Inductive Gradient Boosting

Charles A. Pehlivanian

Abstract

We propose a bottom-up, inductive approach to tree selection within the popular Gradient Boosting algorithm and its derivatives (LightFBM, XGBoost, etc.) based on exact solution of the quantification problem at the iterative step. An inductive classifier is then used to approximate the exact solution, leading to classifiers that compare favorably out of sample to the classic methods.

1 Preliminaries

Let $n \in \mathbb{N}$ be positive and set $\mathcal{V} = \{1, \ldots, n\}$. Let $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_n\}$ be finite real sequences with $y_i > 0$, for all i. It is assumed that X, Y are orderd to satisfy $\frac{x_1}{y_1} \leq \frac{x_2}{y_2} \leq \cdots \leq \frac{x_n}{y_n}$. In many applications, the tuple (x_i, y_i) corresponds to occurence and baseline attributes for i, possibly associated with a spatial location. Denote by $\mathcal{D} = \mathcal{D}_{X,Y}$ the set of tuples $\{(x_i, y_i)\}$ associated with X, Y. $\mathcal{D}$ is assumed to have an order induced by a priority function on the sequences X, Y. A partition $\mathcal{P} = \{S_1, \ldots, S_t\}$ of size t of $\mathcal{V}$ can be identified with a pointset in $\mathbf{R}^2$ by associating $S \subseteq \mathcal{V}$ with the point $(\sum_{i \in S} x_i, \sum_{i \in S} y_i)$, called the *partition point* of S. Set $p_\emptyset = (0, 0)$. In this way the sequences X, Y induce an embedding of $\mathcal{P}$ into $\mathbf{R}^2$. The notion of score function comes from the spatial scan statistics literature.

Definition 1. *A score function is a continuous $f(x, y) \colon \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}$, nondecreasing in x, with continuous extension to the origin in any wedge $\mathcal{W}(\mu_1, \mu_2) = \left\{(x, y) : y > 0, \mu_1 \leq \frac{x}{y} \leq \mu_2\right\}$, for $-\infty < \mu_1 \leq \mu_2 < \infty$, with the extension satisfying $f(0, 0) = 0$. If f is of the form $f(x, y) = x^\alpha y^{-\beta}$ for some $\alpha, \beta > 0$, then f is a rational score function.*

We will only consider rational score functions in this paper. The regularity condition on $\mathcal{W}$ in wedges simply guarantees a continuous extension to the origin on any positive cone in $\mathbf{R}^+$, for rational score functions the constraint corresponds to the constraint $\alpha > \beta$. We do not assume smoothness beyond continuity, nor (quasi)convexity, etc., unless explicitly stated. The function f induces a real-valued set function on $2^\mathcal{V}$ by defining $F(S) = f(\sum_{i \in S} x_i, \sum_{i \in S} y_i)$, for $S \subseteq \mathcal{V}$.

Definition 2. *A priority function is a function $g \colon \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}$ that induces an ordering on the dataset $\mathcal{D}$. We refer to $g(x, y) = \frac{x}{y}$ as the standard priority function.*

Unless otherwise stated, the sets X, Y will be assumed to be already indexed in standard priority order, i.e., $g(x_1, y_1) \leq \cdots \leq g(x_n, y_n)$, where g is the standard priority function. We only consider the standard priority function in this paper.

For a given f, We are interested in maximal partitions for F, i.e., solutions to the program

$$\mathcal{P}^* = \underset{\mathcal{P}=\{S_1 \ldots, S_t\}}{\operatorname{argmax}} \sum_{j=1}^{t} F(S_j) = \underset{\mathcal{P}=\{S_1, \ldots, S_t\}}{\operatorname{argmax}} \sum_{j=1}^{t} f(\sum_{i \in S_j} x_i, \sum_{i \in S_j} y_i) \tag{1}$$

which for a rational score function is simply

$$\mathcal{P}^* = \underset{\mathcal{P}=\{\pi_1 \ldots, \pi_T\}}{\operatorname{argmax}} \sum_{j=1}^{t} \frac{(\sum_{i \in P_j} x_i)^\alpha}{(\sum_{i \in P_j} y_i)^\beta} \tag{2}$$

Definition 3. *A consecutive subset of $\mathcal{V}$ is a subset of the form $\{j, j+1, \ldots, k\}$ for some $1 \leq j \leq k \leq n$. A consecutive partition $\mathcal{P} = \{S_1, \ldots, S_t\}$ is a partition of $\mathcal{V}$ such that each S_i is a consecutive subset.*

Letting $\mathcal{P}_c$ be the set of consecutive partitions of $\mathcal{V}$, it is easy to see that $|\mathcal{P}_c| = \frac{n(n+1)}{2} + 1$.

Definition 4. *The score function F satisfies the Consecutive Partitions Property (CPP) if the solution*

$$\mathcal{P}^* = \underset{\substack{\mathcal{P} \\ |\mathcal{P}|=t}}{\operatorname{argmax}} \sum_{j=1}^{T} F(\sum_{i \in P_j} x_i, \sum_{i \in P_j} y_i)$$

is a consecutive partition, for all X, Y. F satisfies the Weak Consecutive Partitions Property (WCPP) if the solution

$$\mathcal{P}^* = \underset{\substack{\mathcal{P} \\ |\mathcal{P}|\leq t}}{\operatorname{argmax}} \sum_{j=1}^{T} F(\sum_{i \in P_j} x_i, \sum_{i \in P_j} y_i)$$

is a consecutive partition, for any X, Y. F satisfies CPP($\mathbf{R}^+$), WCPP($\mathbf{R}^+$) if it satisfies CPP, WCPP, respectively, for $X \subseteq \mathbf{R}^+$, Y.

It was shown in [2] that if f satisfies some simple properties, that the solution to 1 is realized at a consecutive partition, namely

Theorem 1. *Let $f(x, y) = x^{\alpha} y^{-\beta}$ be a rational score function with $\alpha > \beta > 0$. Then f satisfies CPP if and only if $\alpha - \beta = 1$ and α is even. f satisfies WCPP if and only if $\alpha - \beta \geq 1$ and α is even.*

In addition a dynamic programming approach provides an order $\mathcal{O}(n^2 t)$ solution to 2, with storage requirements proportional to nt.

1.1 Applications to gradient boosting

Given a set of quadratic polynomials $p_i(x) = \frac{1}{2} h_i x^2 + g_i x + c_i$, with $h_i > 0$ for all i, the minimum values attained are $\frac{-g_i^2}{2h_i}$. Assume that the polynomials are ordered by the x-coordinate of the vertex of the minimum value, $\frac{-g_i}{h_i}$, and define the aggregate $p_{S_j}(x) = \frac{1}{2} \sum_{i \in S_j} h_i x^2 + \sum_{i \in S_j} g_i x + \sum_{i \in S_j} c_i$, for $S_j \subseteq \mathcal{V}$. Since p_{S_j} is minimized at $(\frac{-\sum_{i \in S_j} g_i}{\sum i \in S_j h_i}, -\frac{1}{2} \frac{(\sum i \in S_j g_i)^2}{\sum i \in S_j h_i} + \sum_{i \in S_j} c_i)$, the solution to 1 for $\alpha = 2$, $\beta = 1$ finds the x-values that minimize the sum of any set of aggregate polynomials of size t. This minimization is at the heart of the iterative step of the well-known gradient boosting algorithm, see, e.g. [3], or the recent implementations XGBoost ([4]), LightGBM ([5]). An additive update $f_t(x_i)$ is sought for the step $(t-1)$ classifier which minimizes the loss

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \tag{3}$$

where $l(y_i, \hat{y}_i)$ is an arbitrary convex loss function, and $\Omega(f_t)$ is an l^2-regularization term. In the classical gradient boosting approach, the loss in Equation 3 is approximated by a quadratic polynomial, and a decision tree classifier is chosen as f_t. Denoting by $\{(S_j, w_j)\}_{j=1}^{t}$ the leaf sets (sets of constant leaf value for f_t) and w_j the leaf value, equation 3 becomes

$$\mathcal{L}^{(t)} = \sum_{j=1}^{t} [(\sum_{i \in S_j} \beta_i) w_j + \frac{1}{2} (\sum_{i \in S_j} \alpha_i + \lambda) w_j^2] + \gamma t \tag{4}$$

where $\beta_i = \frac{\partial l(y_i, \hat{y}^{(t-1)})}{\partial \hat{y}^{(t-1)}}$, $\alpha_i = \frac{\partial^2 l(y_i, \hat{y}^{(t-1)})}{\partial^2 \hat{y}^{(t-1)}}$. Obtaining the optimal leaf sets and values is equivalent to solving

$$\mathcal{L}^{(t)} = \max_{\mathcal{P} = \{S_1 \ldots, S_t\}} \sum_{j=1}^{t} \frac{(\sum_{i \in S_j} \beta_i)^2}{\sum_{i \in S_j} \alpha_i + \lambda} + \gamma t \tag{5}$$

and setting $w_j = -\frac{\sum_{i \in S_j} \alpha_i}{\sum_{i \in S_j} \beta_i + \lambda}$ (for details see [4]). By our Theorem 1, this optimization can be solved with time requirement no more than $\mathcal{O}(n^2 t)$.

We take this approach to the split-finding problem at each step. We specify t and solve Equation 5 exactly at each step, obtaining an exact specification of leaf values. This specification is not enough, a split rule is needed in order to classify out of sample data. We take a "bottom-up" approach to splitting, fitting an inductive classifier on the obtained leaf values. Thus the main difference between our approach and the classical one is that we seek an inductive classifier, rather than solving for a transductive classifier. Note that the approach still requires specification of an impurity measure.

1.2 Empirical results

[Forthcoming]

References

[1] Francis Bach, *Learning with Submodular Functions: A Convex Optimization Perspective*, Carnegie Mellon University, Pittsburgh, USA, 2011

[2] Charles A. Pehlivanian, Daniel B. Neill, *Efficient Optimization of Partition Scan Statistics via the Consecutive Partitions Property*, preprint, 2021

[3] J.Friedman, *Greedy function approximation: a gradient boosting machine*, Annals of Statistics, 29(5):1189–1232, 2001.

[4] Chen, Tianqi and Guestrin, Carlos, *XGBoost*, Proceedings of the 22nd ACM SIGKDD International Confernce of Knowledge Discovery and Data Mining, August, 2016.

[5] LightGBM documentation, https://lightgbm.readthedocs.io/en/latest/