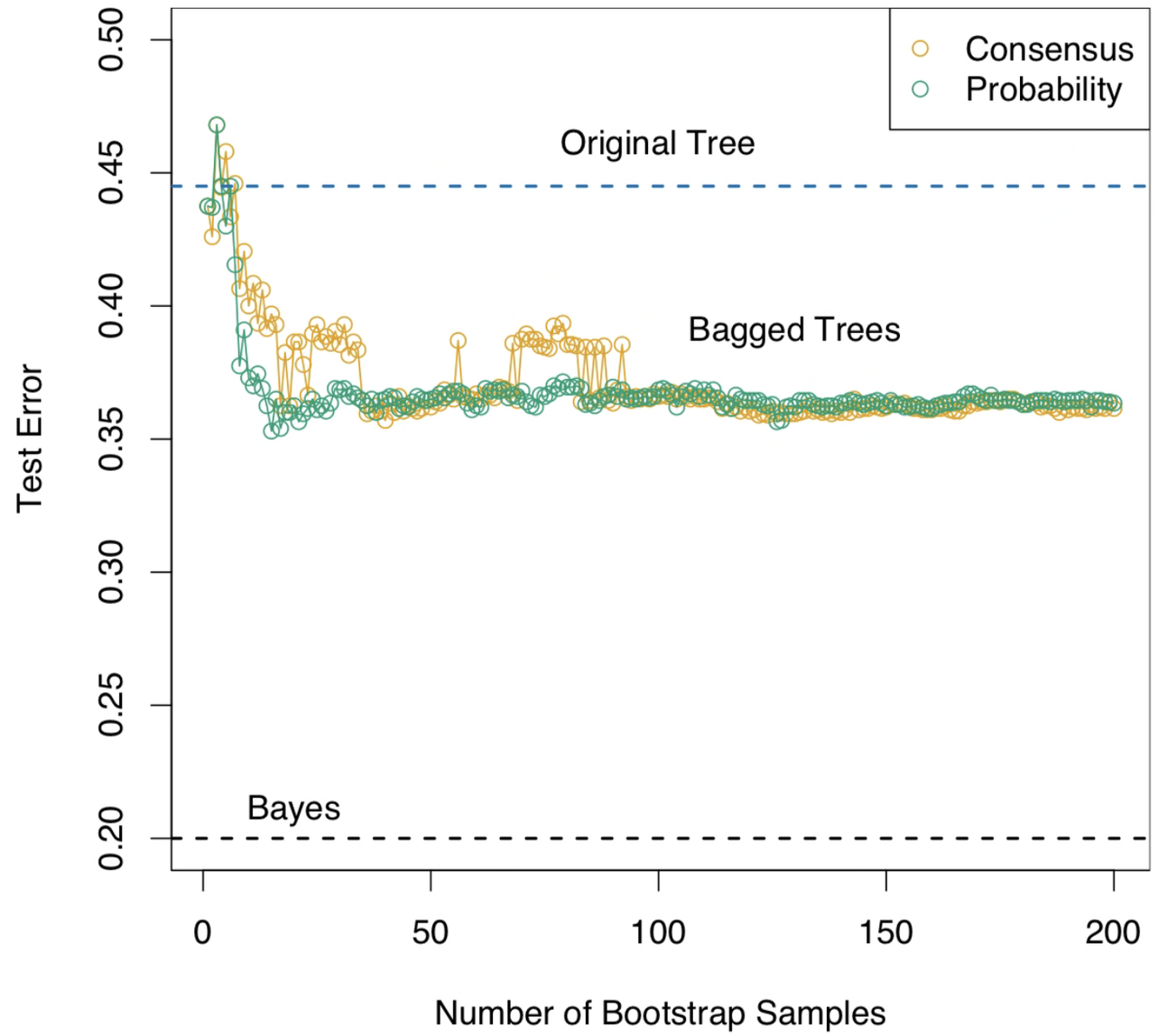


FIGURE 8.10. *Error curves for the bagging example of Figure 8.9. Shown is the test error of the original tree and bagged trees as a function of the number of bootstrap samples. The orange points correspond to the consensus vote, while the green points average the probabilities.*

Why does bagging work?

Imagine that instead of having B bootstrap samples, we instead have B independent training datasets. each resulting in a classifier $\hat{h}_b$, $b = 1, 2, \ldots, B$. Assume that

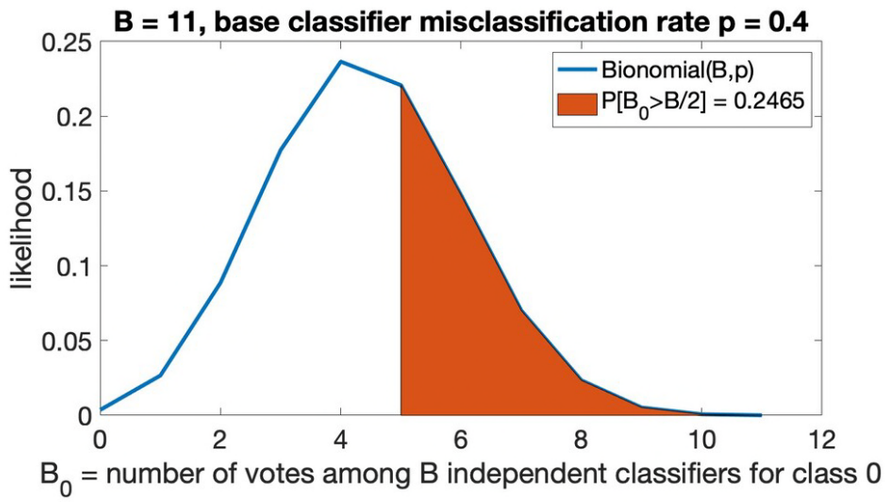
misclassification rate $\mathbb{P}(\hat{h}_b(x) \neq y) = 0.4$ for all b

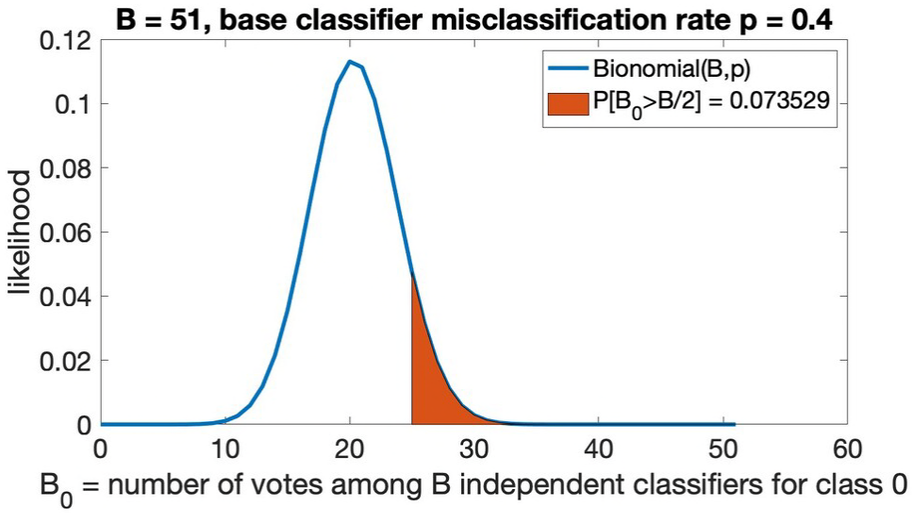
Now with bagging, $\hat{h}(x) = \begin{cases} 0 & \text{if } \sum_{b=1}^{B} \mathbb{1}_{\{\hat{h}_b(x)=0\}} > \sum_{b=1}^{B} \mathbb{1}_{\{\hat{h}_b(x)=1\}} \\ 1 & \text{otherwise} \end{cases}$

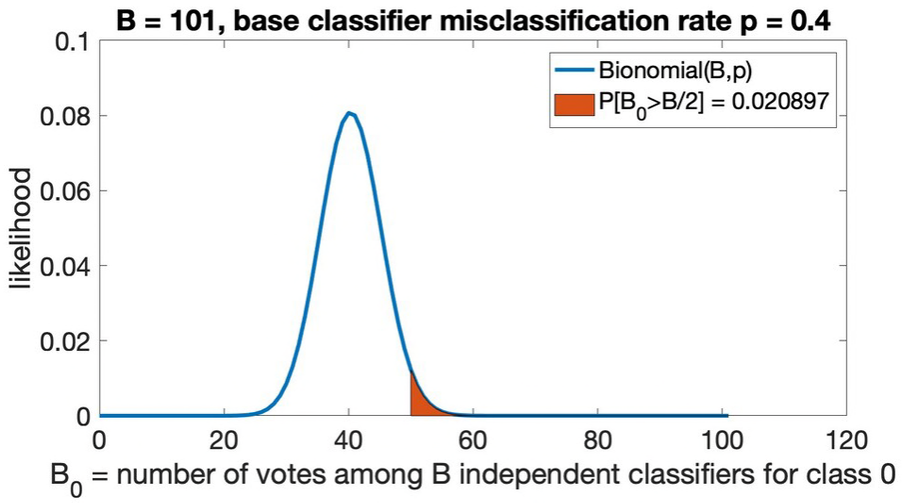
Let $B_0 = \{\#\text{ votes for label } 0\} = \sum_{b=1}^{B} \mathbb{1}_{\{\hat{h}_b(x)=0\}}$; similarly $B_1 = B - B_0$

Then $\hat{h}(x) = \underset{k \in \{0,1\}}{\text{argmax}}\ B_k = \mathbb{1}_{\{B_1 > B_0\}} = \mathbb{1}_{\{B_0 < B/2\}}$

Assume the correct label for point x is 1.
Note $B_0 \sim \text{Binom}(B, 0.4) \Rightarrow$ misclassification rate $\mathbb{P}(\hat{h}(x) = 0) = \mathbb{P}(B_0 > B/2)$







as $B \to \infty$, $\mathbb{P}(B_0 > B/2) \to 0$

but in example, error did not $\to 0$. Why?

Because we don't have independent training samples, but rather bootstrap samples

Similar arguments in setting where $\mathbb{P}(\hat{h}_0(x) \neq y) > 1/2$ show bagging is very bad with poor initial classifiers.

	PROS	CONS
DECISION TREES	• interpretable • model-free • computationally efficient.	high variance — small perturbation in data can yield very different splits ⇒ very different classification rule.
RANDOM FORESTS	Less variance good predictive accuracy	not interpretable computationally complex

Additional Reading

- https://www.stat.berkeley.edu/~breiman/bagging.pdf
- Elements of Statistical Learning, 8.7, 9.2
- Intro to Statistical Learning, Chapter 8