BOOSTING (ADABOOST)

Main idea of boosting is to turn a weak learner into a strong learner

Weak Learner:

A weak learner does slightly better than random guess, but generally not very accurate on its own. A example of a weak learner is a decision tree with only 1 separator.

Formally:

Definition 10.1 (γ-Weak-Learnability)

A learner, A, is a γ-Weak-Learner for hypothesis class $\mathcal{H}$ if:

$\exists$ function $m_{\mathcal{H}} : (0,1) \to \mathbb{N}$ s.t. for every $\begin{cases} \text{confidence } \delta \in (0,1) \\ \text{distribution } D \text{ over domain } X \\ \text{label function } f : X \to \{+1, -1\} \end{cases}$

With realizability assumption, then when running algorithm A on $m \geq m_{\mathcal{H}}(\delta)$ examples, the algorithm returns a hypothesis h such that, with probability at least $1-\delta$:

$$L_{D,f}(h) \leq \frac{1}{2} - \gamma$$

Weak hypothesis class $\mathcal{H}$:

A hypothesis class $\mathcal{H}$ is γ-weak-learnable if there exists a γ-weak-learner for that class

Comparing this definition to PAC learning's definition

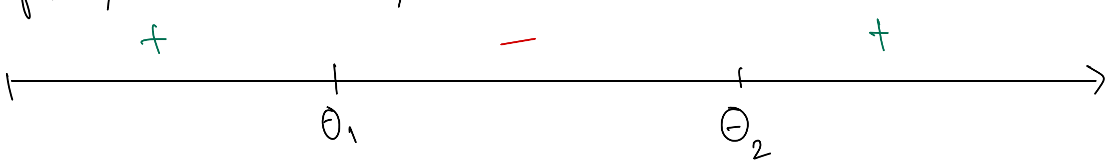
Its almost identical, PAC learning could be thought of as "strong learner", the difference is:

- PAC learning: $L_{D,f} \leq \varepsilon$ (where ε is very small)
- Weak Learer: $L_{D,f} \leq \frac{1}{2} - \gamma$ (where γ is very small)

Example 10.1 Weak Learning of 3-Piece Classifiers Using Decision Stumps

Let $\begin{cases} X \in \mathbb{R} \\ \mathcal{H} = \{ h_{\theta_1, \theta_2, b} : \theta_1 < \theta_2 , \ \theta_1, \theta_2 \in \mathbb{R} , \ b \in \{+1, -1\} \} \\ \text{where } \quad h_{\theta_1,\theta_2,b}(x) = \begin{cases} +b & \text{if } x < \theta_1 \text{ or } x > \theta_2 \\ -b & \text{if } \theta_1 \leq x \leq \theta_2 \end{cases} \end{cases}$

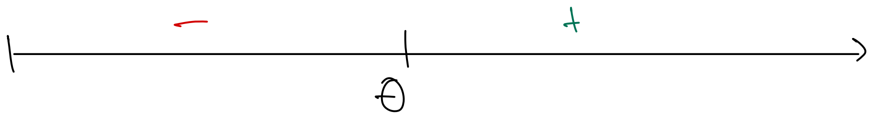
for example, if $b = +1$, then:



Let $\mathcal{B}$ be the class of decision stumps, meaning:

$$\mathcal{B} = \{ x \longmapsto \text{sign}(x - \theta) . b \; : \; \theta \in \mathbb{R} \; , b \in \{+1, -1\} \}$$

for example, if $b = +1$, then:



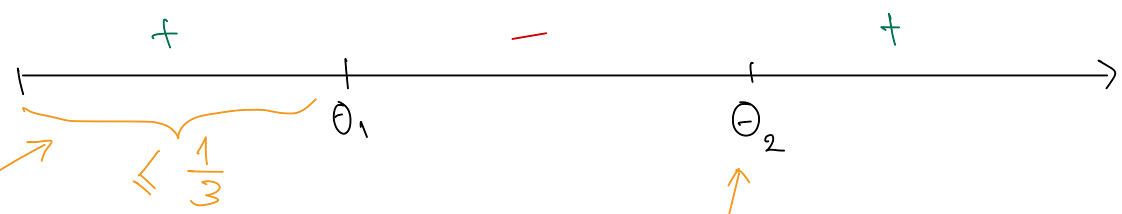
- In the following we will show that algorithm $ERM_{\mathcal{B}}$ is γ-Weak-Learner for $\mathcal{H}$, for $\gamma = 1/12$

To see that, we start by proving exists a decision stump s.t. $L_D(h) \leqslant \frac{1}{3}$.

Consider the following points:

- For every $h \in \mathcal{H}$, there are 3 regions on $\mathbb{R}$ line with alternate labels
- No matter how the line is divided, there exist at least one region where probability mass at most $\frac{1}{3}$
- A decision stump can be place to agree with the labels on 2 heavier regions, and disagree with the lightest region (probability mass $\leqslant \frac{1}{3}$)
- Let $h \in \mathcal{H}$ be a zero error hypothesis, a decision stump that <u>disagrees</u> with h must be on a region that has error at most $\frac{1}{3}$

Visually, let consider $b = +1$



pick θ_2 as decision stumps since this gives lowest empirical risk

Then $L_D(h) \leqslant \frac{1}{3}$ since error mass only happens in <u>first region</u>

So, using PAC defition, we can say with probability at least $1 - \delta$, and sample size of $m \geqslant \Omega\left(\frac{1}{\varepsilon} \log \frac{1}{\delta}\right)$, algorithm $ERM_{\mathcal{B}}$ return a hypothesis h such that:

$$L_D(h) \leqslant \frac{1}{3} + \varepsilon$$

If we set $\gamma = \varepsilon = \frac{1}{12}$, then

$$\begin{aligned} L_D(h) &\leqslant \frac{1}{3} + \varepsilon \\ &= \frac{1}{3} + \frac{1}{12} \\ &= \frac{1}{2} - \frac{1}{12} \\ &= \frac{1}{2} - \gamma \end{aligned}$$

So we can conclude $ERM_{\mathcal{B}}$ is a γ-Weak-Learner

Ada-Boost:

A natural question to ask when we have a weak learner is how to turn it to a strong learner without having to get more training data.
One way to achieve that is to use Ada-Boost algorithm

Pseudo code

input:

training set $S = (x_1, y_1), \dots, (x_m, y_m)$

weak learner WL

number of rounds T

initialize $D^{(1)} = \left(\frac{1}{m}, \dots, \frac{1}{m}\right)$ uniform distribution

for $t = 1, \dots, T$:

invoke weak learner $h_t = WL(D^{(t)}, S)$

compute $\varepsilon_t = \sum_{i=1}^{m} D_i^{(t)} \mathbb{1}_{y_i \neq h_t(x_i)}$ $\quad \varepsilon_t \overset{\text{def}}{=} L_{D^{(t)}}(h_t) \leq \frac{1}{2} - \gamma$

let $w_t = \frac{1}{2} \log\left(\frac{1}{\varepsilon_t} - 1\right)$ weight of h_t, inversely proportional to ε_t

update $D_i^{(t+1)} = \dfrac{D_i^{(t)} \exp(-w_t y_i h_t(x_i))}{\sum_{j=1}^{m} D_j^{(t)} \exp(-w_t y_j h_t(x_j))}$ for all $i = 1, \dots, m$

$\rightarrow$ divide this to normalize

output the hypothesis $h_s(x) = \text{sign}\left(\sum_{t=1}^{T} w_t h_t(x)\right)$

Intuition:

At each iteration, $D_i^{(t+1)}$, the probability mass of i^{th} example, x_i, is updated, such that the mass $\begin{cases} \text{increase if } h_t(x_i) \neq y_i \\ \text{decrease if } h_t(x_i) = y_i \end{cases}$

This will force the learner to focus on the misclassified examples next iteration.

How fast training error decrease?

Since $\varepsilon_t = \sum_{i=1}^{m} D_i^{(t)} \mathbb{1}_{h_t(x) \neq y_i}$

always ≥ 0 and < 1 hence decreasing

$$= \sum_{i=1}^{m} \boxed{\frac{D_i^{(t-1)} \exp(-w_{t-1} y_i h_{t-1}(x_i))}{\sum_{j=1}^{m} D_j^{(t-1)} \exp(-w_{t-1} y_j h_{t-1}(x_j))}} \cdot \mathbb{1}_{h_t(x) \neq y_i}$$

So ε_t decrease exponentially with the number of boosting rounds.

How good is the output hypothesis resulted from Ada-Boost?

Theorem 10.2 will answer this question

Theorem 10.2 Upper bound of h_S, output of Ada-Boost algorithm

Let S be a training set and assume that at each iteration of AdaBoost, the weak learner returns a hypothesis for which $\varepsilon_t \leqslant \frac{1}{2} - \gamma$. Then, the training error of the output hypothesis of AdaBoost is at most:

$$L_S(h_S) = \frac{1}{m} \sum_{i=1}^{m} 1_{h_S(x_i) \neq y_i} \leqslant \exp(-2\gamma^2 T)$$

Proof:

For each iteration t, denotes

- $f_t(x) = \overbrace{\sum_{p \leqslant t} w_p h_p(x)}^{\text{weighted sum of weak learners up until } t}$

  so f_T is the output hypothesis of Ada Boost
- $Z_t = \frac{1}{m} \sum_{i=1}^{m} \underbrace{\exp(-y_i f_t(x_i))}_{= \exp\left(-w_p y_i h_p(x_i)\right)}$

exponential loss function

For any hypothesis h, we know this inequality always holds

$$1_{h(x) \neq y} \leqslant \exp(-y h(x)) \qquad \left\langle \begin{array}{l} 1_{h(x) \neq y} \in \{0, 1\} \\ \exp(-y h(x)) \geqslant 0 \end{array} \right\rangle$$

$$\Leftrightarrow \quad \frac{1}{m} \sum_{i=1}^{m} 1_{h(x_i) \neq y_i} \leqslant \frac{1}{m} \sum_{i=1}^{m} \exp\left(-y_i h(x_i)\right)$$

Replacing $f_T(x) = \sum_T w \cdot h(x)$:

$$\Leftrightarrow \quad \frac{1}{m} \sum_{i=1}^{m} 1_{f_T(x_i) \neq y_i} \leqslant \frac{1}{m} \sum_{i=1}^{m} \exp(-y_i f_T(x_i))$$

$$\Leftrightarrow \quad L_S(f_T) \leqslant Z_T \qquad \text{Inequality 1}$$

Now we try to upper bound Z_T, consider this fact:

$f_0 \equiv 0$ < initial hypothesis of AdaBoost should not make any predictions, hence always returns 0 >

$\Rightarrow \quad Z_0 = 1$

From the above fact, we can rewrite Z_T as:

$$Z_T = \frac{Z_T}{Z_0} = \frac{Z_T}{Z_{T-1}} \cdot \frac{Z_{T-1}}{Z_{T-2}} \cdots \frac{Z_2}{Z_1} \cdot \frac{Z_1}{Z_0} \qquad \langle \text{telescopic series} \rangle$$

$$= \prod_{i=0}^{T-1} \frac{Z_{t+1}}{Z_t} \qquad \text{Equation 1}$$

Now if we can upper bound $\frac{Z_{t+1}}{Z_t} \leqslant \exp(-2\gamma^2)$ then we are done!

From our denotion above:

$$\frac{Z_{t+1}}{Z_t} = \frac{\sum_{i=1}^{m} \exp(-y_i f_{t+1}(x_i))}{\sum_{j=1}^{m} \exp(-y_j f_t(x_j))}$$

$$= \frac{\sum_i \exp(-y_i f_t(x_i)) \cdot \exp(-y_i w_{t+1} h_{t+1}(x_i))}{\sum_j \exp(-y_j f_t(x_j))} \quad \rightarrow D_i^{(t+1)}$$

$$= \sum_i D_i^{(t+1)} \cdot \exp(-y_i w_{t+1} h_{t+1}(x_i))$$

$$= \exp(-w_{t+1}) \underbrace{\sum_{i:\, y_i h_{t+1}(x_i) = 1} D_i^{(t+1)}}_{\text{correctly classified examples}} + \exp(w_{t+1}) \underbrace{\sum_{i:\, y_i h_{t+1}(x_i) = -1} D_i^{(t+1)}}_{\text{incorrectly classified examples}}$$

$$= \exp(-w_{t+1})(1 - \varepsilon_{t+1}) + \exp(w_{t+1}) \cdot \varepsilon_{t+1}$$

Replacing $w_{t+1} = \frac{1}{2}\log\left(\frac{1}{\varepsilon_{t+1}} - 1\right)$:

$$= \frac{1}{\sqrt{1/\varepsilon_{t+1} - 1}}(1 - \varepsilon_{t+1}) + \sqrt{1/\varepsilon_{t+1} - 1} \cdot \varepsilon_{t+1}$$

$$= \sqrt{\frac{\varepsilon_{t+1}}{1 - \varepsilon_{t+1}}}(1 - \varepsilon_{t+1}) + \sqrt{\frac{1 - \varepsilon_{t+1}}{\varepsilon_{t+1}}} \cdot \varepsilon_{t+1}$$

$$= 2\sqrt{\varepsilon_{t+1}(1 - \varepsilon_{t+1})}$$

Since $\begin{cases} \varepsilon_{t+1} \leq \frac{1}{2} - \gamma & \langle \text{assume } h_{t+1} \text{ is a } \gamma\text{-weak-learner} \rangle \\ \text{function } g(\varepsilon) = \varepsilon(1 - \varepsilon) \text{ monotonically increasing in } [0, \frac{1}{2}] \end{cases}$

We can say that:

$$2\sqrt{\varepsilon_{t+1}(1 - \varepsilon_{t+1})} \leq 2\sqrt{\left(\frac{1}{2} - \gamma\right)\left(\frac{1}{2} + \gamma\right)}$$

$$= \sqrt{1 - 4\gamma^2}$$

$$\leq \sqrt{\exp(-4\gamma^2)} \qquad \langle 1 - a \leq \exp(-a) \rangle$$

$$= \exp\left(\frac{1}{2} \cdot -4\gamma^2\right)$$

$$= \exp(-2\gamma^2)$$

Combine with Equation 1:

$$Z_T = \prod_{i=0}^{T-1} \frac{Z_{t+1}}{Z_t} \leq \prod_{i=0}^{T-1} \exp(-2\gamma^2)$$

$$= \exp(-2\gamma^2 T)$$

And since $L_S(f_T) \leq Z_T$ (inequality 1), we conclude that:

$$L_S(f_T) \leq \exp(-2\gamma^2 T)$$

Remark 10.2 What if the weak-learner fails?

- At every iteration of AdaBoost, the weak-learner outputs a hypothesis with error at most $\frac{1}{2} - \gamma$, but we've learned that the learner can fails with probability at most δ

$$P[\text{weak learner fails}] \leq \delta$$

$$\bigcup_{i=1}^{T} P[\text{weak learner fails}] \leq \delta T$$

$$\Rightarrow \quad P[\text{weak learner won't fails at all}] \geq 1 - \delta T$$

- We can prove later in Exercise 1 that the dependence of sample complexity m on δ can always be $\log(\frac{1}{\delta})$, and therefore invoking weak learner with very small δ is not problematic, therefore we can assume δT is also small.
- Furthermore, since the weak learner is applied with distributions over the training set, therefore we can implement the weak learner so that it will have zero probability of failure ($\delta = 0$)