Tree Models

1 Introduction

[TBW]

2 Tree models

Assume input variables $\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_p] \in \mathcal{X}$, and output variables $\mathbf{y} \in \mathbb{R}$ or a categorical variable $\mathbf{y} \in \{1, \ldots, K\}$. The tree models are created by recursively splitting input space $\mathcal{X}$ on which the input variables are defined. Such recursive splitting creates partitions $\{R_n\}$ in the input space $\mathcal{X}$, and R_n defines the n-th region in the input space. We define the output prediction of the tree model as follows,

$$\hat{f}(\mathbf{x}) = \sum_n c_n I(\mathbf{x} \in R_n),$$

in which R_n is different input regions created by the tree on $\mathcal{X}$. For the case of classification problem, c_n is the *class label* (a categorical value), and for the case of regression it is usually a continuous value representing the *predicted output value*. In other words, label (value) of one given input, is the average of the labels(values) for any given input that belongs to the same region.

Now the problem is that how can we really create a desired suitable partitioning of the input space $\mathcal{X}$, in order to have an appropriate generalization(avoiding both underfitting and overfitting).
Let's say we have a tree structure and the partitioning created in the input space denoted by R_n. To minimize the error predictive and the real output, we want to minimize the following objective,

$$\min_{c_m} \sum_{m=1}^{M} \sum_{\mathbf{x}_i \in R_m} (y_i - c_m)^2 .$$

By fixing the structure of the tree, the optimal value for each region R_m, is the mean of the values for that region, i.e. $c_m = \text{mean}\left(\{y_i | \mathbf{x}_i \in R_m\}\right)$. To create a measure for splitting the branches, we need to define a criterion, namely *goodness of split function*, $\Phi\left(t, (i,j)\right)$. For the case of regression we can use the following definition,

$$\Phi\left(t, (i,j)\right) = \text{RSS}(t) - \left[\text{RSS}(t_R) - \text{RSS}(t_L)\right], \ \text{RSS}(t) = \sum_{R_m \in \{R_i\}} \sum_{\mathbf{x}_i \in R_m} \left(y_i - c_t\right)^2. \tag{1}$$

One important property for $\Phi(.)$ is that it is always positive. For the case of a categorical classification problem, for K categories, one could define it in a different way,

$$\Phi\left(t, (i,j)\right) = i(t) - \left[p_R.i(t_R) + p_L.i(t_L)\right], \ i(t) = I\left(\hat{p}_t(1), \ldots, \hat{p}_t(K)\right), \tag{2}$$

in which $\hat{p}_t(j)$ is the impurity measure of the class j at node t and $\sum_j p_j = 1$. The function $I(.)$ is called *impurity measure*, and has it's maximum value when $p_1 = p_2 = \ldots = p_K = \frac{1}{K}$. It is minimum when $p_j = 1$. A popular example of impurity measures is *entropy measure*, $\sum_{j=1}^{K} p_j \left(1 - p_j\right)$.
Generally, one would prefer to grow the tree as much as possible and then prune it using *cost-complexity measure*,

$$R_\alpha(T) = R(T) + \alpha \left|T\right|.$$

In the above formulation, $R(.)$ is the error term for tree. In the regression defined as eq. 1, and in classification defined as eq. 2. The parameter α is the *relaxation* parameter. We prune the tree based the *relaxed* criterion, which possibly increase the error on the training data, while gaining more generalization over unseen data.

One unanswered question is the choice of α. One may follow different tastes for this choice, e.g. AIC, BIC, cross-validation.

2.1 Advantages and disadvantages of tree models

Tree models bring some advantages over the other models. These models are called *non-parametric* since the tree structure is created only based on the input data, and parameters of the information gains. So the tree is implicitly created by the data, and is not manually engineered. This also makes the interpretation of the structures easier. By cleverly tuning the splitting criterion one can make it robust to outliers.

2.2 What is a random forest?

In*Random Forest* the goal is to grow a group of trees, to decrease variance in the output and increase the certainty and stability of regression/classification done using the conventional tree models. In order to do so, we use the method of *Bootstrap Aggregation* or simply *Bagging*. In Bagging, instead of using all training data for creating the model, we choose only a subset of the data, to create a model. We then consider the result of aggregation among trained function on bootstrap samples, which is more consistent. The algorithm is shown in Alg. 1.

Algorithm 1: Bootstrap Aggregation or Bagging.

Input: Training data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$
Output: Regression/classification, $\hat{f}_{\text{bag}}(x)$
Generate random Bootstrap samples, $\mathcal{D}^b = \left\{(x_i^b, y_i^b)\right\}_{i=1}^{M}$, where $b = 1, \ldots, B$.
Train $\hat{f}_{\text{bag}}^b$ using $\mathcal{D}^b$.
For regression: Averaging: $\frac{1}{B}\sum_{b=1}^{B} \hat{f}_{\text{bag}}^b$.
For classification: Majority voting among $\hat{f}_{\text{bag}}^1, \ldots, \hat{f}_{\text{bag}}^B$.

Based on the idea of Bagging, we train tree model on each of the Bootstrap samples. I could assume various restrictions while growing tree models. One idea would be to randomly select m variables from the p variables, and then pick the best split among them. Selecting sensible values for m could reduce the correlation between trees in the forest.
One good idea to find an estimate of error is to use those sample point that are not included in $\mathcal{D}^b$; these sample points are called Out-Of-Bag (OOB) samples.
One could calculate an estimate of *variable importance* for each variable, using Gini index, which is the splitting criterion. Improvement in the split-criterion (e.g. Gini index) is the importance measure attributed to the variable. The importance measure is the accumulated improvement in the split-criterion over all the trees in the forest for each variable.
Other than using the above method for variable importance one could use shuffling the values of the m-th variable inside the OOB samples, and then calculating the difference of the number of correctly classified before and after this change; repeat this work over all of the trees and average the overall result.

3 Bibliographical notes

In preparation of this document I have used lecture notes for STAT:542 at UIUC by Feng Liang.

References