

# Bagging and Random Forests

David Rosenberg

New York University

March 28, 2017

# The Benefits of Averaging

# A Lousy Estimator

- Let  $Z, Z_1, \dots, Z_n$  i.i.d.  $\mathbb{E}Z = \mu$  and  $\text{Var}Z = \sigma^2$ .
- We could use any single  $Z_i$  to estimate  $\mu$ .
- Performance?
- Unbiased:  $\mathbb{E}Z_i = \mu$ .
- Standard error of estimator would be  $\sigma$ .
  - The **standard error** is the standard deviation of the sampling distribution of a statistic.
  - $\text{SD}(Z) = \sqrt{\text{Var}(Z)} = \sqrt{\sigma^2} = \sigma$ .

## Variance of a Mean

- Let  $Z, Z_1, \dots, Z_n$  i.i.d.  $\mathbb{E}Z = \mu$  and  $\text{Var}Z = \sigma^2$ .
- Let's consider the average of the  $Z_i$ 's.
  - Average has the same expected value but smaller standard error:

$$\mathbb{E} \left[ \frac{1}{n} \sum_{i=1}^n Z_i \right] = \mu \quad \text{Var} \left[ \frac{1}{n} \sum_{i=1}^n Z_i \right] = \frac{\sigma^2}{n}.$$

- Clearly the average is preferred to a single  $Z_i$  as estimator.
- Can we apply this to reduce variance of general decision functions?

# Averaging Independent Prediction Functions

- Suppose we have  $B$  independent training sets from the same distribution.
- Learning algorithm gives  $B$  decision functions:  $\hat{f}_1(x), \hat{f}_2(x), \dots, \hat{f}_B(x)$
- Define the average prediction function as:

$$\hat{f}_{\text{avg}} = \frac{1}{B} \sum_{b=1}^B \hat{f}_b$$

- What's random here?

# Averaging Independent Prediction Functions

- Fix some  $x \in \mathcal{X}$ .
- Then average prediction on  $x$  is

$$\hat{f}_{\text{avg}}(x) = \frac{1}{B} \sum_{b=1}^B \hat{f}_b(x).$$

- Consider  $\hat{f}_{\text{avg}}(x)$  and  $\hat{f}_1(x), \dots, \hat{f}_B(x)$  as random variables (since training data random).
- $\hat{f}_1(x), \dots, \hat{f}_B(x)$  are i.i.d.
- $\hat{f}_{\text{avg}}(x)$  and  $\hat{f}_b(x)$  have the same expected value, but
- $\hat{f}_{\text{avg}}(x)$  has smaller variance:

$$\begin{aligned} \text{Var}(\hat{f}_{\text{avg}}(x)) &= \frac{1}{B^2} \text{Var} \left( \sum_{b=1}^B \hat{f}_b(x) \right) \\ &= \frac{1}{B} \text{Var}(\hat{f}_1(x)) \end{aligned}$$

# Averaging Independent Prediction Functions

- Using

$$\hat{f}_{\text{avg}} = \frac{1}{B} \sum_{b=1}^B \hat{f}_b$$

seems like a win.

- But in practice we don't have  $B$  independent training sets...
- Instead, we can use **the bootstrap**....

## Review: Bootstrap



# The Bootstrap Sample

## Definition

A **bootstrap sample** from  $\mathcal{D}_n$  is a sample of size  $n$  drawn *with replacement* from  $\mathcal{D}_n$ .

- In a bootstrap sample, some elements of  $\mathcal{D}_n$ 
  - will show up multiple times,
  - some won't show up at all.
- Each  $X_i$  has a probability  $(1 - 1/n)^n$  of not being selected.
- Recall from analysis that for large  $n$ ,

$$\left(1 - \frac{1}{n}\right)^n \approx \frac{1}{e} \approx .368.$$

- So we expect  $\sim 63.2\%$  of elements of  $\mathcal{D}$  will show up at least once.

# The Bootstrap Method

## Definition

A **bootstrap method** is when you *simulate* having  $B$  independent samples from  $P$  by taking  $B$  bootstrap samples from the sample  $\mathcal{D}_n$ .

- Given original data  $\mathcal{D}_n$ , compute  $B$  bootstrap samples  $D_n^1, \dots, D_n^B$ .
- For each bootstrap sample, compute some function

$$\phi(D_n^1), \dots, \phi(D_n^B)$$

- Work with these values as though  $D_n^1, \dots, D_n^B$  were i.i.d.  $P$ .
- **Amazing fact:** Things often come out very close to what we'd get with independent samples from  $P$ .

# Bagging

# Bagging

- Draw  $B$  bootstrap samples  $D^1, \dots, D^B$  from original data  $\mathcal{D}$ .
- Let  $\hat{f}_1, \hat{f}_2, \dots, \hat{f}_B$  be the decision functions for each set.
- The **bagged decision function** is a **combination** of these:

$$\hat{f}_{\text{avg}}(x) = \text{Combine} \left( \hat{f}_1(x), \hat{f}_2(x), \dots, \hat{f}_B(x) \right)$$

- How might we combine
  - decision functions for regression?
  - binary class predictions?
  - binary probability predictions?
  - multiclass predictions?
- Bagging proposed by Leo Breiman (1996).

# Bagging for Regression

- Draw  $B$  bootstrap samples  $D^1, \dots, D^B$  from original data  $\mathcal{D}$ .
- Let  $\hat{f}_1, \hat{f}_2, \dots, \hat{f}_B : \mathcal{X} \rightarrow \mathbf{R}$  be the predictions functions for each set.
- Bagged prediction function is given as

$$\hat{f}_{\text{bag}}(x) = \frac{1}{B} \sum_{b=1}^B \hat{f}_b(x).$$

- Empirically,  $\hat{f}_{\text{bag}}$  often performs similarly to what we'd get from training on  $B$  independent samples:
  - $\hat{f}_{\text{bag}}(x)$  has same expectation as  $\hat{f}_1(x)$ , but
  - $\hat{f}_{\text{bag}}(x)$  has smaller variance than  $\hat{f}_1(x)$

## Out-of-Bag Error Estimation

- Each bagged predictor is trained on about 63% of the data.
- Remaining 37% are called **out-of-bag (OOB)** observations.
- For  $i$ th training point, let

$$S_i = \{b \mid D^b \text{ does not contain } i\text{th point}\}.$$

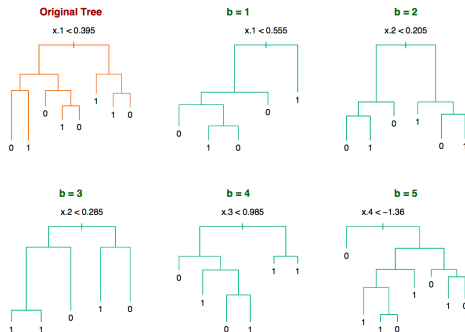
- The **OOB prediction** on  $x_i$  is

$$\hat{f}_{\text{OOB}}(x_i) = \frac{1}{|S_i|} \sum_{b \in S_i} \hat{f}_b(x).$$

- The OOB error is a good estimate of the test error.
- OOB error is similar to cross validation error – both are computed on training set.

# Bagging Classification Trees

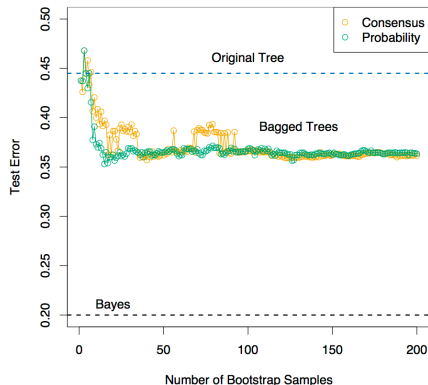
- Input space  $\mathcal{X} = \mathbf{R}^5$  and output space  $\mathcal{Y} = \{-1, 1\}$ .
- Sample size  $N = 30$  (simulated data)



From ESL Figure 8.9

# Comparing Classification Combination Methods

- Two ways to combine classifications: consensus class or average probabilities.



From ESL Figure 8.10



## Terms “Bias” and “Variance” in Casual Usage (Warning! Confusion Zone!)

- Restricting the hypothesis space  $\mathcal{F}$  “**biases**” the fit
  - **towards** a simpler model and
  - **away** from the best possible fit of the training data.
- Full, unpruned decision trees have very little bias.
- Pruning decision trees introduces a bias.
- **Variance** describes how much the fit changes across different random training sets.
- If different random training sets give very similar fits, then algorithm has high **stability**.
- Decision trees are found to be high variance (i.e. not very stable).

## Conventional Wisdom on When Bagging Helps

- Hope is that bagging reduces variance without making bias worse.
- General sentiment is that bagging helps most when
  - Relatively unbiased base prediction functions
  - High variance / low stability
    - i.e. small changes in training set can cause large changes in predictions
- Hard to find clear and convincing theoretical results on this
- But following this intuition leads to improved ML methods, e.g. Random Forests

# Random Forests

## Recall the Motivating Principal of Bagging

- Averaging  $\hat{f}_1, \dots, \hat{f}_B$  reduces variance, if they're based on i.i.d. samples from  $P_{\mathcal{X} \times \mathcal{Y}}$
- Bootstrap samples are
  - independent samples from the training set, but
  - are **not** independent samples from  $P_{\mathcal{X} \times \mathcal{Y}}$ .
- This dependence limits the amount of variance reduction we can get.
- Would be nice to reduce the dependence between  $\hat{f}_i$ 's...

# Variance of a Mean of Correlated Variables

- For  $Z, Z_1, \dots, Z_n$  i.i.d. with  $\mathbb{E}Z = \mu$  and  $\text{Var}Z = \sigma^2$ ,

$$\mathbb{E} \left[ \frac{1}{n} \sum_{i=1}^n Z_i \right] = \mu \quad \text{Var} \left[ \frac{1}{n} \sum_{i=1}^n Z_i \right] = \frac{\sigma^2}{n}.$$

- What if  $Z$ 's are correlated?
- Suppose  $\forall i \neq j, \text{Corr}(Z_i, Z_j) = \rho$ . Then

$$\text{Var} \left[ \frac{1}{n} \sum_{i=1}^n Z_i \right] = \rho \sigma^2 + \frac{1-\rho}{n} \sigma^2.$$

- For large  $n$ , the  $\rho \sigma^2$  term dominates – limits benefit of averaging.

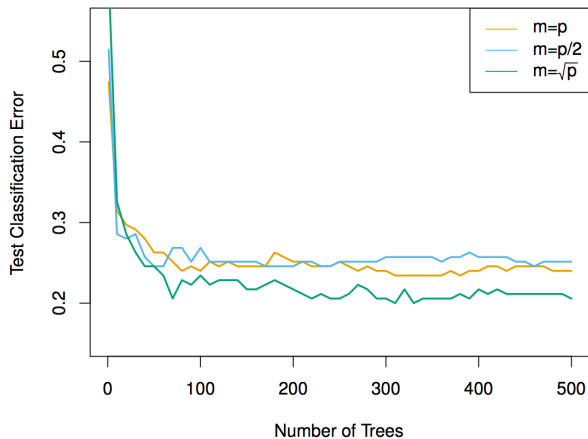
# Random Forest

## Main idea of random forests

Use **bagged decision trees**, but modify the tree-growing procedure to reduce the correlation between trees.

- **Key step** in random forests:
  - When constructing **each tree node**, restrict choice of splitting variable to a randomly chosen subset of features of size  $m$ .
- Typically choose  $m \approx \sqrt{p}$ , where  $p$  is the number of features.
- Can choose  $m$  using cross validation.

# Random Forest: Effect of $m$ size



From *An Introduction to Statistical Learning, with applications in R* (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani.