Machine Learning 202

Homework 2

Mike Bowles, PhD & Patricia Hoffman, PhD

This problem refers to Professor Jerome Friedman's paper: Greedy Function Approximation: A Gradient Boosting Machine (see reference below). Use Friedman's equations (39) and (40) on page 16 to generate 100 continuous (real valued) functions. Fit gradient boosted trees using a squared error loss function. Summarize the performance of the 100 gradient boosted trees that the r function, gbm, created.

Here is the link to the paper:

Greedy Function Approximation: A Gradient Boosting Machine by Jerome Friedman

http://www-stat.stanford.edu/~jhf/ftp/trebst.pdf

Page 16 follows

6.1 Random function generator

One of the most important characteristics of any problem affecting performance is the true underlying target function $F^*(\mathbf{x})$ (1). Every method has particular targets for which it is most appropriate and others for which it is not. Since the nature of the target function can vary greatly over different problems, and is seldom known, we compare the merits of regression tree gradient boosting algorithms on a variety of different randomly generated targets. Each one takes the form

$$F^*(\mathbf{x}) = \sum_{l=1}^{20} a_l g_l(\mathbf{z}_l). \tag{39}$$

The coefficients $\{a_l\}_1^{20}$ are randomly generated from a uniform distribution $a_l \backsim U[-1, 1]$. Each $g_l(\mathbf{z}_l)$ is a function of a randomly selected subset, of size n_l, of the n–input variables $\mathbf{x}$. Specifically,

$$\mathbf{z}_l = \{x_{P_l(j)}\}_{j=1}^{n_l}$$

where each P_l is a separate random permutation of the integers $\{1, 2, \cdots, n\}$. The size of each subset n_l is itself taken to be random, $n_l = \lfloor 1.5 + r \rfloor$, with r being drawn from an exponential distribution with mean $\lambda = 2$. Thus, the expected number of input variables for each $g_l(\mathbf{z}_l)$ is between three and four. However, most often there will be fewer than that, and somewhat less often, more. This reflects a bias against strong very high order interaction effects. However, for any realized $F^*(\mathbf{x})$ there is a good chance that at least a few of the 20 functions $g_l(\mathbf{z}_l)$ will involve higher order interactions. In any case, $F^*(\mathbf{x})$ will be a function of all, or nearly all, of the input variables.

Each $g_l(\mathbf{z}_l)$ is an n_l–dimensional Gaussian function

$$g_l(\mathbf{z}_l) = \exp(-\frac{1}{2}((\mathbf{z}_l - \mu_l)^T \mathbf{V}_l (\mathbf{z}_l - \mu_l)) \tag{40}$$

where each of the mean vectors $\{\mu_l\}_1^{20}$ is randomly generated from the same distribution as that of the input variables $\mathbf{x}$. The $n_l \times n_l$ covariance matrix $\mathbf{V}_l$ is also randomly generated. Specifically,

$$\mathbf{V}_l = \mathbf{U}_l \mathbf{D}_l \mathbf{U}_l^T$$

where $\mathbf{U}_l$ is a random orthonormal matrix (uniform on Haar measure) and $\mathbf{D}_l = diag\{d_{1l} \cdots d_{n_l l}\}$. The square–roots of the eigenvalues are randomly generated from a uniform distribution $\sqrt{d_{jl}} \backsim U[a, b]$, where the limits a, b depend on the distribution of the input variables $\mathbf{x}$.