# Chapter 10

# Boosting and Additive Trees

## Derivation of Adaboost

[2][3]

Let $\mathcal{F}$ denote the set of all weak classifiers, where each weak classifier is a mapping from the space of measurement $X$ to the binary label $\{-1, 1\}$. Let us further denote $\text{lin}\{\mathcal{F}\}$ as the set of all linear combinations of the functions in $\mathcal{F}$. Note that the codomain of $F \in \text{lin}\{\mathcal{F}\}$ is no longer restricted to be $\{-1, 1\}$, as in general it can take any value in $\mathbb{R}$.

For any $F \in \text{lin}\{\mathcal{F}\}$, we define a cost function $\text{cost}(F) = \sum_{i=1}^{m} e^{-y_i F(x_i)}$, where $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ is the set of all training samples expressed in the form of (measurement, label) pairs. For each measurement $x \in X$, we can view $|F(x)|$ as the amount of belief that the label of $x$ is $\text{sign}(F(x))$[1], and thus we can view $\text{cost}(F)$ as the penalty caused by the deviation of the belief $F$ induced on the training samples and their true labels. Then, it makes sense to focus on the problem of finding $F \in \text{lin}\{\mathcal{F}\}$ with minimum $\text{cost}(F)$. Next, we will introduce a greedy iterative approach to obtain a sub-optimal solution to the problem, and it will become clear later that the approach is the AdaBoost algorithm.

Assume that the algorithm works in iterations, and after the $t^{\text{th}}$ iteration, we obtain a function $F_t \in \text{lin}\{\mathcal{F}\}$, which can be expressed as $F_t = \alpha_1 f_1 + \alpha_2 f_2 + \ldots, \alpha_t f_t$ with $f_1, f_2, \ldots, f_t \in \mathcal{F}$ and $\alpha_1, \alpha_2, \ldots, \alpha_t \in \mathbb{R}$. In the $(t+1)^{\text{th}}$ iteration, Let us try to find a $(\alpha, f) \in \mathbb{R} \times \mathcal{F}$ pair such that $\text{cost}(F_t + \alpha f)$ is smaller than $\text{cost}(F_t)$.

$$
\begin{aligned}
&\text{cost}(F_t + \alpha f) - \text{cost}(F_t) \\
=&\sum_{i=1}^{m} e^{-y_i F_t(x_i)} e^{-y_i \alpha f(x_i)} - \sum_{i=1}^{m} e^{-y_i F_t(x_i)} \\
=&\sum_{i=1}^{m} e^{-y_i F_t(x_i)} \left(1 - \alpha y_i f(x_i) e^{-y_i \alpha f(x_i)} + o(\alpha^2)\right) - \sum_{i=1}^{m} e^{-y_i F_t(x_i)} \\
=&\sum_{i=1}^{m} \alpha \left(-y_i f(x_i)\right) e^{-y_i \alpha f(x_i)} e^{-y_i F_t(x_i)} + o(\alpha^2)
\end{aligned}
$$

[1] $\text{sign}(a) = 1$ if $a > 0$ and $\text{sign}(a) = -1$ if $a < 0$

Note that $(-y_i f(x_i)) = 1$ if $y_i \neq f(x_i)$ and $(-y_i f(x_i)) = -1$ if $y_i = f(x_i)$, and thus we can rewrite the above equation as

$$
\begin{aligned}
&\sum_{i:y_i\neq f(x_i)} \alpha e^{-y_i\alpha f(x_i)} e^{-y_i F_t(x_i)} - \sum_{i:y_i=f(x_i)} \alpha e^{-y_i\alpha f(x_i)} e^{-y_i F_t(x_i)} + o(\alpha^2) \\
=&\sum_{i:y_i\neq f(x_i)} \alpha e^{\alpha} e^{-y_i F_t(x_i)} - \sum_{i:y_i=f(x_i)} \alpha e^{-\alpha} e^{-y_i F_t(x_i)} + o(\alpha^2) \\
=&\alpha(e^{\alpha}+e^{-\alpha}) \sum_{i:y_i\neq f(x_i)} e^{-y_i F_t(x_i)} - \alpha e^{-\alpha}\sum_{i=1}^{m} e^{-y_i F_t(x_i)} + o(\alpha^2) \\
=&\alpha e^{-\alpha}\left(\sum_{i=1}^{m} e^{-y_i F_t(x_i)}\right)\left(\frac{e^{\alpha}+e^{-\alpha}}{e^{-\alpha}} \sum_{i:y_i\neq f(x_i)} \frac{e^{-y_i F_t(x_i)}}{\sum_{j=1}^{m} e^{-y_j F_t(x_j)}} - 1\right) + o(\alpha^2)
\end{aligned}
\tag{10.1}
$$

From the above equation, we know that if we fixed the alpha to be very small, then the desired $f$ we are looking for (denoted as $f_{t+1}$) should be the one that mininize the difference in the cost. More precisely,

$$f_{t+1} = \arg\min_{f\in\mathcal{F}} \sum_{i=1}^{m} \mathbf{1}\left[y_i \neq f(x_i)\right] \frac{e^{-y_i F_t(x_i)}}{\sum_{j=1}^{m} e^{-y_j F_t(x_j)}}. \tag{10.2}$$

Moreover, from Equation (10.1), in order to drive the cost in the descent direction, we should have

$$\epsilon_{t+1} \triangleq \min_{f\in\mathcal{F}} \sum_{i=1}^{m} \mathbf{1}\left[y_i \neq f(x_i)\right] \frac{e^{-y_i F_t(x_i)}}{\sum_{j=1}^{m} e^{-y_j F_t(x_j)}} < 1/2. \tag{10.3}$$

Now that we know how to find $f_{t+1} \in \mathcal{F}$ which gives the steepest descent in the cost when $\alpha$ is very small, the next step is to find $\alpha_t$ which yield the largest descent when the direction is fixed to $f_{t+1}$. Such a $\alpha_t$ can be found by simply taking the derivative of cost$(F_t + \alpha f_{t+1})$ over $\alpha$ and enforce it to be zero.

$$
\begin{aligned}
\frac{d\text{cost}(F_t+\alpha f_{t+1})}{d\alpha} =& -\sum_{i=1}^{m} e^{-y_i F_t(x_i)} y_i f_{t+1}(x_i) e^{-y_i\alpha f_{t+1}(x_i)} = 0 \\
\implies e^{2\alpha_{t+1}} =& \frac{\sum_{i=1}^{m} \mathbf{1}\left[f_{t+1}(x_i) = y_i\right] e^{-y_i F_t(x_i)}}{\sum_{i=1}^{m} \mathbf{1}\left[f_{t+1}(x_i) \neq y_i\right] e^{-y_i F_t(x_i)}} \\
\implies \alpha_{t+1} =& \frac{1}{2}\log\left(\frac{1-\epsilon_{t+1}}{\epsilon_{t+1}}\right)
\end{aligned}
$$

where $\epsilon_{t+1}$ is defined in Equation (10.3)

## Exercise 10.1

The updated exponential lose when $G_m$ is plugged in to the Additive Model is expressed as

$$
\begin{aligned}
&(e^{\beta}-e^{-\beta})\sum_{i=1}^{N} w_i^{(m)} I(y_i \neq G(x_i)) + e^{-\beta}\sum_{i=1}^{N} w_i^{(m)} \\
=&\sum_{i=1}^{N} w_i^{(m)}\left[e^{\beta}\text{err} - e^{-\beta}\text{err} + e^{-\beta}\right].
\end{aligned}
$$

Taking the derivative of the term within the square bracket and enforcing it to be zero, we obtain

$$\begin{aligned} & e^{\beta}\text{err} + e^{-\beta}\text{err} - e^{-\beta} = 0 \\ \Rightarrow & e^{2\beta} = \frac{1-\text{err}}{\text{err}} \Rightarrow \beta = \frac{1}{2}\log\frac{1-\text{err}}{\text{err}}. \end{aligned}$$

## Exercise 10.2

Let us rewrite $f(x)$ as $f_x$ to emphasis that we are working on a fixed $x$.

$$\mathbb{E}\left[e^{Yf_x}\right]_{Y|x} = \mathbb{P}\left(Y=1|X\right)e^{-f_x} + \mathbb{P}\left(Y=-1|X\right)e^{f_x}.$$

The $f_x$ that minimizes the above term is

$$-\mathbb{P}\left(Y=1|X\right)e^{-f_x^*} + \mathbb{P}\left(Y=-1|X\right)e^{f_x^*} = 0 \Rightarrow f_x^* = \frac{1}{2}\frac{\mathbb{P}\left(Y=1|X\right)}{\mathbb{P}\left(Y=-1|X\right)}.$$