Bagging and Random Forests

David S. Rosenberg

New York University

April 10, 2018
Ensemble Methods: Introduction
Ensembles: Parallel vs Sequential

- Ensemble methods combine multiple models

- **Parallel ensembles**: each model is built independently
  - e.g. bagging and random forests
  - Main Idea: Combine many (high complexity, low bias) models to reduce variance

- **Sequential ensembles**:
  - Models are generated sequentially
  - Try to add new models that do well where previous models lack
The Benefits of Averaging
Let \( Z, Z_1, \ldots, Z_n \) 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 $Z, Z_1, \ldots, Z_n$ 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 prediction functions?
Averaging Independent Prediction Functions

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

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

- What’s random here?
- The $B$ independent training sets are random, which gives rise to variation among the $\hat{f}_b$’s.
Averaging Independent Prediction Functions

- Fix some particular $x_0 \in \mathcal{X}$.
- Then average prediction on $x_0$ is
  \[
  \hat{f}_{\text{avg}}(x_0) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}_b(x_0).
  \]

- Consider $\hat{f}_{\text{avg}}(x_0)$ and $\hat{f}_1(x_0), \ldots, \hat{f}_B(x_0)$ as random variables.
  - Since the training sets were random
  - We have no idea about the distributions of $\hat{f}_1(x_0), \ldots, \hat{f}_B(x_0)$ – they could be crazy...
  - But we do know that $\hat{f}_1(x_0), \ldots, \hat{f}_B(x_0)$ are i.i.d. And that’s all we need here...
Averaging Independent Prediction Functions

- The average prediction on $x_0$ is

$$\hat{f}_{\text{avg}}(x_0) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}_b(x_0).$$

- $\hat{f}_{\text{avg}}(x_0)$ and $\hat{f}_b(x_0)$ have the same expected value, but

- $\hat{f}_{\text{avg}}(x_0)$ has smaller variance:

$$\text{Var}(\hat{f}_{\text{avg}}(x_0)) = \frac{1}{B^2} \text{Var} \left( \sum_{b=1}^{B} \hat{f}_b(x_0) \right) = \frac{1}{B} \text{Var} \left( \hat{f}_1(x_0) \right)$$
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 $D_n$.

- Given original data $D_n$, compute $B$ bootstrap samples $D^1_n, \ldots, D^B_n$.
- For each bootstrap sample, compute some function
  \[
  \phi(D^1_n), \ldots, \phi(D^B_n)
  \]
- Work with these values as though $D^1_n, \ldots, D^B_n$ were i.i.d. $P$.
- **Amazing fact:** Things often come out very close to what we’d get with independent samples from $P$. 

David S. Rosenberg  (New York University)  
DS-GA 1003 / CSCI-GA 2567  
April 10, 2018 14 / 29
Bagging
Bagging

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

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

- How might we combine
  - prediction functions for regression?
  - binary class predictions?
  - binary probability predictions?
  - multiclass predictions?

- Bagging proposed by Leo Breiman (1996).
Bagging for Regression

- Draw $B$ bootstrap samples $D^1, \ldots, D^B$ from original data $\mathcal{D}$.
- Let $\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_B : \mathcal{X} \to \mathbb{R}$ 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 
  \[ \hat{f}_{\text{OOB}}(x_i) = \frac{1}{|S_i|} \sum_{b \in S_i} \hat{f}_b(x_i). \]
- 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.
Input space $\mathcal{X} = \mathbb{R}^5$ and output space $\mathcal{Y} = \{-1, 1\}$.

- Sample size $n = 30$
- Each bootstrap tree is quite different
- Different splitting variable at the root
- This high degree of variability from small perturbations of the training data is why tree methods are described as **high variance**.

From HTF Figure 8.9
Comparing Classification Combination Methods

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

From HTF Figure 8.10

David S. Rosenberg  (New York University)  DS-GA 1003 / CSCI-GA 2567  April 10, 2018  20 / 29
Restricting the hypothesis space $\mathcal{F}$ “biases” the fit
- away from the best possible fit of the training data, and
- towards a [usually] simpler model.

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

Use bagged decision trees, but modify the tree-growing procedure to reduce the dependence 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

- Usual approach is to build very deep trees (low bias)
- Diversity in individual tree prediction functions from
  - bootstrap samples (somewhat different training data) and
  - randomized tree building
- Bagging seems to work better when we are combining a diverse set of prediction functions.
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.
Appendix
Variance of a Mean of Correlated Variables

For $Z, Z_1, \ldots, Z_n$ i.i.d. with $E[Z] = \mu$ and $\text{Var}[Z] = \sigma^2$,

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

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.