Estimating the Operating Characteristics of Ensemble Methods

Anthony Gamst Jay-Calvin Reyes Alden Walker

September 2017

Suppose we would like to compare the performance of two binary classification algorithms. The receiver operating characteristic (ROC) curve for each algorithm is shown in Figure 1.

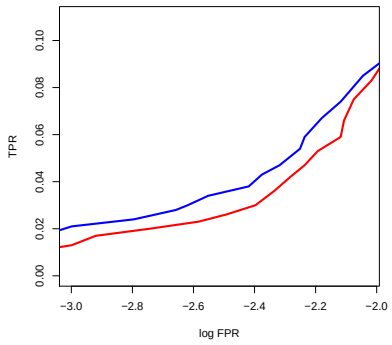


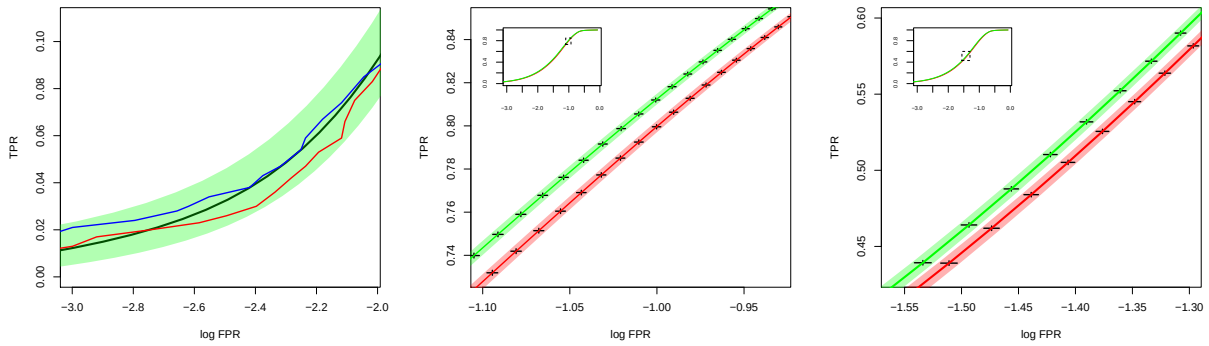
Figure 1: ROC curves for two binary classifiers. Which algorithm is better?

The obvious conclusion is that since the blue ROC curve is higher, the corresponding algorithm must be better. This conclusion is not correct. Both ROC curves were produced using the same algorithm (random forest with 256 trees), the same training data set, and the same test data set. Each forest made different choices when building its decision trees, leading to random variation in the ROC curves.

To determine whether one binary classification algorithm is better than another we need to construct reliable confidence bands around the ROC curves. This paper describes a computationally efficient technique to do this for ensemble methods that base their decision on a vote among weak classifiers. The main idea is to bootstrap the weak classifiers and to Poisson bootstrap [2] the feature vectors in the test set.

Bootstrapping a model can require a huge amount of work if we need to repeatedly fit the model to large data sets. Fortunately, in many cases the technique described here lets us determine the effect of doing infinite resampling *without actually refitting a single model*. Suppose that each weak classifier returns a prediction of either 0 or 1 (as opposed to a value in the unit interval, say). Then the points in the ROC curve can be expressed as a function of these votes (which are random variables). We compute the mean and variance of this function to produce error bars at each point along the ROC curve. Furthermore, most of the computation does not depend on the number of weak classifiers in the ensemble, so for very little additional work we can determine how many weak classifiers we should train to achieve a particular level of predictive accuracy.

Applying the technique to the two ROC curves above shows that they both lie within the confidence band for a 256-tree random forest as shown in Figure 2a. Therefore, the variability between the red and blue ROC curves is unsurprising.



(a) 256-tree random forests (b) with and without bootstrap aggregation (c) splitting on $\sqrt{m}$ and $2\sqrt{m}$ features

Figure 2: ROC curves for variations of random forests. Shaded regions are approximate 95% confidence bands. Two standard deviation confidence intervals (black) for log FPR and TPR are shown at several points along the curve. Insets show full ROC curves.

We also applied the technique to random forests with various voting and splitting rules and got some surprising results. In particular, we found that *not* bootstrapping the training data set when producing each tree led to statistically significant gains in predictive accuracy (see Figure 2b). Trying more than the standard number of features to split each node also improved the predictive accuracy (see Figure 2c). We repeated the experiment using several datasets with very different characteristics, and the improvements were present every time.

The results are surprising because Breiman [1] recommended using a different bootstrap sample of the training data to build each tree. He also suggested trying $\sqrt{m}$ features to split each node, where m is the total number of features in the data set. Some widely-used implementations of random forests (such as `sklearn`) follow these recommendations by default.

Acknowledgments

We would like to thank Larry Carter, Skip Garibaldi, Kyle Hofmann, Doug Jungreis, Dan Mauldin, and Kartik Venkatram for helpful comments and suggestions.

References

[1] L. Breiman, Random Forests. Machine Learning, 45, 5-32, 2001.

[2] N. Chamandy, O. Muralidharan, A. Anjmi, and S. Naidu, Estimating Uncertainty for Massive Data Streams. Google Technical Report, 2012.