Statistical Machine Learning 11 — Bagging October 6, 2015

Lecturer: Prof. Homrighausen Scribe: Veronica Burt

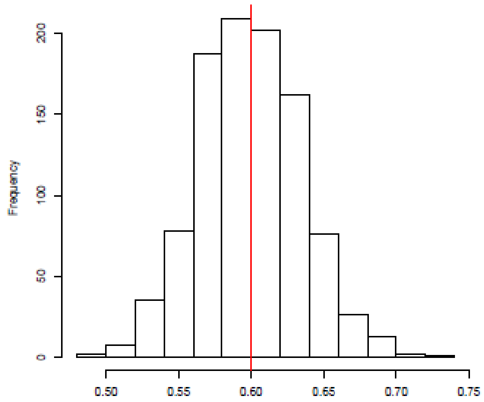
1 Previous Lectures

We have been discussing

- CART: Classification and Regression Trees
- Finding regression trees using squared error loss, which are piecewise (and constant over those regions)
- Specifying loss functions to determine "good splits" to make in trees, specifically cross-entropy and the Gini index

2 Bagging

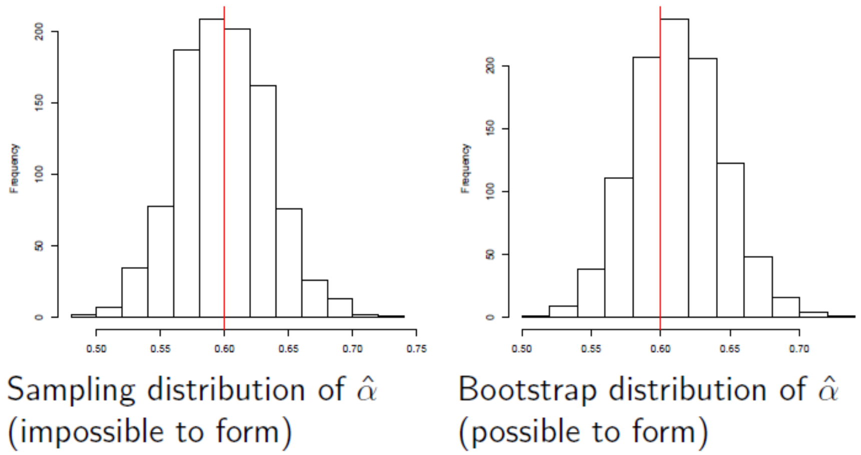
- We know that as we include more partitions or splits in a tree, the bias decreases. However, as the number of partitions approaches n, we end up with very few observations per node, resulting in a high variance.
- This high level of variance means that slight changes in our data can produce drastically different trees. If the variance was small, slight changes to our datasets would produce similar trees.
- Bagging, or Bootstrap AGgregation, is a procedure that reduces variance.
- Aggregation refers to combining data/predictions, such as through a sum or average.
- The idea is to take a low bias, high variance estimator and try to stabilize the variance.
- For example, we know from the Central Limit Theorem that if we have n uncorrelated observations, each with variance σ^2, then the variance of the sample mean is σ^2/n.
- In terms of trees, if we have B uncorrelated training sets, we could create B different trees, $\hat{f}^1(X), ..., \hat{f}^B(X)$, and average them together, $\hat{f}_B(X) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^b(X)$.
- This will drive the variance of the procedure to zero: $\mathbb{V}(\hat{f}_B(X)) = \frac{1}{B}\mathbb{V}(\hat{f}^b(X)) = \sigma^2/B$.
- This procedure does not affect the bias: $Bias(\hat{f}_B(X)) = bias(\hat{f}^b)$, for any b.
- In practice, we rarely have access to several different training sets. We use bootstrapping to simulate having many training sets.
- The bootstrap can be used to estimate uncertainty without Gaussian approximations, making it widely applicable.



This is the sampling distribution of $\hat{\alpha}$

3 Bootstrapping

- Suppose we want to invest in two financial instruments, X and Y. The return on these investments are random processes, but we want to minimize risk (volatility) when we allocate our money. This means we want to find $\alpha \in (0,1)$ such that it minimizes $\text{Var}(\alpha X + (1-\alpha)Y)$.
- The minimizing α is $\alpha_* = \frac{\sigma_Y^2 - \sigma_{XY}^2}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}^2}$, where σ_{XY}^2 represents the covariance between X and Y.
- We can estimate the minimizing α with $\hat{\alpha} = \frac{\hat{\sigma}_Y^2 - \hat{\sigma}_{XY}^2}{\hat{\sigma}_X^2 + \hat{\sigma}_Y^2 - 2\hat{\sigma}_{XY}^2}$
- While we have an estimator of α, we don't have an estimate of its variability. It will take a long time for asymptotic properties to kick in. It would be nice to have thousands of estimates of α so we could estimate the standard error.
- If we could simulate a large number of draws of the data, we could get a large number of estimates for α, and then calculate a standard error.
- The mean of these estimates is $\bar{\alpha} = \frac{1}{1000}\sum_{r=1}^{1000} \hat{\alpha}_r = 0.559$, which is extremely close to the truth of $\alpha = 0.6$.
- The standard error is $\sqrt{\frac{1}{1000-1}\sum_{r=1}^{1000}(\hat{\alpha}_r - \bar{\alpha})^2} = .035$
- Now that we have a standard error, we have an idea of the accuracy of $\hat{\alpha}$ for a single sample. We can compute a confidence interval to find the plausible values of our estimate: $\hat{\alpha} \in (\alpha - 2 * .035, \alpha + 2 * .035)$.
- In practice, we cannot typically draw numerous independent samples from a population. However, we can bootstrap!
- To bootstrap, we draw a large number of samples from our observed data. We sample our observed data with replacement, meaning that the same data point can be drawn more than once. This places $1/n$ mass on each of our n data points.
- So, if we denote the population distribution as $\mathbb{P}$ and denote the empirical distribution as $\hat{\mathbb{P}}$, we know that $||\mathbb{P} - \hat{\mathbb{P}}||$ gets exponentially smaller as n gets larger, since our empirical distribution will be more representative of the population distribution as our sample size increases.
- If we let $\hat{\mathbb{P}}_B$ denote our bootstrapped distribution, we see that $||\mathbb{P} - \hat{\mathbb{P}}_B|| \leq ||\mathbb{P} - \hat{\mathbb{P}}|| + ||\hat{\mathbb{P}} - \hat{\mathbb{P}}_B||$, and we know $||\hat{\mathbb{P}} - \hat{\mathbb{P}}_B||$ will be small by Stein's Lemma.



Sampling distribution of $\hat{\alpha}$ (impossible to form)

Bootstrap distribution of $\hat{\alpha}$ (possible to form)

Figure 1: Note that the bootstrap distribution of $\hat{\alpha}$ is good approximation to the sampling distribution of $\hat{\alpha}$.

4 Bootstrapping Example

- As an example, suppose we observed the following data: $\mathcal{D} = (4.3, 3, 7.2, 6.9, 5.5)$.
- Now we draw with replacement from our dataset to create several new samples. Classically, we draw the same number of points as our original dataset for each sample, but there are versions of bootstrapping which include corrections, change of sample sizes, etc.

  $\mathcal{D}_1^* = (7.2, 4.3, 7.2, 5.5, 6.9)$

  $\mathcal{D}_2^* = (6.9, 4.3, 3.0, 4.3, 6.9)$

  ...

  $\mathcal{D}_B^* = (4.3, 3.0, 3.0, 5.5, 6.9)$
- Now we can form the bootstrap mean and the standard error of the mean:

  $mean_B = \frac{1}{B}\sum_{b=1}^{B} \hat{\alpha}_b^*$

  $\text{SE}_B = \sqrt{\frac{1}{B}\sum_{b=1}^{B}(\hat{\alpha}_b^* - mean_B)^2}$
- To summarize: suppose we have data $\mathcal{D} = (Z_1, ..., Z_n)$, where $Z_i = (X_i, Y_i)$, and we want information about the sampling distribution of some statistic $\hat{f}$ that is trained on $\mathcal{D}$. All we have to do is:
  1. Fix a large number B
  2. For each $b = 1, ..., B$, form a new bootstrap draw from $\mathcal{D}$, called $\mathcal{D}^*$
  3. Compute $\hat{f}_b^*$ from each $\mathcal{D}^*$.
  4. Now we can estimate the distribution of $\hat{f}$ by looking at the distribution of $\hat{f}_b^*$.

5 Bagging and Bootstrapping

- Now, instead of having B separate training sets for the Bagging process, we train on B bootstrap draws: $\hat{f}_1^*(X), ..., \hat{f}_B^*(X)$.
- Then we average them together (this is the aggregation part): $\hat{f}_{bag}(X) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}_b^*(X)$.

- Now we can apply this process to trees:
    1. Choose a large number B.
    2. For each $b = 1, ...B$, grow an unpruned tree on the b^{th} bootstrap draw from the observed data.
    3. Average all of the trees together.
- B is not a tuning parameter. We can think of it as more of a time constraint.

6 Results of Bagging

- When we choose to use Bagging, we have to decide whether to average the probabilities of $\hat{f}^b(X)$ together or average the classifications of $\hat{f}^b(X)$ together.
- The predictions from the different trees will be correlated, since the bootstrap draws are not independent.
- Each unpruned tree will have high variance and low bias. Therefore, averaging many trees together results in an estimator that has lower variance and still low bias.
- We can no longer look directly at a dendrogram since we are growing such a large number of trees; we cannot form a nice diagram that shows the segmentation of the predictor space.
- However, bagging does result in other helpful information:

    Mean decrease variable importance

    Out-of-Bag error estimation

    Permutation variable importance

    Proximity Plot

7 Mean Decrease Variable Importance

- Note that at every split of a node, the loss function decreases.
- We can add up the amount of decrease for each covariate over all of the trees to get an indication of feature importance.
- We know intuitively that if a large drop in the loss function occurs when the feature is split upon, then the feature is important.
- To find the mean decrease variable importance:

    1. For each of the B trees and each of the p features, we record the amount that the Gini index or cross-entropy is reduced by the presence of that feature.

    2. Report the average reduction over all B trees

    3. Features for which this average (relative to the averages for other features) is high are considered important.

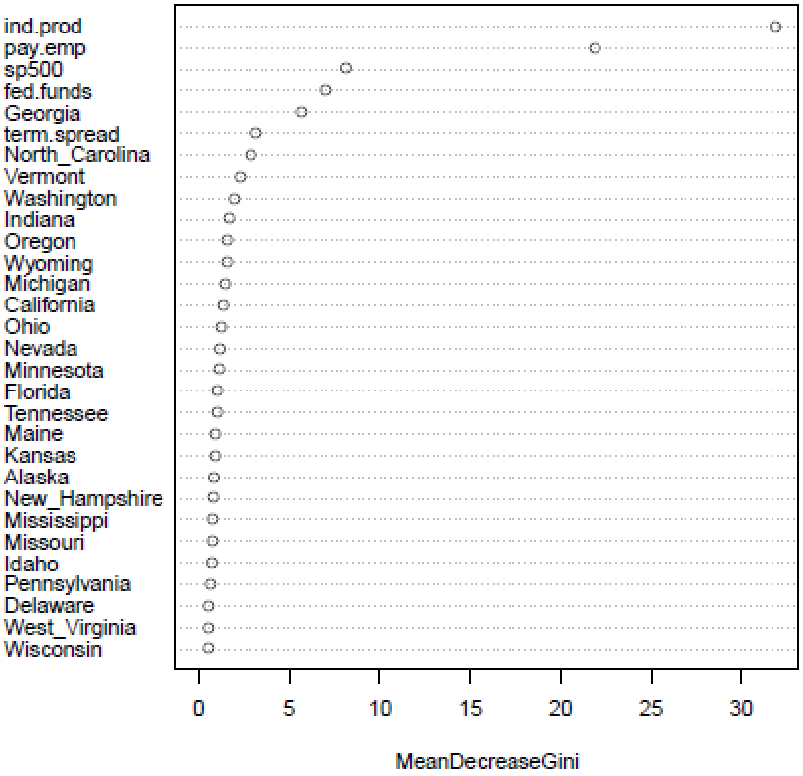


Figure 2: This plot shows us the Mean Decrease Variable Importance for the listed features, with more important variables at the top of the plot.

8 Out of Bag Samples (OOB)

- It has been shown that, on average, drawing n samples with replacement from n observations results in 2/3 of the observations being selected for each sample.
- The approximate 1/3 of observations that are not selected are referred to out-of-bag (OOB).
- This implicitly creates a training and test sets, which are uncorrelated for a particular b (assuming the original data is a random sample).
- We can think of this as cross-validation for free (we did not have to divide our original dataset into training and test sets, sacrificing our sample size).
- This provides us with a free estimation of prediction risk for each tree.
- We can get an overall estimate of prediction risk by averaging these prediction risk estimates together over all of our bootstrapped trees.

9 Permutation Variable Importance

- If a feature is highly important, the out-of-bag prediction error should increase considerably after permuting the out-of-bag values for the feature.
- To compute the permutation variable importance for the j^{th} feature:
  1. Form a bootstrap draw, $\mathcal{D}^b$. Train an unpruned tree on $\mathcal{D}^b$, which we denote as $\hat{f}^b$.
  2. Record the out-of-bag prediction accuracy for the b^{th} tree.
  3. Randomly permute the j^{th} feature in the out-of-bag sample.
  4. Recompute the prediction error and record the change from step 1.
- Important features will have a high change in prediction error (relative to other features).

10 Proximity Plot

- For the b^{th} tree, we can examine which out-of-bag observations are assigned to the same terminal node.
- This is similar to the concept of nearest neighbor; if two observations end at the same terminal node, they should be similar to one another.
- To form a proximity plot:

  1. Form a $n \times n$ matrix P and increment $P[i,i'] \leftarrow P[i,i'] + 1$ if Z_i and $Z_{i'}$ are predicted to the same terminal node.

  2. Use a dimension reduction technique to visualize the data in 2-3 dimensions. For example, multidimensional scaling is most commonly used because between observation ditances are preserved.
- Even if the data consists of both qualitative and quantitative variables, or if the data has a high dimension, we can still visualize their similarity through the forward operator of the bagged estimator.

11 Random Forest

- The Random Forest is an extension of Bagging which deals with the problem of correlated trees from bootstrapped samples.
- We know that predictions from correlated trees are similar. Bagging won't work if the trees are too correlated; the variance won't be reduced.
- To create a random forest:
  1. Draw a bootstrap sample and start to build a tree.
  2. At each split, randomly select m of the possible p features to use for the split.
  3. Select a new sample of size m of the features for each additional split.
- Usually, we use $m = \sqrt{p}$. This forces the trees to be different from each other, since at each split, we do not consider the majority of the features.
- Random forests are useful in situations where there is 1 very important feature and several not so important features:

  If we consider all of the features at each split, every tree will contain this feature.

  This results in very similar (correlated) trees.

  Averaging highly correlated trees will not result in as much variance reduction as averaging uncorrelated trees.

  Preventing some trees from using this feature results in different trees, which means they are less correlated.
- Bagging is the same as a Random Forest where we let $m = p$, since we consider each feature at each split.
- If we average B i.i.d. random variables, the variance of the average is σ^2/B.
- If we average B random variables with correlation ρ, the variance of the average is $\rho\sigma^2 + (1-\rho)\sigma^2/B$
- As we increase B, the second term of the variance goes to zero, but we cannot reduce the first term of the variance. This demonstrates how correlation limits the benefit of averaging.