

# From Convex Analysis to Learning, Prediction, and Elicitation\*

## Lecture 12: Gradient Boosting, AdaBoost

Lunjia Hu

### 1 Gradient Descent

**Fact 1.** Let  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  be a differentiable function. For every  $x \in \mathbb{R}^d$ , as  $z \in \mathbb{R}^d$  approaches  $\mathbf{0}$ ,

$$f(x + z) - f(x) = \langle z, \nabla f(x) \rangle + o(\|z\|).$$

In particular, if  $z = -\alpha \nabla f(x)$  for  $\alpha \geq 0$ , as  $\alpha$  approaches  $0^+$ ,

$$f(x + z) - f(x) = -\alpha \|\nabla f(x)\|_2^2 + o(\alpha).$$

When  $\nabla f(x) \neq \mathbf{0}$  and  $\alpha$  is sufficiently small, we have  $f(x + z) < f(x)$ .

**Definition 1.** Let  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  be a differentiable convex function. We say  $f$  is  $\lambda$ -smooth w.r.t. norm  $\|\cdot\|$  if for every  $x, z \in \mathbb{R}^d$ ,

$$f(x + z) - f(x) \leq \langle z, \nabla f(x) \rangle + \frac{\lambda}{2} \|z\|^2.$$

**Gradient Descent.** Suppose  $f$  satisfies  $\lambda$ -smoothness w.r.t. the  $\ell_2$  norm  $\|\cdot\|_2$ . Choosing  $z = -\alpha \nabla f(x)$ , we get

$$f(x + z) - f(x) \leq -\alpha \|\nabla f(x)\|_2^2 + \frac{\lambda \alpha^2}{2} \|\nabla f(x)\|_2^2.$$

Choosing  $\alpha = \frac{1}{\lambda}$  to minimize the right-hand side, we get

$$f(x + z) - f(x) \leq -\frac{1}{2\lambda} \|\nabla f(x)\|_2^2.$$

### 2 Gradient Boosting

As we discuss in the previous section, in gradient descent, we choose the update vector  $z$  to align with the direction of the negative gradient  $-\nabla f(x)$ . Sometimes, we don't have full access to the gradient  $\nabla f(x)$ , but instead have access to some signal  $s \in \mathbb{R}^d$  such that  $\langle s, \nabla f(x) \rangle \geq w$  for some

---

\*<https://lunjiahu.com/convex-analysis/>

threshold  $w > 0$ . Assuming  $f$  is  $\lambda$ -smooth w.r.t. a general norm  $\|\cdot\|$ , we choose the update vector  $z = -\alpha s$  to align with the direction of  $-s$  and get

$$f(x + z) - f(x) \leq -\alpha \langle s, \nabla f(x) \rangle + \frac{\lambda \alpha^2}{2} \|s\|^2 \leq -\alpha w + \frac{\lambda \alpha^2}{2} \|s\|^2.$$

To minimize the right-hand side, we choose  $\alpha = \frac{w}{\lambda \|s\|^2}$  and get

$$f(x + z) - f(x) \leq -\frac{w^2}{2\lambda \|s\|^2}.$$

**FTRL as gradient boosting.** Recall the FTRL algorithm for OLO:

- Initialize  $z_1 = \mathbf{0} \in \mathbb{R}^d$ ;
- In each round  $t = 1, \dots, T$ ,
  1. play

$$x_t \leftarrow \arg \min_{x \in X} (\varphi(x) - \langle x, z_t \rangle), \quad (1)$$

2. observe  $y_t \in Y$ , and
3. update  $z_{t+1} \leftarrow z_t - \eta y_t$ .

When we analyze FTRL, we proved the following result:

**Lemma 2.** *Let  $\varphi : X \rightarrow \mathbb{R}$  be a convex function defined on a convex set  $X \subseteq \mathbb{R}^d$ , and let  $\psi : \mathbb{R}^d \rightarrow \mathbb{R}$  be its convex conjugate. Assume  $\psi$  is  $\lambda$ -smooth w.r.t. norm  $\|\cdot\|$ . In the FTRL algorithm above, we have*

$$\begin{aligned} \eta \langle x_t - x^*, y_t \rangle &= \Gamma_{\varphi, \psi}(x^*, z_t) - \Gamma_{\varphi, \psi}(x^*, z_{t+1}) + \Gamma_{\varphi, \psi}(x_t, z_{t+1}) \\ &\leq \Gamma_{\varphi, \psi}(x^*, z_t) - \Gamma_{\varphi, \psi}(x^*, z_{t+1}) + \frac{\lambda \eta^2}{2} \|y_t\|^2. \end{aligned} \quad (2)$$

This lemma can be understood as an instance of gradient boosting. Specifically, the gradient of  $\Gamma_{\varphi, \psi}(x^*, z_t)$  as a function of  $z_t$  is exactly  $x_t - x^*$ :

$$\begin{aligned} \nabla_{z_t} \Gamma_{\varphi, \psi}(x^*, z_t) &= \nabla_{z_t} (\varphi(x^*) + \psi(z_t) - \langle x^*, z_t \rangle) \\ &= \nabla_{z_t} \psi(z_t) - x^* \\ &= x_t - x^*. \end{aligned} \quad (\text{because } \Gamma_{\varphi, \psi}(x_t, z_t) = 0 \text{ by (1)})$$

Therefore, if the left-hand side of (2) is positive, we have  $\langle y_t, \nabla_{z_t} \Gamma_{\varphi, \psi}(x^*, z_t) \rangle > 0$ , so we can reduce  $\Gamma_{\varphi, \psi}(x^*, z_t)$  by updating  $z_t$  along the direction of  $-y_t$ . That is why we have the update rule  $z_{t+1} \leftarrow z_t - \eta y_t$  in FTRL.

**Applying gradient boosting to machine learning.** Suppose we have  $n$  data points from  $X \times \mathbb{R}$ :  $(x_1, y_1), \dots, (x_n, y_n)$ . A model  $h : X \rightarrow \mathbb{R}$  corresponds to a vector  $\mathbf{h} = (h_1, \dots, h_n) := (h(x_1), \dots, h(x_n))$ . The (empirical) loss of this model is

$$L(\mathbf{h}) = \sum_{i=1}^n \ell(h_i, y_i). \quad (3)$$

The gradient of  $L$  is

$$\nabla L(\mathbf{h}) = \left( \frac{\partial}{\partial h} \ell(h_1, y_1), \dots, \frac{\partial}{\partial h} \ell(h_n, y_n) \right). \quad (4)$$

Thus, if we can find a signal  $\mathbf{s} \in \mathbb{R}^n$  such that  $\langle \mathbf{s}, \nabla L(\mathbf{h}) \rangle > 0$ , then we can apply gradient boosting to update the current model  $\mathbf{h}$  to reduce the loss  $L$ . In the next section we will see a famous example of gradient boosting: AdaBoost.

### 3 AdaBoost

In the most classic version of AdaBoost, the labels  $y_1, \dots, y_n$  are binary:  $y_i \in \{-1, 1\}$ . The loss function is  $\ell(h, y) := e^{-yh}$  for  $y \in \{-1, 1\}$  and  $h \in \mathbb{R}$ . By (3),

$$L(\mathbf{h}) = \sum_{i=1}^n e^{-y_i h_i}.$$

By (4),

$$\nabla L(\mathbf{h}) = (e^{-y_1 h_1} (-y_1), \dots, e^{-y_n h_n} (-y_n)).$$

Let  $w_i := e^{-y_i h_i}$  be the weight on point  $(x_i, y_i)$ . Suppose we have a weak classifier  $s : X \rightarrow \{-1, 1\}$  such that its weighted error is below  $1/2$ :

$$\sum_{i=1}^n w_i \mathbb{I}(s(x_i) \neq y_i) < \frac{1}{2} \sum_{i=1}^n w_i. \quad (5)$$

Note that  $\mathbb{I}(s(x_i) \neq y_i) = (1 - y_i s(x_i))/2$ , so (5) is equivalent to

$$\sum_{i=1}^n w_i y_i s(x_i) > 0. \quad (6)$$

Define  $w_{\text{cor}}$  as the sum of  $w_i$  where  $(x_i) = y_i$  (i.e.  $y_i s(x_i) = 1$ , or  $s(x_i)$  is *correct*), and define  $w_{\text{mis}}$  as the sum of  $w_i$  where  $s(x_i) \neq y_i$  (i.e.  $y_i s(x_i) = -1$ , or  $s(x_i)$  makes a *mistake*). Inequality (6) is equivalent to

$$w_{\text{cor}} > w_{\text{mis}}.$$

Define  $\mathbf{s} = (s_1, \dots, s_n) := (s(x_1), \dots, s(x_n))$ . Inequality (6) is equivalent to  $\langle \mathbf{s}, \nabla L(\mathbf{h}) \rangle < 0$ . We can thus run gradient boosting to reduce  $L$ . That is, for  $\alpha \geq 0$ , we consider update  $\mathbf{h}$  to  $\mathbf{h}' = \mathbf{h} + \alpha \mathbf{s}$ .

$$L(\mathbf{h} + \alpha \mathbf{s}) = \sum_{i=1}^n w_i e^{-\alpha y_i s_i} = w_{\text{cor}} e^{-\alpha} + w_{\text{mis}} e^{\alpha}.$$

The optimal choice of  $\alpha$  that minimizes  $L(\mathbf{h} + \alpha\mathbf{s})$  is

$$\alpha = \frac{1}{2} \ln(w_{\text{cor}}/w_{\text{mis}}).$$

For this choice of  $\alpha$ , the new loss is

$$L(\mathbf{h} + \alpha\mathbf{s}) = 2\sqrt{w_{\text{cor}}w_{\text{mis}}} < w_{\text{cor}} + w_{\text{mis}} = L(\mathbf{h}).$$

If we additionally assume that  $w_{\text{cor}}/w_{\text{mis}} \geq 1 + \varepsilon > 1$ , we have  $L(\mathbf{h} + \alpha\mathbf{s}) \leq (1 - \Omega(\varepsilon^2))L(\mathbf{h})$ . If initially  $\mathbf{h} = (0, \dots, 0)$  and  $L(\mathbf{h}) = n$ , after  $O(\varepsilon^{-2} \log n)$  such updates, we will have  $L(\mathbf{h}) < 1$ , which implies that  $\text{sign}(h_i) = y_i$  for every  $i = 1, \dots, n$ .