Gradient Boosting

Suppose we have a family of weak classifiers $\mathcal{F}$.

Set $f_0 = \underset{f \in \mathcal{F}}{\arg\inf} \sum_{i=1}^{n} L(f(x_i), y_i)$

Update the estimate for $m = 1, 2, \dots, M$

$$f_m = \sum_{i=0}^{m} h_i \qquad h_i \in \mathbb{R}^N \; ; \; h_0 = f(x).$$

Steepest descent: $h_m = -\underbrace{\rho_m}_{\text{length}} \underbrace{g_m}_{\text{direction}} \qquad g_m \in \mathbb{R}^n$

$$g_m = \nabla_{f_{m-1}} L(f) = \nabla_{f_{m-1}} \sum_{i=1}^{n} L(y_i, f_m(x_i)) \qquad \text{(steepest direction)}$$

$$\rho_m = \underset{\rho}{\arg\min}\, L(f_{m-1} - \rho g_m) \qquad \text{(best length)}$$

$$f_m = f_{m-1} - \rho_m g_m \qquad \text{(step downwards)}$$

Random Forests

Suppose we have a tree method $h(\Theta, x)$ where Θ is the parameter set.

Algorithm:

for $b = 1, \dots, B$

a. Draw a bootstrap sample of data Z^b of size N.

b. Grow a tree T_b on Z^* by the following procedure.

(i) Select $m \ll p$ variables
(ii) Pick the best variable/split point
(iii) Split node

2. Output $\{T_b\}_{b=1}^{B}$.

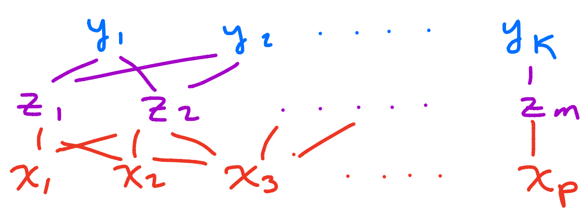
Prediction Rule: $\hat{f}_{RF}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$ (regression)

$\hat{f}_{RF}(x) =$ majority vote $\{T_b\}_{b=1}^{B}$ (classification).

Suppose $(X,Y) \sim P_{X,Y}$ and define the margin

$$mg(X,Y) = \text{margin} = \text{Avg}_m\left[I(h_m(x) = y)\right] - \max_{k \neq Y} \text{Avg}\left(I(h_m(x) = k)\right)$$

Neural Networks



Three layer neural network.

Assumption information is passed $x \longrightarrow z \longrightarrow y$

In the second layer

$$Z_m = \sigma\left(\alpha_{0m} + \alpha_m^T x\right) \qquad m = 1, \dots, M$$

activation

first layer information

$$T_k = \beta_{0k} + \beta_k^T Z, \quad k = 1, \dots, K$$

$$Y_k = \underbrace{g_k}\left(T_k\right)$$

Chosen to match response Y_k.

e.g. $g(T) = T$ (regression)

$$g(T) = \frac{e^{T_k}}{\sum_{\ell=1}^{K} e^{T_\ell}} \quad \text{(classification).}$$

For a single Y

$$Y = g\left(\beta_0 + \sum_{i=1}^{M} \beta_m \, \sigma\left(\alpha_{0m} + \alpha_m^T x\right)\right)$$

<u>Training</u>: Uses Gradient descent to find local optimum

$$\left(\hat{\alpha}_{0m}, \hat{\alpha}_m, \hat{\beta}_{0k}, \hat{\beta}_k\right)_{m=1,\ k=1}^{M,\ K}$$

Called "Back Propagation"