An Introduction to Tree-Based Methods

William Suzuki

FEARP-USP

2019

Regression Trees

Types of methods: Regression Trees, Bagging, Random Forests, Boosting.

Regression Trees are simple and useful for interpretation.
Problem: poor prediction. Typically worse than: LASSO, Ridge, Splines, Generalized Additive Models (GAM).

Bagging, Random Forests, Boosting: combine a large number of trees. Improve prediction accuracy.
Problem: at the expense of some loss in interpretation.

Regression Trees

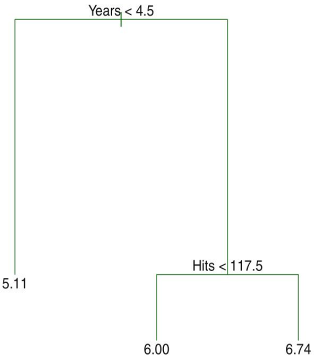
Source: *Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani;* An Introduction to Statistical Learning: with Applications in R

Motivation: Predicting Baseball Players' Salaries Using Regression Trees

$y = \{\text{log salary of a baseball player}\}$
$X = \{\text{years playing; last year's number of hits}\}$

Regression Trees

Figure: Hitters regression tree example



$y = \{\texttt{Salary}\}$

$X = \{\texttt{Years, Hits}\}$

Methodology

We define the following regions for the covariates X:

$R_1 = \{X|\texttt{Years<4.5}\}$

$R_2 = \{X|\texttt{Years>=4.5, Hits<117.5}\}$

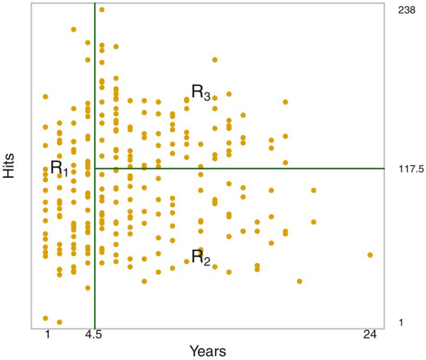
$R_3 = \{X|\texttt{Years>=4.5, Hits>=117.5}\}$

R_1, R_2, R_3 are called *terminal nodes* or *leaves*.

The points along the tree where the predictor space is split are referred to as *internal nodes*.

Regression Trees

Figure: `Hitters` three-region partition



Regression Trees

Interpretation:

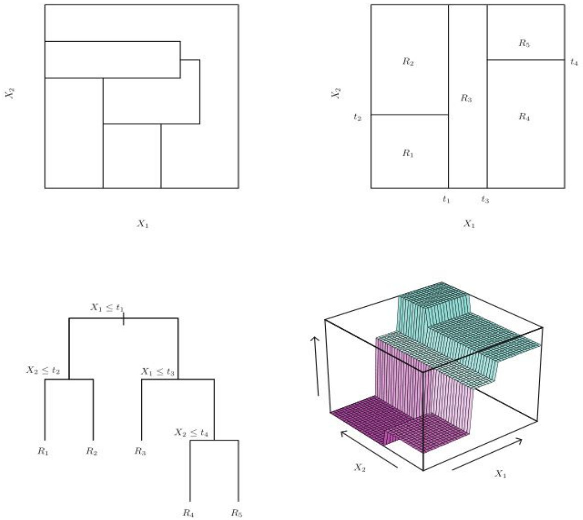
`Years` is the most important factor in determining `Salary`. Less experience means lower salary.
`Hits` are not model relevant for lesser experienced players.

`Hits` are model relevant for more experienced players (5+ years).
More `Hits` means higher `Salary`.

Good: easy to interpret, and nice graphical representation.
Problem: model too simple.

Regression Trees

Figure: Visualize Regression Tree



Regression Trees

Algorithm steps:

For any j and s, we define the pair of half-planes:

$$R_1(j,s) = \{X|X_j < s\} \text{ and } R_2(j,s) = \{X|X_j \geq s\}$$

we seek the value of j and s that minimize the equation

$$\min_{j,s} \sum_{i \text{ in } R_1} (y_i - y_{R_1})^2 + \sum_{i \text{ in } R_2} (y_i - y_{R_2})^2$$

Regression Trees

Tree Pruning

Problem: The algorithm goes till a stop criterion, the resulting tree might be too complex. Overfitting.

Strategy: grow a very large tree T_0 and then prune T_0 in order to obtain a *subtree*.

Regression Trees

Weakest Link Pruning

α is a nonnegative tuning parameter.

For each subtree $T \subset T_0$

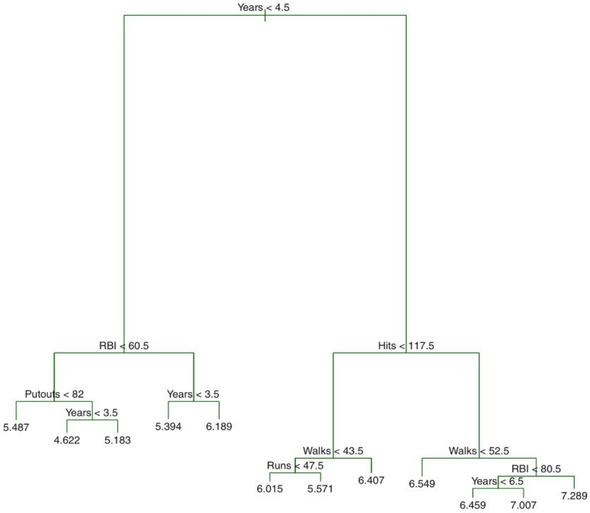
$$\sum_{m=1}^{|T|} \left[\sum_{i \text{ in } R_m} (y_i - y_{R_m})^2 \right] + \alpha |T|$$

$|T|$ number of leafs in the tree.

Use cross-validation to select α.

Regression Trees

Figure: Unpruned Regression Tree for `Hitters`



Regression Trees

Figure: MSE graphs for `Hitters`

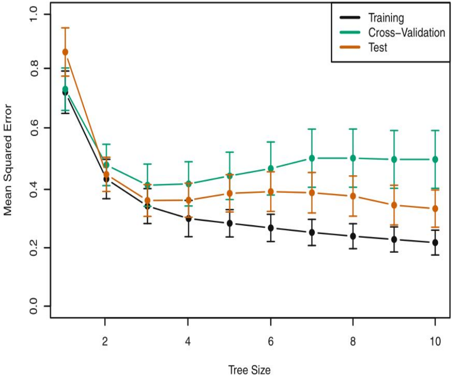


FIGURE 8.5. *Regression tree analysis for the* `Hitters` *data. The training, cross-validation, and test MSE are shown as a function of the number of terminal nodes in the pruned tree. Standard error bands are displayed. The minimum cross-validation error occurs at a tree size of three.*

Classification Trees

A classification tree is very similar to a regression tree, except that it is tree used to predict a qualitative response rather than a quantitative one.

In classification trees we predict that each observation belongs to the most commonly occurring class of training observations in the region to which it belongs.

$$\forall m \qquad G_m = \sum_{k=1}^{K} p_k(1 - p_k) \qquad D_m = \sum_{k=1}^{K} p_k \log p_k$$

Regression Trees

- Trees are very easy to explain to people.
- Trees can be displayed graphically, and are easily interpreted.
- Trees can easily handle qualitative predictors.
- (bad) Comparatively low predictive power.

Next: Bagging, Random Forests, Boosting.

Bagging

Bootstrap aggregation = *bagging* is a general-purpose procedure for reducing the bagging variance of a statistical learning method.

Bagging typically results in improved accuracy over prediction using a single tree. Unfortunately, however, it can be difficult to interpret the resulting model.

The bagging estimator is:

$$\hat{f}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}_b(x)$$

Out-of-Bag

out-of-bag (OOB)

One can show that on average, each bagged tree makes use of around two-thirds of the observations.

This will yield around B/3 predictions for the ith observation.

OOB MSE (for a regression problem) or classification error (for a classification problem)

It can be shown that with B sufficiently large, OOB error is virtually equivalent to leave-one-out cross-validation error.

Methodology

Random forests provide an improvement over bagged trees by way of a random forest small tweak that decorrelates the trees. As in bagging, we build a number of decision trees on bootstrapped training samples.
each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p predictors.

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

Methodology

This may sound crazy, but it has a clever rationale. Suppose that there is one very strong predictor in the data set, along with a num- ber of other moderately strong predictors. Then in the collection of bagged trees, most or all of the trees will use this strong predictor in the top split. Consequently, all of the bagged trees will look quite similar to each other.Hence the predictions from the bagged trees will be highly correlated.Un- fortunately, averaging many highly correlated quantities does not lead to as large of a reduction in variance as averaging many uncorrelated quanti- ties.

We can think of this process as decorrelating the trees, thereby making the average of the resulting trees less variable and hence more reliable

Regression Trees

Figure: Classification error graphs for `Heart`

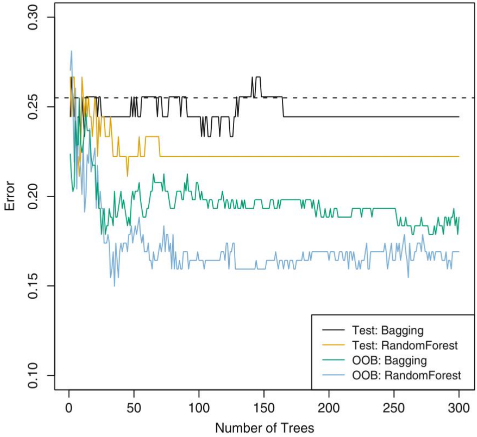


FIGURE 8.8. *Bagging and random forest results for the* `Heart` *data. The test error (black and orange) is shown as a function of B, the number of bootstrapped training sets used. Random forests were applied with $m = \sqrt{p}$. The dashed line*

Regression Trees

Figure: Classification error graphs for `Heart`

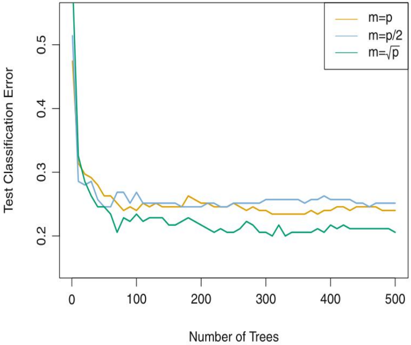


FIGURE 8.10. *Results from random forests for the 15-class gene expression data set with $p = 500$ predictors. The test error is displayed as a function of the number of trees. Each colored line corresponds to a different value of m, the number of predictors available for splitting at each interior tree node. Random forests ($m < p$) lead to a slight improvement over bagging ($m = p$). A single classification tree has an error rate of 45.7%.*

Boosting

Algorithm steps: Boosting for Regression Trees

1. Set $f(x) = 0$ and $r = y$ for all i in the training set.
2. For $b = 1, 2, ..., B$, repeat:
   2.1 Fit a tree f_b with d splits ($d + 1$ leafs) to the training data (X, r).
   2.2 Update f by adding in a shrunken version of the new tree:
   $$f(x) \leftarrow f(x) + \lambda f_b(x)$$
   2.3 Update the residuals,
   $$r \leftarrow r - \lambda f_b(x)$$
3. Output the boosted model,
$$f(x) = \sum_{b=1}^{B} \lambda f_b(x)$$

Boosting

Usually $\lambda = 0.01$ or 0.001

Methodology

References