Introduction to XGBoost

March 13, 2019

Outline

Objective Functions

Definition (Objective Functions)

$\theta \in \Theta$ - the parameter space.

$$Obj(\theta) = L(\theta) + \Omega(\theta)$$

L: **Loss function** measures how well model fit on training data
Ω: **Regularization** measures complexity of model

Example (Objective Function for Ridge Regression)

$\beta \in \mathbb{R}^n$. λ is a constant number.

$$Obj(\beta) = \|y - X\beta\|_2^2 + \lambda\|\beta\|_2^2$$

Outline

Regression Trees (CARTs)

How do we use **Classification And Regression Tree (CART)** to regress or classify?

A Brief Introduction

Example (Regression Tree)

Every regression tree T has the following form:

$$\hat{y}_i = \sum_{m=1}^{M} \omega_m 1_{x \in R_m}.$$

Two Elements completely determine a regression tree:

1. Partition of Sample Space $\{R_1, \ldots, R_M\}$,
2. and Weights Vector $\omega = (\omega_1, \ldots, \omega_M)$.

A Brief Introduction

Example (Regression Tree: Revisit)

Given a partition of sample space $\{R_1, \dots, R_M\}$.
For every sample point x, there must exist $n \in \{1, 2, \dots, M\}$ such that $x \in R_n$. Define

$$q : x \mapsto n,$$

which describes the partition $\{R_1, \dots, R_M\}$.
$\Longrightarrow$ re-write regression tree as

$$T = \omega_q$$

Outline

Generate Regression Tree

Problem: Given N observations (x_i, y_i), $i = 1, 2, \ldots, N$. We want to build a regression tree such that

$$L = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

is minimized.

Solution: Notice the following **fact**:

Given q, there is unique $\hat{\omega}$ such that L is minimized.

$$\begin{aligned} \hat{\omega}_m &= \frac{1}{|R_m|} \sum_{x_i \in R_m} y_i \\ &:= \operatorname{avg} R_m \quad m = 1, \ldots, M. \end{aligned}$$

Generate Regression Tree

Because of the fact, our problem become:
How to find the best tree structure q?
Answer: We generate the tree from top to bottom.

Example (Generate a regression tree)

Define $R_{j,s} = \{X \mid X_j \leq s\}$.
Solve the following optimization problem:

$$\min_{j,s} \quad \left(\sum_{x_i \in R_{j,s}} (y_i - \mathrm{avg} R_{j,s})^2 + \sum_{x_i \notin R_{j,s}} (y_i - \mathrm{avg} R_{j,s}^c)^2 \right)$$

How to solve it: s is choosen from $\{x_{ij}\}_{i=1,\ldots,N}$. Enumerate every pair (j, s).
Then we will add a split at each subnode of the first node. But **when should we stop adding a new split?**

Outline

Pre-Stopping and Pruning

Pre-Stopping: We can stop adding a new spit when ...

- *max_depth*
- *min_child_weight*
- *gamma*
- ...

Pruning

Suppose we have a partition into M regions $R_1, \ldots, R_M$. Tree T is built on the partition.

Definition (Loss funtion)

Let $N_m = |R_m|$, $\hat{\omega}_m = \mathrm{avg} R_m$, and $Q_m(T) = \frac{1}{|R_m|} \sum_{x_i \in R_m} (y_i - \hat{\omega}_m)^2$.

$$L(T) := \sum_{m=1}^{|T|} N_m Q_m(T).$$

Definition (Complexity of our model)

Given a constant α. Define

$$\Omega(T) = \alpha |T|.$$

Pruning

Problem: We have built a large tree T_0. Find a subtree T_α such that

$$C_\alpha(T) = L(T) + \Omega(T) = \sum_{m=1}^{|T|} N_m Q_m(T) + \alpha|T|$$

is minimized.
Solution: Let $\mathfrak{N} \in T^\circ := \{\text{All internal nodes of } T\}$. The operator

$$\sigma_{\mathfrak{N}} : T \mapsto \sigma_{\mathfrak{N}} T$$

collapses the node $\mathfrak{N}$ to get the subtree $\sigma_{\mathfrak{N}} T$.

Pruning

Solving the following optimization problem

$$\min_{\mathfrak{N} \in T^\circ} \quad (L(T) - L(\sigma_{\mathfrak{N}} T))$$

we get $\tilde{\mathfrak{N}}$.
Now let's begin from T_0. We successively collapse node and finally get a sequence of subtrees

$$\{\sigma_{\tilde{\mathfrak{N}}_0} T_0, \sigma_{\tilde{\mathfrak{N}}_1} \sigma_{\tilde{\mathfrak{N}}_0} T_0, \ldots, \{\mathfrak{R}\}\},$$

where $\mathfrak{R}$ is the root of T_0.
Find T_α such that the value of $(L(T) - L(\sigma_{\mathfrak{N}} T))$ is the smallest in this set.

Outline

Tree Ensembles

Definition (Tree Ensemble Model)

Given a fixed sample point $x_i = (x_{i1}, \ldots, x_{in})$, our model will predict:

$$\hat{y}_i = \sum_{k=1}^{t} f_k(x_i),$$

where f_k is a CART, for each k.

Question: How to find parameters $\{f_1, \ldots, f_t\}$ of this model?
Answer: Define an objective function, and optimize it!.

Outline

Tree Boosting

Aim: Find all parameters $\{f_1, \ldots, f_S\}$ of our model

$$\hat{y} = \sum_{k=1}^{S} f_k(x).$$

Method: We'll find f_k one by one. Assume we have a part of our model:

$$\hat{y}_i^{(t-1)} = \sum_{j=1}^{t-1} f_j(x_i).$$

We want to find the t-th parameter in the space of **all CARTs**

$$\Theta = \{f_1^{(t)}, \ldots, f_K^{(t)}, \ldots\}.$$

We construct the following optimization problem to find f_t:

$$\min_{f_t \in \Theta} \quad Obj^{(t)}(f_t) = L^{(t)}(f_t) + \Omega(f_t)$$

Outline

$L^{(t)}$ - Loss Function

Re-Write Loss Function $L^{(t)}$:

$$L^{(t)}(f_t) = \sum_{j=1}^{N} l(y_j, \hat{y}_j^{(t)})$$
$$= \sum_{j=1}^{N} l(y_j, \hat{y}_j^{(t-1)} + f_t(x_j)).$$

Definition (Objective Function $Obj^{(t)}$)

$$Obj^{(t)}(f_t) = \sum_{j=1}^{N} l(y_j, \hat{y}_j^{(t-1)} + f_t(x_j)) + \Omega(f_t) + \sum_{k=1}^{t-1} \Omega(f_k)$$

Note: It is hard to minimize it. We need a simlified version of loss.

Taylor Expansion Approximation of Loss

Notice that for the j-th sample point

$$l(y_j, \hat{y}_j^{(t-1)} + f_t(x_j)) \approx l(y_j, \hat{y}_j^{(t-1)}) + g_j f_t(x_j) + \frac{1}{2} h_j f_t^2(x_j),$$

where $g_j = \partial_{\hat{y}^{(t-1)}} l(y_j, \hat{y}^{(t-1)})$ and $h_j = \partial^2_{\hat{y}^{(t-1)}} l(y_j, \hat{y}^{(t-1)})$.

Definition (Loss Function)

$$L^{(t)}(f_t) = \underline{\sum_{j=1}^{N}(l(y_j, \hat{y}_j^{(t-1)}))} + \sum_{j=1}^{N}(g_j f_t(x_j) + \frac{1}{2} h_j f_t^2(x_j))$$

Note: $\underline{\sum_{j=1}^{N}(l(y_j, \hat{y}_j^{(t-1)}))}$ doesn't depend on f_t.

Ω - Complexity of CART

We re-write a given CART T as follow

$$T(x) = \omega_{q(x)}.$$

Note: Every CART T is determined by two element:

- q - the structure of T
- ω - the weights of each leaf of T

Definition

Given constants γ and λ, we define

$$\Omega(T) = \gamma|T| + \frac{1}{2}\lambda\sum_{j=1}^{|T|} w_j^2$$

Revisit: Objective Function

Definition (Final Objective Function)

Set ...

$$I_j = \{\}$$

$$Obj^{(t)}(f_t) \approx \sum_{j=1}^{N}(g_j f_t(x_j) + \frac{1}{2}h_j f_t^2(x_j)) + \gamma|T| + \frac{1}{2}\lambda\sum_{j=1}^{|T|} w_j^2$$

$$= \sum_{j=1}^{|f_t|}[\underline{(\sum_{i\in I_j} g_i)\omega_j + \frac{1}{2}(\sum_{i\in I_j} h_i + \lambda)\omega_j^2}] + \gamma|f_t|$$

$$:= \sum_{j=1}^{|f_t|}[\underline{G_j\omega_j + \frac{1}{2}(H_j + \gamma)\omega_j^2}] + \gamma|f_t|$$

Revisit: Objective Function

Definition (Final Objective Function)

$$Obj^{(t)}(f_t) = \sum_{j=1}^{|f_t|}[G_j\omega_j + \frac{1}{2}(H_j + \gamma)\omega_j^2] + \gamma|f_t|$$

Note: Given a tree structure q, we can immediately get a CART minimizing $Obj^{(t)}$, by setting $\omega_j = -\frac{G_j}{H_j+\lambda}$.
And with the same q,

$$Obj^{(t)}(f_t) = -\frac{1}{2}\sum_{j=1}^{|f_t|}\frac{G_j^2}{H_j + \lambda} + \gamma|f_t|.$$

Outline

Tree Boosting

Reminder of Our Aim: Find all parameters $\{f_1, \ldots, f_S\}$ of our model

$$\hat{y} = \sum_{k=1}^{S} f_k(x).$$

Method: We construct the following optimization problem to find t-th parameter f_t:

$$\min_{f_t \in \Theta} \quad Obj^{(t)}(f_t) = \sum_{j=1}^{|f_t|} [G_j \omega_j + \frac{1}{2}(H_j + \gamma)\omega_j^2] + \gamma |f_t|.$$

Generate f_t

Similar to generate a CART, it suffices to find **the best tree structure** q.
Note: To get q_{new}, we add two nodes $\mathfrak{N}_L$ and $\mathfrak{N}_R$ at $\mathfrak{N}$ of q_{old}.
Define

$$\begin{aligned}\mathrm{Gain} &= Obj^{(t)}(q_{\text{old}}) - Obj^{(t)}(q_{\text{new}}) \\ &= \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma\end{aligned}$$

We add a new split with the highest value of Gain.

Pre-Stopping and Post-Prunning

Pre-Stopping We can stop adding a new split when

- Gain is nagetive.
- Reach the maximum depth.
- ...

Post-Prunning

Reference

- **CARTs part refers to this classical textbook.** *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* 2nd Edition. Chapter 9.
- **XGBoost refers to Tianqi Chen's slide,** *Introduction to Boosted Trees.* `https://homes.cs.washington.edu/~tqchen/pdf/BoostedTree.pdf`
- **and to this document,** `http://xgboost.readthedocs.io/en/latest/model.html`