GRADIENT BOOSTING: OVERVIEW, THEORY AND APPLICATIONS TO BIG DATA ANALYTICS

Tze Leung Lai

Stanford University

August 2017

From LMS to batch gradient learning algorithms

- Widrow and Hoff (1960) introduced the LMS (least mean squares) algorithm for fast on-line updating of the minimum-variance filter, which is basically a gradient descent method. For the stochastic regression model $y_t = \boldsymbol{\theta}^T \mathbf{x}_t + \epsilon_t$ (with $\mathcal{F}_{t-1}$-measurable $\mathbf{x}_t$ and i.i.d. errors ϵ_t), the least squares estimate $\hat{\boldsymbol{\theta}}_t$ can be expressed via the Kalman recursions

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + (y_t - \boldsymbol{\theta}_{t-1}^\top \mathbf{x}_t)\boldsymbol{\Gamma}_t \mathbf{x}_t, \boldsymbol{\Gamma}_t = \boldsymbol{\Gamma}_{t-1} - \frac{\boldsymbol{\Gamma}_{t-1}\mathbf{x}_t\mathbf{x}_t^\top \boldsymbol{\Gamma}_{t-1}}{1 + \mathbf{x}_t^\top \boldsymbol{\Gamma}_{t-1}\mathbf{x}_t}.$$

  The LMS recursions are $\hat{\boldsymbol{\theta}}_t = \hat{\boldsymbol{\theta}}_{t-1} + (y_t - \hat{\boldsymbol{\theta}}_{t-1}^\top \mathbf{x}_t)\gamma_t \mathbf{x}_t$, using a scalar (rather than matrix) gain sequence γ_t.
- Choice of gradient descent rate:

$$\gamma_t = c \text{ or } \gamma_t^{-1} = \sum_{i=1}^t \|\mathbf{x}_i\|^2 (\boldsymbol{\Gamma}_t^{-1} = \sum_{i=1}^t \mathbf{x}_i \mathbf{x}_i^\top).$$

- Batch learning for faster and more stable descent: Updates occur only after batches of $(\mathbf{x}_i, y_i), t_j \le i < t_{j+1}$, are collected. This is particularly relevant for high-dimensional $\boldsymbol{\theta}$.

Boosting (1997 to 2002): AdaBoost and Gradient Boosting

- Freund and Schapire (1997) introduced the AdaBoost algorithm, which suitably combines “weak” learners from training data into much better performing predictors or classifiers. This inspired
  - Hastie and Tibshirani (2000): additive logistic regression approach
- Friedman (2001) proposed gradient boosting approach to function estimation, which is motivated by applications to trees and other additive models in regression and classification.

High-dimensional nonlinear regression for big data

- Consider a regression function f that has an additive expansion:

$$f(\mathbf{x}) = \alpha + \sum_{k=1}^{p} \beta_k \phi_k(\mathbf{x}; \mathbf{b}_k), \tag{1}$$

  where ϕ_k is a basis function that involves a nonlinear parameter vector $\mathbf{b}_k \in \Gamma$ and is linearly associated with a regression coefficient β_k.

- The observed data are $\{\mathbf{x}_t, y_t\}_{t=1}^{n}$ and a loss function $L(y_t, f(\mathbf{x}_t))$ is used to fit the model; high-dimensional if $p > n$. Choice of L:
  - squared error loss: $L(y, f) = (y - f)^2/2$
  - the absolute error loss: $L(y, f) = |y - f|$
  - logistic regression with $L(y, f) = \log(1 + e^{-2yf})$ for $y \in \{-1, 1\}$
  - Huber's loss for M-estimators

$$L(y, f) = \begin{cases} \frac{1}{2}(y - f)^2, & |y - f| < c \\ c|y - f| - \frac{1}{2}c^2, & |y - f| \geq c. \end{cases}$$

Friedman's gradient boosting algorithm

- The algorithm represents a generalization that connects stagewise additive expansions to steepest-descent minimization, and can be described as follows:

```
1. $\hat{f}^0(\mathbf{x}) = 0$
2. For $k = 1, \ldots, m$ do
    (a) $\hat{u}_t^{k-1} = -\frac{\partial L}{\partial f}(Y_t, \hat{f}^{k-1}(\mathbf{x}_t)), t = 1, \cdots, n$
    (b) $\hat{\mathbf{b}}_k = \arg\min_{\mathbf{b} \in \Gamma, \beta \in \mathbb{R}} \sum_{t=1}^n [\hat{u}_t^{k-1} - \beta \phi(\mathbf{x}_t; \mathbf{b})]^2$
    (c) $\hat{\beta}_k = \arg\min_\beta \sum_{t=1}^n L(Y_t, \hat{f}^{k-1}(\mathbf{x}_t) + \beta \phi(\mathbf{x}_t; \hat{\mathbf{b}}_k))$
    (d) $\hat{f}^k(\mathbf{x}) = \hat{f}^{k-1}(\mathbf{x}) + \hat{\beta}_k \phi(\mathbf{x}; \hat{\mathbf{b}}_k)$
   End for
3. Output $\hat{f}^m(\cdot)$
```

- A shrinkage factor (called the "learning rate") ν with $0 < \nu < 1$ can be introduced to change step 2(d) to

$$\hat{f}^k(\mathbf{x}) = \hat{f}^{k-1}(\mathbf{x}) + \nu \hat{\beta}_k \phi(\mathbf{x}; \hat{\mathbf{b}}_k). \tag{2}$$

Extensions that include an intercept term

- Assume the more general model (1) that has an intercept term α and uses ϕ_k instead of ϕ. For (1) and more general loss functions L, we first center $\mathbf{x}_t$ by $\mathbf{x}_t - \bar{\mathbf{x}}$, where $\bar{\mathbf{x}} = n^{-1} \sum_{t=1}^{n} \mathbf{x}_t$, and then modify steps 2(b),(c),(d) to:

$$(\hat{j}_k, \hat{\mathbf{b}}_k) = \arg \min_{1 \leqslant j \leqslant p, \mathbf{b} \in \Gamma, \beta \in \mathbb{R}} \sum_{t=1}^{n} [\hat{u}_t^{k-1} - \hat{\alpha}_{k-1} - \beta \phi_j(\mathbf{x}_t; \mathbf{b})]^2,$$

$$(\hat{\alpha}_k, \hat{\beta}_k) = \arg \min_{\alpha, \beta \in \mathbb{R}} n^{-1} \sum_{t=1}^{n} L(Y_t, \hat{f}^{k-1}(\mathbf{x}_t) + \alpha + \beta \phi_{\hat{j}_k}(\mathbf{x}_t; \hat{\mathbf{b}}_k)),$$

$$\hat{f}^k(\mathbf{x}) = \hat{f}^{k-1}(\mathbf{x}) + \hat{\alpha}_k + \hat{\beta}_k \phi_{\hat{j}_k}(\mathbf{x}; \hat{\mathbf{b}}_k).$$

- For the case $L(y, f) = (y - f)^2/2$, this algorithm reduces to the pure greedy algorithm (PGA, Temlyakov 2000), which is called “matching pursuit” by Mallet and Zhang (1993) for the special case of “time-frequency dictionaries”, and also called L_2-boosting by Bühlmann and Yu (2003).

Convergence results for PGA in linear regression

- Bühlmann and Yu (2003) study the bias-variance trade-off as the number m of iterations increases when $(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)$ are i.i.d.
- Zhang and Yu (2005) prove the convergence of an appropriately terminated version under certain sparsity and moment conditions.
- Bühlmann (2006) refines the argument further for the case $p = \exp(O(n^\xi))$ with $0 < \xi < 1$ and shows that for linear regression the conditional mean squared prediction error

$$\mathrm{CPE} = E\{(f(\mathbf{x}) - \hat{f}^m(\mathbf{x}))^2 | y_1, \mathbf{x}_1, \cdots, y_n, \mathbf{x}_n\} \tag{3}$$

  (in which $\mathbf{x}$ is independent of $(\mathbf{x}_t, y_t)$ and has the same distribution as $\mathbf{x}_t$) converges in probability to 0 if $m = m_n \to \infty$ sufficiently slowly, but does not provide results on how slowly m_n should grow.
- It is widely recognized that early termination can avoid overfitting and some variable selection schemes such as AIC have been proposed to choose m_n, but a definitive theory on how m_n should be chosen is lacking.

Insights from OGA (orthogonal greedy algorithm)

- There is a definitive convergence theory for the orthogonal greedy algorithm (OGA), also called "orthogonal matching pursuit". It provides important insights to address the aforementioned convergence issues of PGA.
- A major difference between OGA and PGA is that at each iteration OGA selects a new input variable whereas PGA can select the same input variable in multiple iterations. For OGA, Ing and Lai (2011) show that optimal bias-variance tradeoff in high-dimensional sparse linear models entails that m_n should be $O((n/\log p_n)^{1/2})$, assuming that $\log p_n = o(n)$. Central to the derivation of the optimal convergence rate result is a *semi-population version* of OGA that uses the same forward stepwise variable selection procedure but assumes the corresponding regression parameters to be known.
- Inspired by OGA, we have developed a modified gradient boosting algorithm to address convergence issues not only for PGA but also for gradient boosting with general L in the general model (1).

Modified gradient boosting (MGB)

```
1. $\hat{f}^0(\mathbf{x}) = 0$, $\hat{J}_0 = \emptyset$, $\hat{\alpha}_0 = 0$
2. For $k = 1, \ldots, m$ do
    (a) $\hat{u}_t^{k-1} = -\frac{\partial L}{\partial f}(Y_t, \hat{f}^{k-1}(\mathbf{x}_t))$
    (b) Choose $\hat{\mathbf{b}}_k \in \Gamma$ and $\hat{j}_k$ such that $\hat{j}_k = \arg\min_j \#_j^{k-1}$ with
        $$(n^{-1}\sum_{t=1}^{n}\phi_j^2(\mathbf{x}_t, \hat{\mathbf{b}}_k))^{-1/2}|n^{-1}\sum_{t=1}^{n}(\hat{u}_t^{k-1} - \hat{\alpha}_{k-1})\phi_j(\mathbf{x}_t, \hat{\mathbf{b}}_k)|$$
        $$\geq \epsilon \max_{\tilde{j}, \mathbf{b} \in \Gamma}(n^{-1}\sum_{t=1}^{n}\phi_{\tilde{j}}^2(\mathbf{x}_t, \mathbf{b}))^{-1/2}|n^{-1}\sum_{t=1}^{n}(\hat{u}_t^{k-1} - \hat{\alpha}_{k-1})\phi_{\tilde{j}}(\mathbf{x}_t, \mathbf{b})|$$
    (c) $(\hat{\alpha}_k, \hat{\beta}_k) = \arg\min_{\alpha,\beta} n^{-1}\sum_{t=1}^{n} L(Y_t, \hat{f}^{k-1}(\mathbf{x}_t) + \alpha + \beta\phi_{\hat{j}_k}(\mathbf{x}_t; \hat{\mathbf{b}}_k))$
    (d) $\hat{f}^k(\mathbf{x}) = \hat{f}^{k-1}(\mathbf{x}) + \hat{\alpha}_k + \hat{\beta}_k\phi_{\hat{j}_k}(\mathbf{x}; \hat{\mathbf{b}}_k)$
    (e) $\hat{J}_k = \{\hat{j}_1, \cdots, \hat{j}_k\}$
   End for
3. For $k = m+1, \cdots, \tilde{m}$, repeat Step 2 but constrain the minimization
   defining $\hat{j}_k$ in Step 2(b) to $j \in \hat{J}_m$.
4. Output $\hat{J}_m$ and $\hat{f}^m(\cdot)$
```

Key features of MGB

- The j and $\tilde{j}$ in step 2(b) are taken over $1 \leq j \leq p_n$ if the cardinality $|\hat{J}_{k-1}| \leq K_n$, but are restricted to $\hat{J}_{K_n}$ otherwise. Therefore, the first stage of MGB stops when K_n distinct $\hat{j}_k$'s are included in the basis expansion. The second stage of MGB continues the preceding procedure with j restricted to the K_n distinct $\hat{j}_k$'s until the loss minimization step converges.
- We choose $(\hat{j}_k, \hat{\mathbf{b}}_k)$ with the smallest number $\#_j^{k-1}$ of iterations up to step $k-1$ among the $1 \leq j \leq p_n$ and $\mathbf{b} \in \Gamma$ that are within ϵ_n times the maximum squared correlation of $(\hat{u}_t^{k-1} - \hat{\alpha}_{k-1})_{1 \leq t \leq n}$ and $\phi_j(\mathbf{x}_t; \mathbf{b})_{1 \leq t \leq n}$; this corresponds to the “weak greedy” algorithms introduced by Temlyakov (2000).
- With a prespecified tolerance for convergence, MGB terminates when the loss is minimized within the tolerance limits.

Simulation study

- Below are the results of a simulation study on a neural network with one hidden layer.

Method	GB	GB+CV	MGB
MSPE	15.40	3.53	2.29

- Simulation setting:
  $y_t = \sum_{k=1}^{3} \beta_k S(\sum_{j=1}^{4} x_{tj} b_{jk}) + \epsilon_t, t = 1, \cdots, n(= 100)$, where S is the sigmoid function $S(z) = 1/(1 + \exp(-z))$, $\epsilon_t \sim N(0, 2.25)$,

  $(\beta_1, \beta_2, \beta_3) = (2, 3, 4), (b_{13}, b_{23}, b_{33}, b_{43}) = (-10, -11, 12, 9)$,
  $(b_{11}, b_{21}, b_{31}, b_{41}) = (1, -2, -3, 4), (b_{12}, b_{22}, b_{32}, b_{42}) = (8, -7, -6, 5)$,

  $x_{tj} = z_{tj} + w_t, \text{ for } 1 \leq t \leq n, 1 \leq j \leq 4$, in which $\{z_{tj} : 1 \leq t \leq n, 1 \leq j \leq 4\}$ are i.i.d. $N(0, 1)$, and $\{w_t : 1 \leq t \leq n\}$ are i.i.d. $N(0, 1)$ and also independent of z_{tj}.

Asymptotic theory of OGA for linear regression

- For the linear regression model

$$y_t = \alpha + \sum_{j=1}^{p} \beta_j x_{tj} + \epsilon_t = \alpha + \boldsymbol{\beta}^\top \mathbf{x}_t + \epsilon_t, t = 1, \cdots, n,$$

  let $y(\mathbf{x}) = \alpha + \boldsymbol{\beta}^\top \mathbf{x}$ and $\hat{y}_m(\mathbf{x}) = \hat{\alpha} + \hat{\boldsymbol{\beta}}_m^\top \mathbf{x}$, where $\hat{\boldsymbol{\beta}}_m$ is the OGA estimate of $\boldsymbol{\beta}$ that terminates after m iterations, and $\mathbf{x}$ is independent of $\{(\mathbf{x}_t, y_t), 1 \le t \le n\}$. Then under certain regularity assumptions (including $\log p_n = o(n)$ and $\sup_{n \ge 1} \sum_{j=1}^{p_n} |\beta_j \sigma_j| < \infty$ with $\sigma_j^2 = E x_{tj}^2$), as $K_n \to \infty$ such that $K_n = O((n/\log p_n)^{1/2})$,

$$\max_{1 \le m \le K_n} \left(\frac{E[\{y(\mathbf{x}) - \hat{y}_m(\mathbf{x})\}^2 | y_1, \mathbf{x}_1, \cdots, y_n, \mathbf{x}_n]}{m^{-1} + n^{-1} m \log p_n} \right) = O_p(1). \qquad (4)$$

- The rate (4) is shown to be optimal in the sense of bias-variance tradeoff.

Asymptotic theory of MGB

- The proof of (4) uses (a) Temlyakov's bounds for weak orthogonal greedy algorithms applied to the semipopulation version of OGA, and (b) exponential bounds relating $\hat{\boldsymbol{\beta}}_m$ to $(\beta_j; j \in \hat{J}_m)$, where $\hat{J}_m$ is the set of variables selected by OGA.
- This semipopulation version of OGA uses the variable selector $\hat{J}_m$ but approximates $y(\mathbf{x})$ by the best linear predictor of $y(\mathbf{x})$ based on $x_j, j \in \hat{J}_m$.
- We can extend (4) to MGB for the high-dimensional nonlinear regression model (1) under certain regularity assumptions on p_n, $\sum_{k=1}^{p_n} |\beta_k|$, ϕ_j and L.
- The basic idea consists of a corresponding semipopulation model and Temlyakov's bounds for weak greedy (instead of weak orthogonal greedy) algorithms.

Weak greedy algorithms and Temlyakov's bound

- H: Hilbert space with inner product $<,>$ and norm $\|\cdot\|$.
  D is a *dictionary* if H=closure of span of D and $\|g\| = 1\ \forall g \in D$.
  $\mathcal{A}(D, M) = \{f \in H : f = \sum_{k=1}^{\infty} c_k\psi_k, \psi_k \in D \text{ and } \sum_{k=1}^{\infty} |c_k| \leq M\}$.
- Weak greedy algorithm: Let $T = \{t_n, n \geq 1\}$ such that $0 < t_n \leq 1$. For each $m \geq 1$, inductively choose $\varphi_m^T \in D$ such that

$$| < f_{m-1}^T, \varphi_m^T > | \geq t_m \sup_{g \in D} | < f_{m-1}^T, g > |,$$

  and define
  $f_m^T = f_{m-1}^T - < f_{m-1}^T, \varphi_m^T > \varphi_m^T$, $G_m^T(f, D) = \sum_{j=1}^{m} < f_{j-1}^T, \varphi_j^T > \varphi_j^T$.
- The case $t_m \equiv 1$ corresponds to the pure greedy algorithm.
- Temlyakov's bound: If t_n is non-increasing and $f \in \mathcal{A}(D, M)$, then $\|f - G_m^T(f, D)\| \leq M(1 + \sum_{k=1}^{m} t_k^2)^{-\frac{1}{2}t_m/(2+t_m)}$.

Discussion and related works

- "Early stopping regularization" for gradient descent learning algorithms is related to the issue of when gradient boosting algorithm should be terminated, as has been discussed by Smale and Zhou (2005) and Yao, Rosasco and Caponnetto (2007). We introduced a new approach to address this issue, which becomes particularly difficult for high-dimensional nonlinear regression problems. This approach achieves "early stopping" by putting an upper bound on the number of distinct basis functions that have been included in a greedy manner, and then continues gradient boosting with these selected basis functions. Nonlinearities in the loss function and $\phi_j(\mathbf{x}_t; \mathbf{b})$ are handled by successive linear approximations via the gradients.
- The semipopulation model that we used effectively to analyze OGA in Ing and Lai (2011) can be extended to the present setting because it basically involves the greedy model selection rule used by the sample but then the paramters are assumed known for the (data-dependent) selected model.

Conclusion

- The basic model (1) arises in single-layer neural networks, as illustrated in the simulation study. Applications in the era of big data have led to multilayer (deep) learning networks. Gradient descent methods and their refinements (e.g. backpropagation) have been used to train these networks.
- Friedman (2002) has introduced stochastic gradient boosting and the theory we have developed can be extended to modified stochastic gradient boosting. More importantly, using the probabilistic interpretation of learning networks introduced by Hinton, Bengio, LeCun and Schmidhuber, we are working towards a theory of deep learning and the associated algorithms by combining recent advances in gradient boosting and adaptive filtering in hidden Markov models.