Miriam Project

Miriam Mercuri 5207057

2023-06-09

Introduction

In a typical prediction problem, there is a trade-off between two factors: bias and variance. Bias refers to the error introduced by simplifying assumptions in a model, while variance refers to the amount by which predictions vary for different samples of data.

When we try to improve the fit of a model, there comes a point where increasing the precision of the fit leads to higher variance in predictions for future observations. On the other hand, reducing prediction variance can result in a higher expected bias for future predictions.

To address this trade-off, Breiman (1996) introduced a technique called bagging, which stands for "Bootstrap AGGregatING." The idea behind bagging is simple. Instead of relying on a single model fit to the observed data for making predictions, multiple bootstrap samples (randomly selected subsets) of the data are created. Each of these samples is used to fit a model, and the predictions from all the fitted models are averaged to obtain the bagged prediction.

Breiman explains that bagging works particularly well for modeling procedures that are sensitive to small changes in the data, such as classification and regression trees (CART). He also provides a theoretical explanation of how bagging reduces the mean-squared prediction error for such unstable procedures.

Unstable classifiers, such as trees, characteristically have high variance and low bias. Stable classifiers have low variance, but can have high bias.

Instability is an essential ingredient for bagging to improve accuracy.

A learning set of L consists of data $\{(y_n, x_n), n = 1, ..., N\}$ where the y's are either class labels or a numerical response. Assume we have a procedure for using the learning set to form a predictor $\delta(x, L)$. Now, suppose we are given a sequence of learning sets $\{L_k\}$ each consisting of N independent observations from the same underlying distribution as L. Our mission is to use the $\{L_k\}$ to get a better predictor than the single learning set predictor $\delta(x, L)$. The restriction is that all

Comparison

Bootstrapping is random sampling with replacement from the available training data. Bagging (= bootstrap aggregation) is performing it many times and training an estimator for each bootstrapped dataset.

When you just/only do bootstrap without an aggregation at the end you only use the standard deviation of the bootstraps replications together, than you call it bootstrap.

If you do the same, but at the end you take an aggregation of voting or anything else, than you call it bagging. They are both ensemble technique. One of the technique in bagging is called as Random Forest. In Random Forest you basically use multiple decision trees.

Random Forest

Random forests are an improvement over bagged trees that aim to make the trees less similar to each other. In bagging, we create multiple decision trees using different training samples. However, in random forests, when building these trees, we make a small change. Instead of considering all predictors at each split, we randomly select a subset of predictors and choose only one from that subset for the split. We refresh this subset for each split, usually picking the square root of the total number of predictors.

This tweak is done because in bagged trees, if there is a strong predictor in the dataset, most or all of the trees will use that predictor in the top split. As a result, the predictions from these trees will be highly correlated. Unfortunately, averaging highly correlated predictions doesn't reduce the variability as much as averaging uncorrelated predictions. Therefore, bagging doesn't significantly reduce the variance compared to a single tree in such cases.

Random forests solve this problem by restricting each split to consider only a subset of predictors. This means that, on average, a portion of the splits won't even consider the strong predictor. By doing this, other predictors have a better chance to contribute to the splits. This process "decorrelates" the trees, making their average less variable and more reliable.

The main difference between bagging and random forests lies in the choice of the predictor subset size. If we use all predictors, it's equivalent to bagging. However, when using the square root of the total predictors, random forests show a reduction in the test error.

Using a small subset of predictors is particularly helpful when we have many correlated predictors.

Just like bagging, random forests don't overfit if we increase the number of trees (denoted as B). Therefore, in practice, a sufficiently large value of B is used until the error rate stabilizes.

In the process of constructing a random forest, multiple decision trees are created. Each tree is generated independently, and for each tree, a random vector Θ_k is generated. The generation of Θ_k is unrelated to the previously generated random vectors $\Theta_1, ..., \Theta_{k-1}$, but they are all drawn from the same distribution.

To grow a tree, the training set and the corresponding Θ_k are used. This combination results in a classifier $h(\mathbf{x}, \Theta_k)$, where $\mathbf{x}$ represents an input vector. In bagging, for example, Θ is generated by counting the number of darts that randomly fall into N boxes, with N being the number of examples in the training set. In random split selection, the content of Θ depends on its usage in constructing the tree.

Once a large number of trees are generated, they collectively contribute to the final classification decision. Each tree casts a vote, and the class with the highest number of votes is considered the most popular. This aggregation of trees is what we refer to as a random forest.

A random forest is a classifier consisting of a collection of tree-structured classifiers {$h(\mathbf{x},\Theta_k)$, k=1,... } where the {Θ_k} are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input $\mathbf{x}$.

References

Breiman L. (1996). *Bagging Predictors*, Statistics Department, University of California, Berkeley.

Breiman L. (2001). *Random forest*, Statistics Department, University of California, Berkeley.

Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani (2021). *An Introduction to Statistical Learning with Applications in R*, Second Edition.

Tae-Hwy Lee, Aman Ullah and Ran Wang. *Bootstrap Aggregating and Random Forest.*

Andreas Buja, Werner Stuetzle.*Observation on Bagging.* University of Pennsylvania and University of Washington.

Peter Bühlmann, Bin Yu (2002).*Analyzing bagging*, ETH Zürich and University of California, Berkely.