GBDT Paper Summary

1. Problem Setting ⟶ Function Estimation

- $Y \to$ response variable
- $X \to$ explanatory variable

① → We hope to do: $F^* = \underset{F}{\arg\min}\ \mathbb{E}_{X,Y}[L(Y, F(X))]$

$$= \underset{F}{\arg\min}\ \mathbb{E}_X[\mathbb{E}_Y[L(Y, F(X)) \mid X]]$$

$\boxed{\text{Pf Sketch}}$ $\mathbb{E}_{X,Y}[L] = \int_x \int_y L(y, F(x))\, p(x,y)\, dx\, dy.$

$= \int_x p(x) \cdot \int_y L(y, F(x)) \cdot p(y|x)\, dy\, dx$

$= \int_x \mathbb{E}_Y[L \mid X=x]\, p(x)\, dx$

$= \mathbb{E}_X[\mathbb{E}_Y[L(Y, F(X)) \mid X]]$

$\boxed{\text{Rmks}}$: 1. MSE Optimality $\Rightarrow F(x) = \mathbb{E}[Y \mid X = x]$

↳ $\boxed{\text{if } L(y, F(x)) = (y - F(x))^2}$

⟷ Linear Regression (Wasserman Lec 26)

2. the previous result is unconstrained result

actually what we do is CONSTRAINED (under some specicific parametric model) i.e., tree model etc.

$\boxed{\text{intractable}}$!!!

→ we do not the exact distribution $(X, Y) \sim P$

→ it is also intractable to $\boxed{\text{estimate}}$ the conditional expectation $\mathbb{E}_Y[\cdot \mid X]$

2. Optimization Framework (Infinite Data) expectation case

a) Parametric Perspective

① Parametrized Model $F(x; P) = F(x, \{\beta_m, \alpha_m\}_{m=1}^{M})$

$$= \sum_{m=1}^{M} \beta_m h(x; \alpha_m)$$

② $\Phi(P) := \mathbb{E}_{(x,y)}[L(y, F(x; P))]$

$$P^* = \underset{P}{\arg\min}\ \Phi(P) \Rightarrow F^*(x) = F(x; P^*)$$

③ From the optimization perspective, we always have:

$$p^* = \sum_{m=0}^{M} p_m \Rightarrow \underline{\text{Gradient Descent Framework}} \text{ etc.}$$

optimization perspective

$$\boxed{g_m} = \nabla_p \Phi(P) \Big|_{p = p_{m-1}} \qquad P_{m-1} = \sum_{i=0}^{m-1} d_i$$

multi-dim

$$d_m = -\rho_m g_m \quad \text{and} \quad \rho_m = \arg\min_{\rho} \Phi(P_{m-1} - \rho g_m)$$

Rmk: Here, we try to characterize the function family through parametric form, i.e.,

$$\boxed{f(x) = \sum_{i=1}^{K} d_i \mathbb{1}\{x \in R_i\}} \rightarrow \underline{\text{Decision Tree Model}}$$

Then formulate the problem as optimization problem:

$$\rightarrow \hat{\omega} = \arg\min_{\omega \in \Omega} R_{pop}(\omega)$$

b) Non-parametric Perspective

①
$$\Phi(F) = \mathbb{E}_{(X,Y)}[L(Y, F(X))]$$
$$= \mathbb{E}_X[\mathbb{E}_Y[L(Y, F(X)) \mid X]]$$
$$:= \mathbb{E}_X[\phi(F(X))]$$
$$\phi(F(X)) = \mathbb{E}_Y[L(Y, F(X)) \mid X]$$

$$F^* = \arg\min \Phi(F) \Leftrightarrow F^*(x) = \arg\min \phi(F(x)) \quad \text{for each } x$$

$$\Leftrightarrow \quad F^*(x) = \arg\min \mathbb{E}_y [L(y, F(X)) \mid X = x]$$

② Note:

→ for the above problem, there are infinitely many parameters

→ But for data sets, there are only a finite number $\{ F(x_i) \}_{i=1}^{N}$ involved!

→ Discussed Below!

③ Infinitely many parameters Senario

→ $\boxed{\text{optimization}}$

Recap: $F^*(x) = \arg\min \mathbb{E}_y [L(y, F(x)) \mid X = x]$

$\quad = \arg\min \phi(F(x))$ for individual x

$\boxed{\text{GD Framework}}$ → intractable

$$\Longrightarrow F^*(x) = \sum_{m=0}^{M} f_m(x)$$

$$f_m(x) = -\rho_m g_m(x)$$

where $\boxed{g_m(x)} = \left. \frac{\partial \phi(F(x))}{\partial F(x)} \right|_{F(x) = F_{m-1}(x)}$ $\quad F_{m-1}(x) = \sum_{i=0}^{m-1} f_i(x)$

$\boxed{\text{1 - dim}}$

$$= \left. \frac{\partial \mathbb{E}_y [L(y, F(x)) \mid x]}{\partial F(x)} \right|_{F(x) = F_{m-1}(x)}$$

under regularity cond. $\quad \mathbb{E}_y \left[\frac{\partial L(y, F(x))}{\partial F(x)} \Big| x \right]_{F(x) = F_{m-1}(x)}$

$$\rho_m = \arg\min_{\rho} \mathbb{E}_{(x,y)} [L(y, F_{m-1}(x) - \rho g_m(x))]$$

3. Finite Data Case $\longrightarrow$ Tractable Senario

① Greedy strategy + empirical loss

$$\begin{cases} (\beta_m, \alpha_m) = \underset{\beta, \alpha}{\arg\min} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \beta h(x_i; \alpha)) \\ F_m(x) = F_{m-1}(x) + \beta_m h(x; \alpha_m) \end{cases}$$

when $\begin{cases} y_i \in \{-1, +1\}, \quad h(x; \alpha) \in \{-1, +1\} \\ L(y, F) = \exp\{-yF\} \end{cases}$

then this is exactly AdaBoost !

② Stage-wise Strategy to find (β_m, α_m)

$\boxed{\text{Motivation}}$ $\rightarrow$ in previous (non-parametric) discussion, we finally express $F^*(x)$ as

$$F^*(x) = \sum_{m=0}^{M} f_m(x)$$

$$= \sum_{m=0}^{M} -\rho_m g_m(x)$$

$\boxed{\text{Issue}}$ Here, we do not have complete expected gradient

$$g_m(x) = \frac{\partial(\phi(F(x)))}{\partial F(x)} \Bigg|_{F(x) = F_{m-1}(x)}$$

$$= \frac{\partial(E_y[L(y, F(x)) \mid x])}{\partial F(x)} \Bigg|_{F(x) = F_{m-1}(x)}$$

What we have is:

$$g_m(x_i) = \left. \frac{\partial(L(y, F(x)))}{\partial F(x)} \right|_{F(x) = F_{m-1}(x_i)}$$

$\boxed{\text{Solution}}$ $F_m(x) = F_{m-1}(x) + \beta_m \cdot h(x; \alpha_m)$

$$\begin{cases} 1. \ \alpha_m = \underset{\alpha, \beta}{\arg\min} \sum_{i=1}^{N} (-g_m(x_i) - \beta h(x_i; \alpha))^2 \\ \\ 2. \ \rho_m = \arg\min \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \rho h(x_i, \alpha_m)) \end{cases}$$

GBDT Algorithm

4. Application

Here $L(y, F) = (y - F)^2/2$. The pseudo–response in line 3 of Algorithm 1 is $\tilde{y}_i = y_i - F_{m-1}(\mathbf{x}_i)$. Thus, line 4 simply fits the current residuals and the line search (line 5) produces the result $\rho_m = \beta_m$, where β_m is the minimizing β of line 4. Therefore, gradient boosting on squared–error loss produces the usual *stagewise* approach of iteratively fitting the current residuals:

Algorithm 2: LS_Boost
$F_0(\mathbf{x}) = \bar{y}$
For $m = 1$ to M do:
  $\tilde{y}_i = y_i - F_{m-1}(\mathbf{x}_i), \quad i = 1, N$
  $(\rho_m, \mathbf{a}_m) = \arg\min_{\mathbf{a},\rho} \sum_{i=1}^{N} [\tilde{y}_i - \rho h(\mathbf{x}_i; \mathbf{a})]^2$
  $F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \rho_m h(\mathbf{x}; \mathbf{a}_m)$
endFor
end Algorithm

12.05

→ idea diagram (non-trivial)

GBDT → ideally, $F^* = \underset{F}{\operatorname{argmin}}\ \mathbb{E}_{(x,y)}[L(y, F(X))]$

↓

break down to → parametric :

$\Rightarrow p^* = \underset{P}{\operatorname{argmin}}\ \mathbb{E}_{(x,y)}[L(y, f(x; P))]$

→ ☆ Non-parametric : → our interest

$\Rightarrow F^* = \underset{F}{\operatorname{argmin}}\ \mathbb{E}_{(x,y)}[L(y, F(X))]$

$= \underset{F}{\operatorname{argmin}}\ \mathbb{E}_X[\mathbb{E}_y[L(y, F(X)) \mid X]]$

→ optimization framework

$\Rightarrow F^*(x) = \sum_{m=0}^{M} f_m(x)$

$= F_{M-1}(x) - \rho_m g_m(x)$

$\longrightarrow g_m(x) = \left.\dfrac{\partial(\mathbb{E}_y[L(y, F(x)) \mid x])}{\partial F(x)}\right|_{F(x) = F_{m-1}(x)}$

↓

Still infeasible

⇓

Surrogate : Finite Data Case

Motivation : we want to approximate $g_m(x) = \left.\dfrac{\partial \phi(F(x))}{\partial F(x)}\right|_{F(x) = F_{m-1}(x)}$

Solution : what we have $\Rightarrow \{g_m(x_i)\}_{i=1}^{N} \rightsquigarrow \boxed{N\text{- discrete point}}$

$$g_m(x_i) = \frac{\partial(L(y_i, F(x_i))}{\partial F(x_i)} \Bigg|_{F(x) = F_{m-1}(x)}$$

$$\Rightarrow \begin{cases} ① \text{ train } \boxed{h(x; \alpha_m) \text{ to fit } \{g_m(x_i)\}_{i=1}^{N}} \\ ② \text{ then } \boxed{h(x; \alpha_m) \approx g_m(x)} \rightarrow \text{our aim} \end{cases}$$

Last Step : select optimal step length ρ_m through Loss Func^n :

$$\rho_m = \underset{\rho}{\arg\min} \sum_{n=1}^{N} L(y_n, F_{m-1}(x_n) + \rho h(x_n; \alpha_m))$$

XGBoost (Boosting Algorithm)

↓

An approximation on Empirical Loss (Taylor)

Tree Structure

Consider the model defined as follows :

$$\rightarrow F_t(x) = \sum_{l=1}^{t} f_l(x)$$

$$= F_{t-1}(x) + f_t(x)$$

$$\Rightarrow ① \; \hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$$

$$② \; g_i := \nabla_{\hat{y}_i^{(t-1)}} L(y_i, \hat{y}_i^{(t-1)})$$

$$③ \; h_i := \nabla^2_{\hat{y}_i^{(t-1)}} L(y_i, \hat{y}_i^{(t-1)})$$

Then, $\boxed{R_{emp}^{(t)}} = \sum_{i=1}^{n} L(y_i, \hat{y}_i^{(t)}) + \Omega(F_t)$

$$= \sum_{i=1}^{n} L(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(F_t)$$

the empirical loss for first-t base learner

$$\approx \sum_{i=1}^{n} \left[L(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t)$$

$F_t(x) = \sum_{\tau=1}^{t} f_\tau(x)$

$$\propto \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t)$$

$$= \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \gamma \cdot T + \frac{1}{2} \lambda \sum_{t=1}^{T} w_t^2$$

$$= \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) \cdot w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] + \gamma T$$

Assume we know the structure of the t-th base learner

$\Updownarrow$

i.e., $\{I_1, I_2, \ldots, I_T\} \longrightarrow$ the partition of $[n]$

Conclusion: for a fixed structure $\boxed{\{I_1, I_2, \ldots, I_T\}} \Leftrightarrow q(x)$

the optimal weight $$w_j^* = - \frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

Note: $I_j := \{ i : q(x_i) = j \}$

then, $$R_{emp}^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T$$ for a given structure $q(\cdot)$

☆

Rmk: Interpretation $\longrightarrow$ $R_{emp}^{(t)}(q)$ can be viewed as a score to measure the quality of tree structure $q(\cdot)$ [like Gini impurity & entropy]

Combining with Recursive Binary Search (like DT)