# Adaboost

$G_m(x)$: base learner

$\alpha_m$: coefficient

additive model: $f(x) = \sum_{m=1}^{M} \alpha_m G_m(x)$

aim at minimizing $Loss(y, f(x))$

$$\min_{(\alpha_m, G_m)} \sum_{i=1}^{N} L(y_i, \sum_{m=1}^{M} \alpha_m G_m(x_i))$$

$(x_1, y_1), (x_2, y_2) \cdots, (x_N, y_N)$ train dataset

Assume $\alpha$ and $G(x)$ are fixed in the first $(m-1)$ iterations so $f_m(x) = f_{m-1}(x) + \alpha_m G_m(x)$. We only need to min the loss of $\alpha_m$ and $G_m$.

$$L(y, f(x)) = e^{-yf(x)}$$

$$(\alpha_m, G_m(x)) = \arg\min_{(\alpha, G)} \sum_{i=1}^{N} e^{-y_i f_m(x_i)}$$

$$= \arg\min_{(\alpha, G)} \sum_{i=1}^{N} e^{-y_i (f_{m-1}(x_i) + \alpha G(x_i))}$$

Make $w_i^{(m)} = e^{-y_i f_{m-1}(x_i)}$:

$$\sum_{i=1}^{N} w_i^{(m)} e^{-y_i \alpha G(x_i)} = e^{-\alpha} \sum_{y_i = G(x_i)} w_i^{(m)} + e^{\alpha} \sum_{y_i \neq G(x_i)} w_i^{(m)} \quad (a)$$

$$= (e^{\alpha} - e^{-\alpha}) \sum_{i=1}^{N} w_i^{(m)} \mathbb{I}(y_i \neq G(x_i)) + e^{-\alpha} \sum_{i=1}^{N} w_i^{(m)} \quad (b)$$

Keys:

① The $G_m(x)$ which make loss min is equivalent to the $G_m(x)$ which makes $\sum_{i=1}^{N} w_i^{(m)} \mathbb{I}(y_i \neq G(x_i))$ min. So we can conclude that each base learner is based on the minimisation of weighted errors.

② $w_i^{(m+1)} = e^{-y_i f_m(x_i)} = e^{-y_i (f_{m-1}(x_i) + \alpha G(x_i))}$

$$= e^{-y_i f_{m-1}(x_i)} \cdot e^{-y_i \alpha G(x_i)} = w_i^{(m)} e^{-y_i \alpha_m G_m(x_i)}$$

if $a_m > 0$,

$y_i = G_m(x_i) \Rightarrow w_i^{(m+1)} = w_i^{(m)} e^{-\alpha_m}$. The weights for correctly classified samples will be smaller.

$y_i \neq G_m(x_i) \Rightarrow w_i^{(m+1)} = w_i^{(m)} e^{\alpha_m}$.

Wrongly classified will larger.

③ set weighted error rate of $G_m(x)$ in train dataset:

$$\varepsilon_m = \frac{\sum_{i=1}^{N} w_i^{(m)} \mathbb{I}(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i^{(m)}}$$

Compute the deriv of (a) over $\alpha$ and $=0$:

$$-e^{-\alpha} \sum_{y_i = G(x_i)} w_i^{(m)} + e^{\alpha} \sum_{y_i \neq G(x_i)} w_i^{(m)} = 0$$

$(\times e^{\alpha})$
$$e^{2\alpha} = \frac{\sum_{=} w_i^{(m)}}{\sum_{\neq} w_i^{(m)}} = \frac{1-\varepsilon_m}{\varepsilon_m}$$

$$\Rightarrow \alpha_m = \frac{1}{2} \ln \frac{1-\varepsilon_m}{\varepsilon_m}$$

$\varepsilon_m$ smaller, $\alpha_m$ larger. Base learner with high accuracy will be given higher weight.

Pseudocode: ( discrete adaboost : output is $\{-1, 1\}$ )

Given training dataset: $T = \{(x_1, y_1) \cdots (x_N, y_N)\}$, $y \in \{-1, +1\}$

① Initialize weights: $w_i^{(1)} = \frac{1}{N}$ $(i = 1, 2, \cdots, N)$

② for $m = 1$ to $M$:

(a) Train base learner with weighted train dataset:

$$G_m(x) = \arg\min_{G(x)} \sum_{i=1}^{N} w_i^{(m)} \mathbb{I}(y_i \neq G(x_i))$$

(b) Compute the error rate of $G_m(x)$:

$$\varepsilon_m = \frac{\sum_{i=1}^{N} w_i^{(m)} \mathbb{I}(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i^{(m)}}$$

(c) Compute $\alpha_m$:

$$\alpha_m = \frac{1}{2} \ln \frac{1 - \varepsilon_m}{\varepsilon_m}$$

(d) Update $w_i$:

$$w_i^{(m+1)} = \frac{w_i^{(m)} \cdot e^{-y_i \alpha_m G_m(x_i)}}{Z^{(m)}} \quad (i = 1, 2, 3, \cdots N)$$

where $Z^{(m)}$ is normalization factor,

$$Z^{(m)} = \sum_{i=1}^{N} w_i^{(m)} e^{-y_i \alpha_m G_m(x_i)}$$

③ $G(x) = \text{sign}\left[\sum_{m=1}^{M} \alpha_m G_m(x)\right]$

## Real adaboost output is the probability

① initialization : $w_i^{(1)} = \frac{1}{N}$ $(i=1,2,\cdots,N)$

② for $m=1$ to $M$ :

(a) train base learner:

$$P_m(x) = P_w(y=1|x) \in [0,1]$$

(b) $G_m(x) = \frac{1}{2}\log\frac{P_m(x)}{1-P_m(x)} \in R$

(c) Update $w_i$ :

$$w_i^{(m+1)} = \frac{w_i^{(m)} \cdot e^{-y_i G_m(x_i)}}{Z^{(m)}}$$

③ $G(x) = \mathrm{sign}[\sum_{m=1}^{M} G_m(x)]$