CSCE 633: Machine Learning

Lecture 19: Boosting

Texas A&M University

10-7-19

Last Time

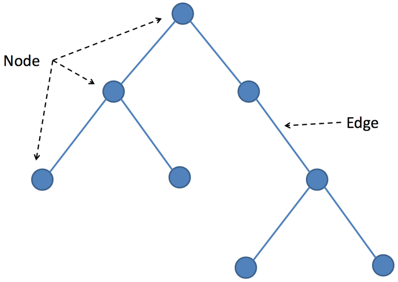
- Decision Trees
- Random Forest

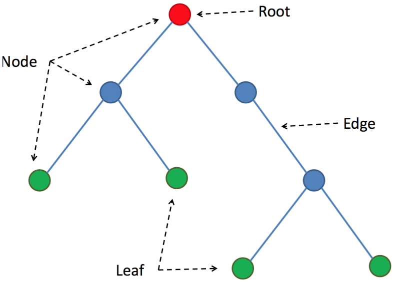
Goals of this lecture

- Reminder: Exam 1 - Monday, October 14 in class
- CLOSED BOOK, CLOSED NOTES - Starts right at 1:50. DO NOT BE LATE! YOU WILL NOT GET EXTRA TIME!
- Boosting

What is a decision tree

A hierarchical data structure implementing the divide-and-conquer strategy for decision making





Can be used for both classification & regression

Gini Index and Entropy

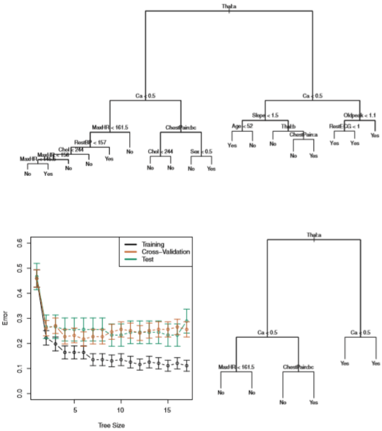
$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

, which measures the total variance across K classes. This is a measure of node purity.

$$H = -\sum_{k=1}^{K} \hat{p}_{mk} \log(\hat{p}_{mk})$$

, Entropy which takes a value near 0 if all the $\hat{p}$ are near zero or one - smaller value if node is pure

Classification and Pruning



Bagging

- Take B different bootstraps of our one dataset
- $\hat{f}_{bag}(x) = \frac{1}{B} \sum_{b=1}^{*B} \hat{f}^{b}(x)$
- Turns out, you can grow these trees without pruning
- Regression - average the values from each tree
- Classification - majority vote from each tree
- Test error can be plotted as a function of B
- B is not a critical parameter (will see shortly) so large B does not mean we overfit

Out of Bag Error

- If we repeatedly fit bootstrapped subsets (say 2/3 of data)
- Each time we are left with 1/3 of the data we can call out of bag
- We can estimate error for this - called Out of Bag Estimation

Random Forest

- set $m \approx \sqrt{p}$

Random Forest

- set $m \approx \sqrt{p}$
- Each time $\frac{p-m}{p}$ predictors aren't even considered

Random Forest

- set $m \approx \sqrt{p}$
- Each time $\frac{p-m}{p}$ predictors aren't even considered
- other predictors have a chance

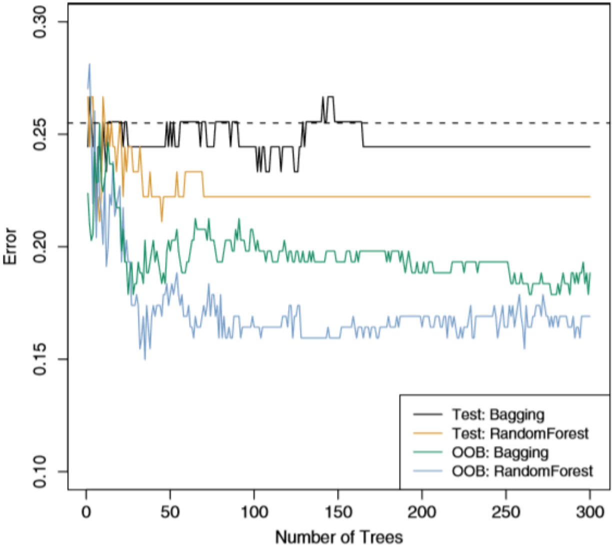
Random Forest

- set $m \approx \sqrt{p}$
- Each time $\frac{p-m}{p}$ predictors aren't even considered
- other predictors have a chance
- Turns out, this process decorrelates trees

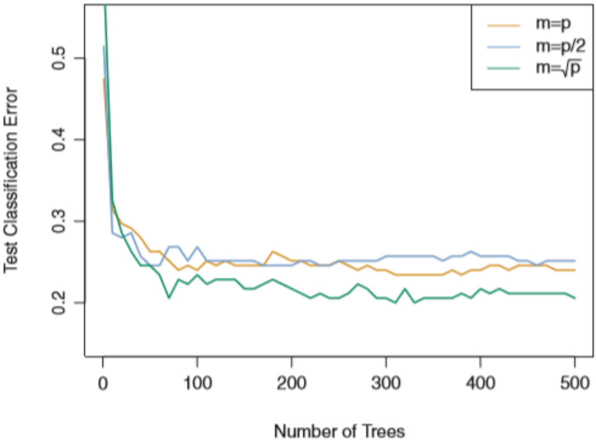
Random Forest

- set $m \approx \sqrt{p}$
- Each time $\frac{p-m}{p}$ predictors aren't even considered
- other predictors have a chance
- Turns out, this process decorrelates trees
- The average tree becomes less variable and thus more reliable

Example: Heart Dataset



RF with different m



Boosting

- Can improve prediction of decision trees even further
- develop a method that can work on any classifier - applied here to regression trees
- Bagging - build each tree randomly
- Random Forest - build each tree randomly, with random variations in predictors allowed
- What if we built trees sequentially?

Boosting

Algorithm 8.2 *Boosting for Regression Trees*

1. Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all i in the training set.
2. For $b = 1, 2, \ldots, B$, repeat:
   (a) Fit a tree $\hat{f}^b$ with d splits ($d+1$ terminal nodes) to the training data (X, r).
   (b) Update $\hat{f}$ by adding in a shrunken version of the new tree:
   $$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x). \tag{8.10}$$
   (c) Update the residuals,
   $$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i). \tag{8.11}$$
3. Output the boosted model,
$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x). \tag{8.12}$$

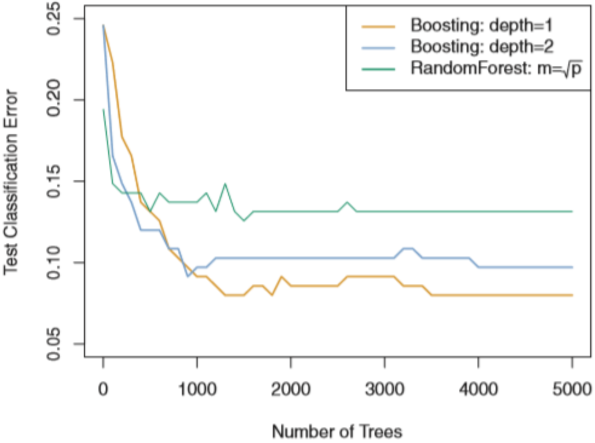
Boosting

- Learn slowly on shallow trees
- Given a current model (number of trees) - calculate residuals
- Build next tree to improve, iteratively, on the residuals
- Slowly improve $\hat{f}$ where it does not perform well!
- Boosted classification is a bit trickier - next time

Boosting: Hyperparameters

- Number of trees - if B is too large, this model DOES overfit
- λ is small, but greater than 0. Typically $0.001 < \lambda < 0.01$
- Depth d of trees is often small, often $d = 1$ decision stumps (very interpretable)
- Boosts a bunch of weak classifiers into a strong classifier.

Boosting



Boosting: depth=1
Boosting: depth=2
RandomForest: m=√p
Test Classification Error
0.05
0.10
0.15
0.20
0.25
0
1000
2000
3000
4000
5000
Number of Trees

Boosting:Formulation

$$f(x) = \beta_0 + \sum_{b=1}^{B} \beta_m \phi_m(x)$$

Or - in other notation

$$f(x) = w_0 + \sum_{m=1}^{M} w_m \phi_m(x)$$

Takeaways and Next Time

- Boosting
- Next Time: More Boosting
- Reminder: Exam 1 - Monday, October 14