2022

Lecture 5: AdaBoost

Week 5

Lecturer: Tianyu Wang

1 AdaBoost

Consider a set of classifiers $\mathcal{H} = \{h : \mathbb{R}^d \to \{+1, -1\}\}$, we can construct a weighted version of this classifier $f : \mathbb{R}^d \to \mathbb{R}$ such that

$$f(x) = \text{sign}\left(\sum_{j=1}^{K} \lambda_j h_j(x)\right), \tag{1}$$

for some $h_1, h_2, \cdots, h_K \in \mathcal{H}$.

AdaBoost is an algorithm for constructing such a weighted model.

Algorithm 1 AdaBoost

1: **Input**: Dataset $\{(x_i, y_i)\}_{i=1}^n$ (uniformly weighted), class of functions $\mathcal{H}$.
2: **Initialization**: Pick $f_0 \in \mathcal{H}$.
3: **for** $t = 0, 1, \cdots, T-1$ **do**
4: Pick $h_{j_{t+1}} \in \mathcal{H}$ so that $d_t^- = \frac{\sum_{i: y_i h_{j_{t+1}}(x_i) = -1} \exp(-y_i f_t(x_i))}{\sum_{i=1}^n \exp(-y_i f_t(x_i))} = \frac{1}{2} - \eta_t$ for some η_t.
5: Let $d_t^- = \frac{\sum_{i: y_i h_{j_{t+1}}(x_i) = -1} \exp(-y_i f_t(x_i))}{\sum_{i=1}^n \exp(-y_i f_t(x_i))}$ and let

$$\alpha_t = \frac{1}{2} \log \frac{1 - d_t^-}{d_t^-}.$$

6: Let $f_{t+1} = f_t + \alpha_t h_{j_{t+1}}$.
7: **end for**
8: **Output**: the model $sgn(f_T)$, where $sgn(\cdot)$ is the sign function.

1.1 Convergence Rate of AdaBoost

Let

$$d_t^+ = \frac{\sum_{i: y_i h_{j_{t+1}}(x_i) = +1} \exp\left(-y_i f_t(x_i)\right)}{\sum_{i=1}^n \exp\left(-y_i f_t(x_i)\right)},$$

and

$$d_t^- = \frac{\sum_{i:y_i h_{j_{t+1}}(x_i)=-1} \exp\left(-y_i f_t(x_i)\right)}{\sum_{i=1}^n \exp\left(-y_i f_t(x_i)\right)} = 1 - d_t^+.$$

Theorem 1.1. *Let $\eta_t = \frac{1}{2} - d_t^-$. For any T, it holds that*

$$\frac{1}{n}\sum_{i=1}^n \mathbb{I}_{[y_i \neq f_T(x_i)]} \leq L_0 \exp\left(-2\sum_{t=1}^T \eta_t^2\right),$$

where $L_0 = \frac{1}{n}\sum_{i=1}^n \exp\left(-y_i f_0(x_i)\right)$.

Proof. Let $L_t = \frac{1}{n}\sum_{i=1}^n \exp\left(-y_i f_t(x_i)\right)$. Note that this L_t is a smooth convex upper bound of the empirical risk $\frac{1}{n}\sum_{i=1}^n \mathbb{I}_{[y_i f_t(x_i)<0]}$.

We have

$$\begin{aligned}
L_{t+1} &= \frac{1}{n}\sum_{i=1}^n \exp\left(-y_i f_t(x_t)\right) \exp\left(-\alpha_t y_i h_{j_{t+1}}(x_i)\right) \\
&= \frac{1}{n}\sum_{i:y_i h_{j_{t+1}}(x_i)=+1} \exp\left(-y_i f_t(x_i)\right)\exp\left(-\alpha_t\right) \\
&\quad + \frac{1}{n}\sum_{i:y_i h_{j_{t+1}}(x_i)=-1} \exp\left(-y_i f_t(x_i)\right)\exp\left(\alpha_t\right) \\
&= \frac{\sum_{i=1}^n \exp\left(-y_i f_t(x_i)\right)}{n}\left[(1-d_t^-)\sqrt{\frac{d_t^-}{1-d_t^-}} + d_t^-\sqrt{\frac{1-d_t^-}{d_t^-}}\right] \\
&= L_t 2\sqrt{d_t^-(1-d_t^-)} \\
&= L_t\sqrt{(1-2\eta_t)(1+2\eta_t)}.
\end{aligned}$$

Thus we have

$$L_T = L_0\prod_{t=1}^T \sqrt{1-4\eta_t^2} \leq L_0\prod_{t=1}^T \sqrt{\exp(-4\eta_t^2)} = L_0\prod_{t=1}^T \exp(-2\eta_t^2) = L_0\exp\left(-2\sum_{t=1}^T \eta_t^2\right).$$

□

Weaker Learners and Boosting

In Theorem 1.1, if $\eta_t^2 > C$ for some constant C, the training error decay exponentially fast. The condition that $\eta_t^2 > C$ is called the *weaker learner/learning assumption*. AdaBoost shows that one can construct a stronger classifier by an ensemble of weaker classifiers.

AdaBoost as Coordinate Minimization

When $\mathcal{H}$ is a finite (but probably very large set), AdaBoost can be viewed a coordinate minimization algorithm. Next we discuss the Coordinate Minimization.

1.2 Coordinate Minimization and Greedy Coordinate Descent

Given a convex objective $f : \mathbb{R}^d \to \mathbb{R}$, one can minimize each coordinate one-by-one until convergence. Below we summarize the Coordinate Minimization algorithm.

Algorithm 2 Coordinate Minimization

Input: convex objective $f : \mathbb{R}^d \to \mathbb{R}$.
Initialization: starting point $\mathbf{x}_0$.
for $t = 1, 2, \cdots$ **do**
/* We use the notation $\mathbf{x}_t = (x_{t,1}, \cdots, x_{t,d})$. */
    $x_{t,1} = \arg\min_x f(x, x_{t-1,2}, \cdots, x_{t-1,d})$;
    $x_{t,2} = \arg\min_x f(x_{t,1}, x, x_{t-1,3}, \cdots, x_{t-1,d})$;
    $x_{t,3} = \arg\min_x f(x_{t,1}, x_{t,2}, x, x_{t-1,4}, \cdots, x_{t-1,d})$;
$\cdots$
    $x_{t,3} = \arg\min_x f(x_{t,1}, \cdots, x_{t,d-1}, x)$;
end for

In general, the coordinate minimization algorithm converges when the objective is smooth convex, and sometimes converges when the objective is not convex and not smooth. Below we prove a convergence theorem smooth convex objectives defined over $\mathbb{R}^2$, which serves as a illustration of analysis of algorithms for convex optimization.

Theorem 1.2. *Consider a differentiable convex function $f : \mathbb{R}^2 \to \mathbb{R}$. Suppose there exists $L > 0$, such that $|\partial_1 f(x_1 + h, x_2) - \partial_1 f(x_1, x_2)| < L|h|$ and $|\partial_2 f(x_1, x_2 + h) - \partial_2 f(x_1, x_2)| < L|h|$ for all $x_1, x_2, h \in \mathbb{R}$. Let f be a convex function with a unique minimizer. Denote by $\mathbf{x}^*$ this minimizer. Let $R(\mathbf{x}_0) = \max_{\mathbf{x}}\{\|\mathbf{x} - \mathbf{x}^*\| : f(\mathbf{x}) \le f(\mathbf{x}_0)\}$. Algorithm 2 satisfies*

$$f(\mathbf{x}_{t+1}) - f(\mathbf{x}^*) \le \frac{2LR(\mathbf{x}_0)}{t},$$

where $\mathbf{x}_t$ is computed by the t-th iteration of Algorithm 2 and $\mathbf{x}^* \in \arg\min_x(f(x))$.

We first define L-smoothness which is widely used in analysis of optimization algorithm, and prove a lemma for general smooth functions.

Definition 1.3. A differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is called L-smooth if

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}')\|_2 \le L\|\mathbf{x} - \mathbf{x}'\|_2, \qquad \forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^d.$$

Note that in Theorem 1.2, the functions $f(\cdot, x)$ and $f(x, \cdot)$ are L-smooth. L-smooth functions have the following property.

Proposition 1.4. *If $f : \mathbb{R}^d \to \mathbb{R}$ is L-smooth, then*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|_2^2.$$

Proof. This proposition is essentially a corollary of the Taylor's Theorem. Let $\nabla^2 f(\mathbf{x})$ be the Hessian of f at $\mathbf{x}$. We have, for any $\mathbf{x}$,

$$\begin{aligned}
\mathbf{z}^\top \left[\nabla^2 f(\mathbf{x})\right] \mathbf{z} &= D_{\mathbf{z}}^2 f(\mathbf{x}) \\
&\qquad \text{(Fact: } \mathbf{z}^\top \left[\nabla^2 f(\mathbf{x})\right] \mathbf{z} \text{ equals the second order derivative of } f \text{ at } \mathbf{x} \text{ along the direction of } \mathbf{z}.) \\
&\qquad (D_{\mathbf{z}} \text{ denotes the derivative along derivation } \mathbf{z}.) \\
&= D_{\mathbf{z}} \langle \nabla f(\mathbf{x}), \mathbf{z} \rangle \\
&\qquad \text{(Recall } \langle \nabla f(\mathbf{x}), \mathbf{z} \rangle \text{ is the first order derivative of } f \text{ at } \mathbf{x} \text{ along the direction of } \mathbf{z}.) \\
&= \lim_{\tau \to 0} \frac{\langle \nabla f(\mathbf{x} + \tau z), \mathbf{z} \rangle - \langle \nabla f(\mathbf{x}), \mathbf{z} \rangle}{\tau} && \text{(The limit definition of derivative.)} \\
&\leq \lim_{\tau \to 0} \frac{\|\nabla f(\mathbf{x} + \tau \mathbf{z}) - \nabla(\mathbf{x})\|_2 \|\mathbf{z}\|_2}{\tau} && \text{(Cauchy-Schwarz inequality)} \\
&\leq \lim_{\tau \to 0} \frac{L\tau \|\mathbf{z}\|_2^2}{\tau} && \text{(by } L\text{-smoothness condition.)} \\
&= L\|\mathbf{z}\|_2^2.
\end{aligned}$$

By Taylor's theorem, it holds that

$$f(\mathbf{y}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{1}{2}(\mathbf{y} - \mathbf{x})^\top \nabla^2 f(\mathbf{x}')(\mathbf{y} - \mathbf{x}),$$

for some x' on the line segment connecting $\mathbf{x}$ and $\mathbf{y}$. Then we have

$$\begin{aligned}
f(\mathbf{y}) &= f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{1}{2}(\mathbf{y} - \mathbf{x})^\top \nabla^2 f(\mathbf{x}')(\mathbf{y} - \mathbf{x}) \\
&\leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|_2^2. && \text{(by what we derived above)}
\end{aligned}$$

□

Proof of Theorem 1.2.

Lemma 1.5. *Instate the assumptions of Theorem 1.2. Let $\mathbf{x}_{t+\frac{1}{2}} = (x_{t+1,1}, x_{t,2})$. Then we have*

$$f(\mathbf{x}_t) - f(\mathbf{x}_{t+\frac{1}{2}}) \geq \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|_2^2.$$

Proof. Since $x_{t+1,1} \in \arg\min_x f(x, x_{t,2})$, we have

$$f(x_{t+1,1}, x_{t,2}) \leq f\left(x_{t,1} - \frac{1}{L}\partial_1 f(\mathbf{x}_t), x_{t,2}\right).$$

Thus we have

$$
\begin{aligned}
f(\mathbf{x}_t) - f(\mathbf{x}_{t+\frac{1}{2}}) &= f(x_{t,1}, x_{t,2}) - f(x_{t+1,1}, x_{t,2}) \\
&\geq f(x_{t,1}, x_{t,2}) - f\left(x_{t,1} - \frac{1}{L}\partial_1 f(\mathbf{x}_t), x_{t,2}\right) \\
&\geq f(x_{t,1}, x_{t,2}) - \left(f\left(x_{t,1}, x_{t,2}\right) - \partial_1 f\left(\mathbf{x}_t\right)\left(\frac{1}{L}\partial_1 f(\mathbf{x}_t)\right) + \frac{L}{2}\frac{1}{L^2}\left(\partial_1 f(\mathbf{x}_t)\right)^2\right) \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{(Apply Proposition 1.4 to function } f(\cdot, x_{t,2}).\text{)} \\
&= \frac{1}{2L}\left(\partial_1 f(\mathbf{x}_t)\right)^2 \\
&= \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|_2^2 \qquad \text{(since } \partial_2 f(\mathbf{x}_t) \text{ by the algorithm procedure.)}
\end{aligned}
$$

□

Proposition 1.6. *Let $\{\alpha_k\}_{k=0}^{\infty}$ be a nonnegative sequence such that, for all $k \in \mathbb{N}$, $\alpha_k - \alpha_{k+1} \geq \gamma\alpha_k^2$ and $\alpha_0 \leq \beta\gamma$ for some positive β and γ. Then it holds that*

$$\alpha_k \leq \frac{1}{\gamma k}, \quad \forall k = 1, 2, \cdots .$$

Proof. For all $k = 1, 2, \cdots$, we have

$$\frac{1}{\alpha_k} - \frac{1}{\alpha_{k-1}} = \frac{\alpha_{k-1} - \alpha_k}{\alpha_{k-1}\alpha_k} \geq \frac{\gamma\alpha_{k-1}^2}{\alpha_{k-1}\alpha_k} = \frac{\gamma\alpha_{k-1}}{\alpha_k} \geq \gamma,$$

where the last inequality uses $\alpha_{k-1} \geq \alpha_k + \gamma\alpha_{k-1}^2 \geq \alpha_k$. Thus, we have

$$\frac{1}{\alpha_k} = \left(\frac{1}{\alpha_k} - \frac{1}{\alpha_{k-1}}\right) + \left(\frac{1}{\alpha_{k-1}} - \frac{1}{\alpha_{k-2}}\right) + \cdots + \left(\frac{1}{\alpha_1} - \frac{1}{\alpha_0}\right) + \frac{1}{\alpha_0} \geq k\gamma + \frac{1}{\alpha_0} \geq k\gamma$$

which finishes the proof. □

With Lemma 1.5 and Proposition 1.6, we start the proof of Theorem 1.2.

Note that $f(\mathbf{x}_0) \geq f(\mathbf{x}_1) \geq f(\mathbf{x}_2) \geq \cdots$. Thus we have $\|\mathbf{x}_t - \mathbf{x}^*\|_2 \leq R(\mathbf{x}_0)$ for all $t = 1, 2, \cdots$. Thus we have

$$
\begin{aligned}
f(\mathbf{x}_t) - f(\mathbf{x}^*) &\leq \nabla f(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{x}^*) \\
&\leq \|\nabla f(\mathbf{x}_t)\|_2\|\mathbf{x}_t - \mathbf{x}^*\|_2 \qquad \text{(by Cauchy-Schwarz inequality)} \\
&\overset{(i)}{\leq} R(\mathbf{x}_0)\|\nabla f(\mathbf{x}_t)\|_2 \qquad \text{(since } \|\mathbf{x}_t - \mathbf{x}^*\| \leq R(\mathbf{x}_0) \text{ as shown above.)}
\end{aligned}
$$

By Lemma 1.5, we have

$$f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) \geq f(\mathbf{x}_t) - f(\mathbf{x}_{t+\frac{1}{2}}) \overset{(ii)}{\geq} \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|_2^2.$$

Combining (i) and (ii) gives

$$(f(\mathbf{x}_t) - f(\mathbf{x}^*)) - (f(\mathbf{x}_{t+1}) - f(\mathbf{x}^*)) \geq \frac{1}{2L}\|\nabla f(\mathbf{x}_t)\|_2^2 \geq \frac{(f(\mathbf{x}_t) - f(\mathbf{x}^*))^2}{2LR(\mathbf{x}_0)^2}.$$

Also, we have

$$\begin{aligned}
f(\mathbf{x}_0) - f(\mathbf{x}^*) &\leq \nabla f(\mathbf{x}^*)^\top (\mathbf{x}_0 - \mathbf{x}^*) + \frac{L}{2}\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2 && \text{(by Proposition 1.4)} \\
&= \frac{L}{2}\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2 && \text{(since } \nabla f(\mathbf{x}^*) = 0\text{)} \\
&\leq \frac{R(\mathbf{x}_0)}{2}, . && \text{(by definition of } R(\mathbf{x}_0)\text{)}
\end{aligned}$$

Since the sequence $\{f(\mathbf{x}_t) - f(\mathbf{x}^*)\}$ satisfies the conditions in Proposition 1.6. Applying Proposition 1.6 with proper constants finishes the proof.

□

1.3 AdaBoost as Coordinate Minimization

As promised before, AdaBoost can be viewed as a coordinate minimization algorithm. Let's revisit AdaBoost with a finite (but very large) hypothesis space $\mathcal{H} = \{h_1, h_2, \cdots, h_K\}$. In this case, an ensemble of functions in $\mathcal{H}$ can be written out as $f = \sum_{j=1}^K \lambda_j h_j$ for some λ_j. The objective function, in terms of λ, is

$$L(\lambda) = \frac{1}{n}\sum_{i=1}^n \exp\left(-y_i \sum_{j=1}^K \lambda_j h_j(x_i)\right).$$

We can perform coordinate minimization in λ to derive AdaBoost. First of all, note that K is very large, and we may not be able to iteration over all of the coordinates. Thus each time we (arbitrarily) pick a coordinate j_{t+1}, and minimize along this coordinate.

Then we minimize along this coordinate j_{t+1}. Consider setting

$$\partial_{j_{t+1}} L(\lambda_t + \alpha_t \boldsymbol{e}_{j_{t+1}}) = 0,$$

and solve for α_t. This gives

$$\begin{aligned}
0 = \frac{d}{d\alpha_t} L(\lambda_t + \alpha_t \boldsymbol{e}_{j_{t+1}}) &= \sum_{i=1}^n y_i h_{j_{t+1}}(x_i) \exp\left(-y_i \sum_{j=1}^K \lambda_{t,j} h_j(x_i) - \alpha_t y_i h_{j_{t+1}}(x_i)\right) \\
&= \sum_{i: y_i h_{j_{t+1}}(x_i) = +1} \exp(-y_i f_t(x_i)) \exp(-\alpha_t) \\
&\quad - \sum_{i: y_i h_{j_{t+1}}(x_i) = -1} \exp(-y_i f_t(x_i)) \exp(\alpha_t),
\end{aligned}$$

which gives

$$\exp(2\alpha_t) = \frac{\sum_{i: y_i h_{j_{t+1}}(x_i) = +1} \exp\left(-y_i f_t(x_i)\right)}{\sum_{i: y_i h_{j_{t+1}}(x_i) = -1} \exp\left(-y_i f_t(x_i)\right)}.$$

Thus we have

$$\alpha_t = \frac{1}{2} \log \frac{\sum_{i: y_i h_{j_{t+1}}(x_i) = +1} \exp\left(-y_i f_t(x_i)\right)}{\sum_{i: y_i h_{j_{t+1}}(x_i) = -1} \exp\left(-y_i f_t(x_i)\right)} = \frac{1}{2} \log \frac{1 - d_t^-}{d_t^-}.$$

This fully recovers AdaBoost.

In practice, people often use decision trees as weak learners.

2 Decision Trees (Classifiers)

People usually use decision trees (of bounded depth and bounded width) as $\mathcal{H}$.

We will use the slide by Byron Boots.

Acknowledgement

AdaBoost is due to Freund and Schapire. The coordinate minimization analysis is due to Beck and Terruashvili. A thank you to wikipedia contributors.