Bagging (Bootstrap Aggregating)

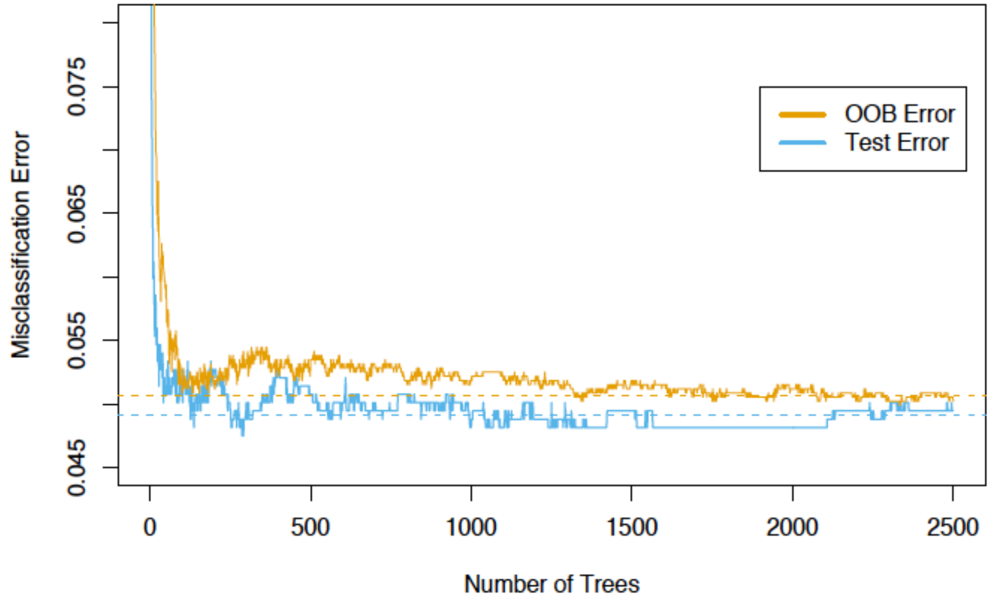
- Idea: Generate multiple trees from different training sets, and apply all models to each test sample and take average (or majority) of the results from all the trees
- How to generate different training sets giving a dataset?
- Cross validation: using a subset of data each time for training and the remaining for testing
- Bootstrap sampling: Sampling by replacement, each sampling contains the same number of samples as the original dataset, but some samples are replicated, others were not included
- Bagging: Generate B models from B bootstrap samplings
  - Regression: Average the prediction results from B models
  - Classification: Take the majority class index
- Apply to other regressors/classifiers as well.

Out of bag (OOB) error

- Each time we draw a bootstrap sampling, we only use ~63% of the samples
  - Probability that a sample is chosen among N samples in each bootstrap sampling

$$1-\left(1-\frac{1}{N}\right)^N \sim 1-e^{-1}=0.632$$

- We can use the remaining samples for testing
- OOB Error
  - For each sample x_n, find the models generated by samplings which do not contain x_n. There are about 0.37B of models. Average predictions by these models for x_n.
  - Compute the regression/classification error for x_n
  - Average the error over all samples
- We can use OOB error as an estimate of the test error.
- Does not require design multiple models for multiple folds as in cross validation. OOB can be estimated from one pass of designing multiple trees.



From ESL Fig. 15.4

Why bagging?

- When a regressor or classifier has tendency to overfit (i.e. sensitive to the training set), bagging reduces the variance of the prediction
  - Reduce the test error
  - Particularly useful for decision trees
- When the sample number N in a given dataset is large
  - The empirical distribution is similar to the true distribution
  - Each bootstrap sampling is similar to an independent realization of the true distribution
  - Bagging amounts to averaging the fits from many identically distributed datasets

Problems with bagging?

- Trees generated by different samplings can be very similar
- Test error reduces slowly as B increases

  - $f_b(x)$: prediction by tree b for test sample x
  - Assume $f_b(x)$ for all b have the same mean μ and variance σ^2
  - Assume these predictions have pair-wise correlation ρ
  - The variance of the average prediction $f(x) = \frac{1}{B}\sum_b f_b(x)$: (Shown on board)

$$\sigma_B^2 = \rho\,\sigma^2 + \frac{1}{B}(1-\rho)\sigma^2$$

Random Forest

- As with Bagging: fit a different tree for each bootstrap sampling
- Recall that when growing a tree, at each current node (region), we split the region by choosing a particular feature and a threshold. The feature and the threshold are chosen among all P features to minimize a certain loss.
- With random forest, randomly choose among a subset of features (P’<P) for splitting each node
- The resulting trees are more different
- Rule of thumb: $P' = \sqrt{P}$ (but should be turned using test error or OOB error)

Bagging vs. RF

- Bagging:

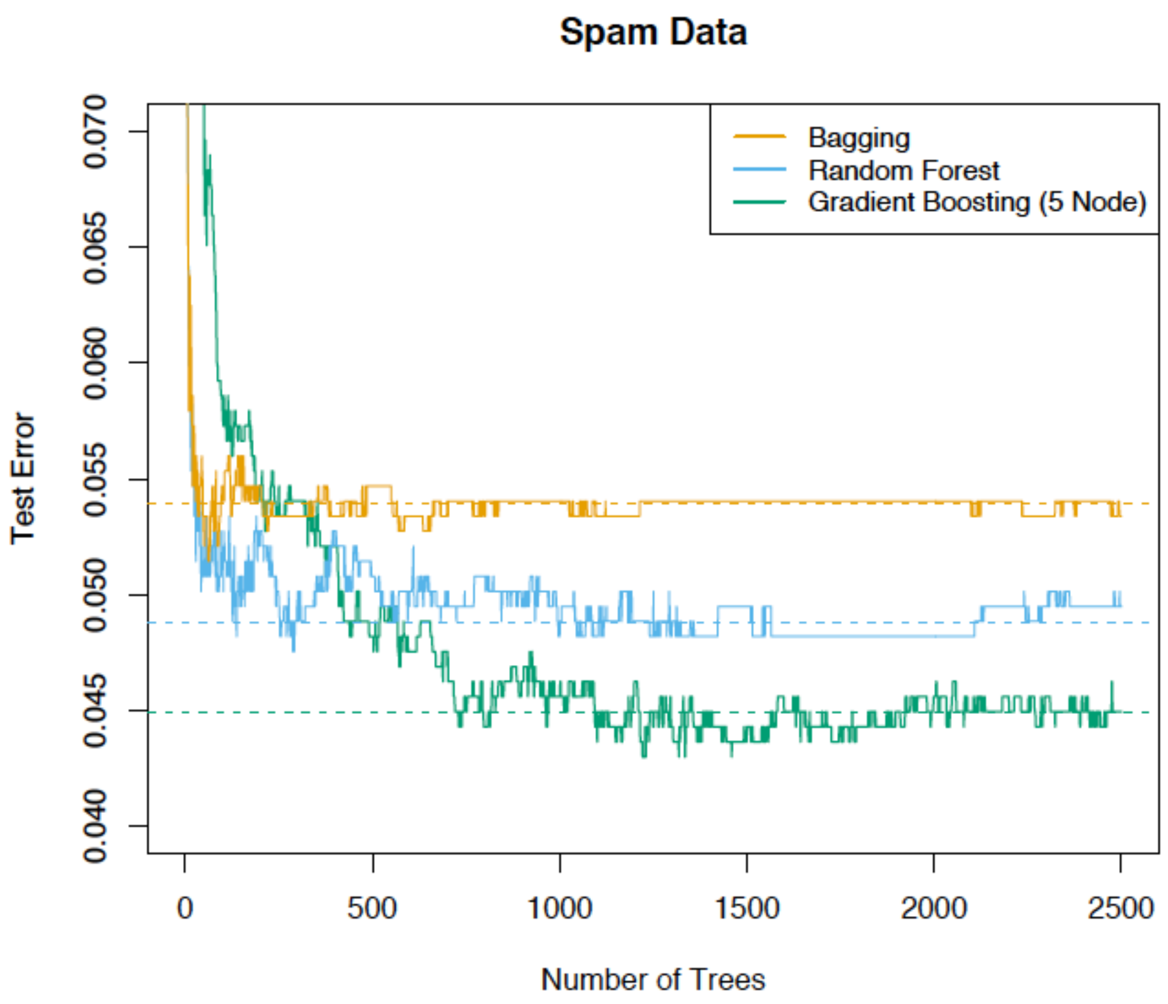
$$\sigma_B^2 = \rho\,\sigma^2 + \frac{1}{B}(1-\rho)\sigma^2$$

- Random forest (assuming $\rho = 0$):

$$\sigma_B^2 = \frac{1}{B}\sigma^2$$

- Recall:

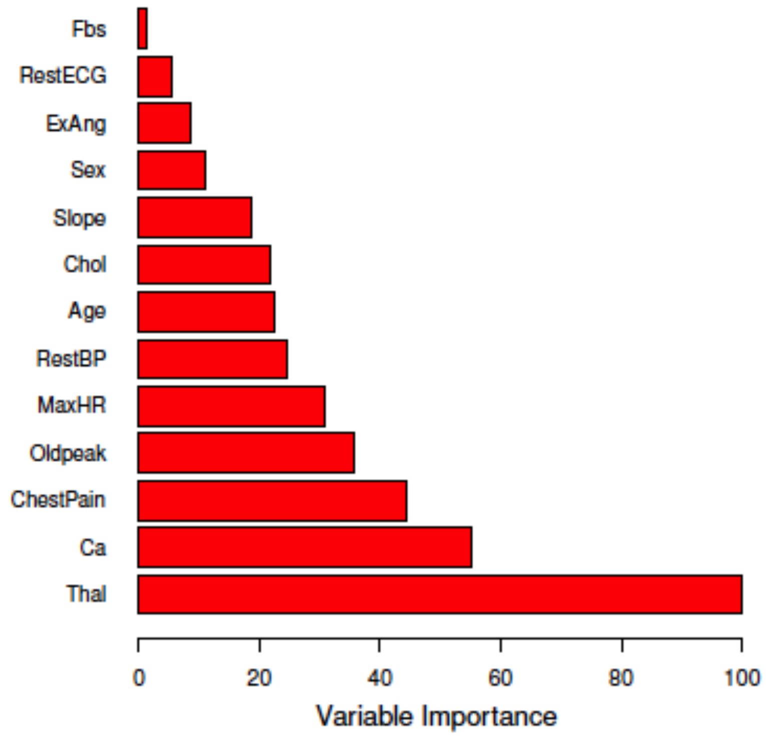
Test error = bias^2+ Variance +Noise Variance



From ESL, Fig. 15.1

Feature importance

- For each feature, add up the loss reduction at splits where this feature was used over all trees.



Demo: Random forest