Bagging, Adaboost, Random Forest. $\}$ → Bootstrap sampling.

Random Forest ≡ Bagged Tree ++

Adaboost ⟶ $\sum_{i=1}^{K} (\text{Tree}_i)\alpha$ ⟶ In Adaboost, the prob. of each sample gets affected by the perf. of the tree

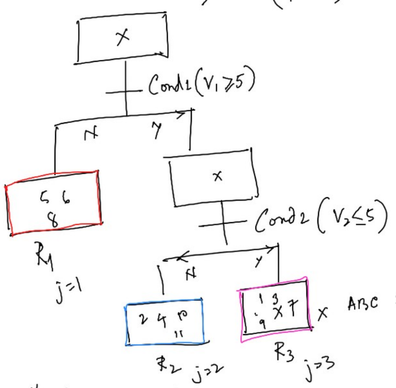
$$w_i = w_i^{(1)} \exp(-\alpha h_i y_i) \quad \left.\begin{matrix} h_i = y_i & \text{then } 1 \\ h_i \neq y_i & \text{" } -1 \end{matrix}\right\}.$$

GBM → Gradient Boosted Machine.

Xgboost, LightGBM, Catboost

$$\underbrace{\theta^{(n)} \longrightarrow \theta^{(0)} - \eta \nabla L(\theta)\Big|_{\theta=\theta^0}}$$ ⟶ Gradient descent Algo.

$X \to (\overset{v_1}{7}, \overset{v_2}{3})$



X ABC :

A : 70%
B : 60%
C : 75%

# of leaves = 3 (J) J = 1,2,3

$T(x;\theta) = \ldots\ldots$

$\theta := \{ R_j, \gamma_j \}$ → Disjoint regions.

For regression $\gamma_j = \bar{x}$

For classification γ_j = class prob.

$$f(x) = \underline{T}(x;\theta) = \sum_{j=1}^{J} \gamma_j I(x \in R_j)$$

→ # of leaves

$\gamma_1, \gamma_2, \gamma_3$

Model parameters $\underline{\theta} = \{ R_j, \gamma_j \}$.

Optimal parameter $\theta^* = \underset{\theta}{\arg\min} \sum_{j=1}^{J} \sum_{x_i \in R_j} L(y_i, \gamma_j)$

→ Continuous Loss func. of θ.

→ $\hat{f}(x_i)$

$$T(x;\theta) = \sum_{j=1}^{J} \gamma_j I(x \in R_j)$$

$$f(x_i) = \gamma_j$$

Additive model ⟵ $F = \sum_{m=1}^{M} h_m$ ⟶ Tree outcome.

$$\begin{matrix} T_1 & \to & T_2 & \to & T_3 & \to & T_4 & \to \cdots \to & T_M \\ \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ h_1 & + & h_2 & + & h_3 & + & h_4 & + & h_M. \end{matrix}$$

(a) 3 + 2 + 3 + ...

$10 \quad \underbrace{3 \; 2 \; 3}_{f_{m-1}(x_i)} \; \underset{m}{\boxed{\cdot}} \cdots$

$m = 4.$

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{j=1}^{J} \sum_{x_i \in R_j} L(y_i, \gamma_j)$$

$$\hat{\Theta}_m = \underset{\Theta_m}{\operatorname{argmin}} \sum_{i=1}^{N} L\Big(\overset{\text{act. value.}}{y_i}, \underbrace{F_{m-1}(x_i)}_{\text{prev. pred.}} + \underbrace{T(x_i; \Theta_m)}_{\text{current tree.}}\Big)$$

$y_i = 10.$

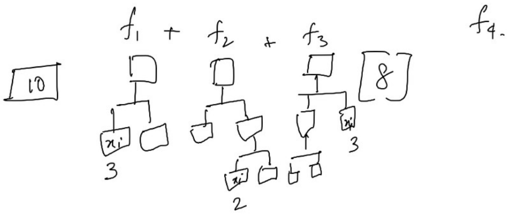
$f_3(x_i) = 3+2+3 = 8.$

$T(x_i; \Theta_m) = ?$

$y_i = 10.$

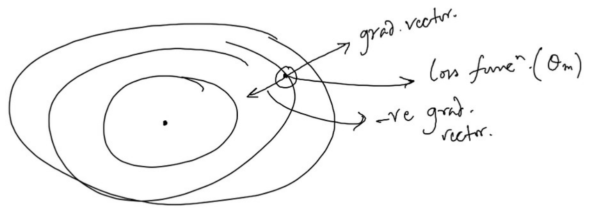
$f_1(x_i) = 3., \; f_2(x_i) = 2, \; f_3(x_i) = 3$

$F_3 = f_1 + f_2 + f_3 = 8$



How to build the next tree given that $m-1$ nos of trees are already produced?

Answer: Build the next tree along the -ve gradient of the loss func^n at that point.



Consider regression Model: $\qquad \Theta = \{R_j, \gamma_j\}$

$$L = mse = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_k(x_i)\right)^2 \qquad f_k \to \gamma_j$$

For individual data point, $L = \frac{1}{2}\left(y_i - f_k(x_i)\right)^2$

$$\frac{\partial L}{\partial f_k(x_i)} = -\underbrace{\left(y_i - f_k(x_i)\right)}_{\text{error.}}$$

For the new tree,

target variable = e_i values.

$$-\frac{\partial L}{\partial f_k(x_i)} = y_i - f_k(x_i) = e_i$$

For regression, build the next tree taking the residuals from the previous tree as the target variable.

For classification, the loss funcⁿ is multinomial deviance.

$$-\sum_{K=1}^{K} \underset{\text{class.}}{I(y_i = c_k)} \log p_k(x) \longrightarrow p_k(x) = \underbrace{\frac{e^{f_k(x)}}{\sum_{l=1}^{K} e^{f_l(x)}}}_{\text{softmax func}^n\text{.}}$$

A
B
A
B
B
A
C
A

GBM must have K nos of parallel trees, each predicting only one of the K classes.

$$\begin{array}{llll}
X \to C_k = C_1 & \longrightarrow f_{C_1}(x) & \to p_{C_1}(x) & \longrightarrow \dfrac{e^{f_{C_1}(x)}}{e^{f_{C_1}(x)} + e^{f_{C_2}(x)} + e^{f_{C_3}(x)} + e^{f_{C_4}(x)}} \\
X \to C_2 & \longrightarrow f_{C_2}(x) & \to p_{C_2}(x) & \\
X \to C_3 & \longrightarrow f_{C_3}(x) & \to p_{C_3}(x) & \\
X \to C_4 & \longrightarrow f_{C_4}(x) & \to p_{C_4}(x). &
\end{array} \qquad K = 4.$$

$$\sum_{i=1}^{4} p_{C_i}(x) = 1.$$

$$L = -\sum I(y_i = C_k)\log p_k(x)$$

$$= -\sum\left[I(y_i = C_k)\log e^{f_k(x)} - \log\left(\sum_{l=1}^{K} e^{f_l(x)}\right)\right]$$

$$L = -\sum I(y_i = C_k) f_k(x) + \sum \log\left(\sum_{l=1}^{K} e^{f_l(x)}\right) I(y_i = C_k)$$

$$\underline{L} = -\sum I(y_i = c_k) f_k(x) + \sum I(y_i = c_k) \log\left(\sum_{l=1}^{K} e^{f_l(x)}\right)$$

K trees — c_1, c_2, c_3 $f_3(x)$, c_4, c_5

$$-\frac{\partial L}{\partial f_k(x)} = I(y_i = c_k) - \frac{1}{\sum_{l=1}^{K} e^{f_l(x)}} \frac{\partial}{\partial \underline{f_k(x)}}\left[\sum e^{f_l(x)}\right]$$

except $e^{f_k(x)}$

$$\frac{\partial}{\partial f_k(x)} e^{f_l(x)} = 0.$$

$$= I(y_i = c_k) - \frac{e^{f_k(x)}}{\sum_{l=1}^{K} e^{f_l(x)}}$$

$$= \underline{\underline{I(y_i = c_k) - p_k(x)}}.$$

$x_i \xrightarrow{} y_i = c_2$

$x_i \rightarrow I(y_i = c_1) - p_1(x_i)$

$x_i \rightarrow I(y_i = c_2) - p_2(x_i)$

$x_i \rightarrow I(y_i = c_3) - p_3(x_i)$