MA4270 Lecture 13 Theory of Boosting

Ada Boost $y_t \in \{\pm 1\}$ $\underline{x}_t \in \mathbb{R}^d$

Input: $\mathcal{D}_n = \{(\underline{x}_t, y_t)\}_{t=1}^n$ m: total # of boosting iteration

- Initialize weight $W_0(t) = \frac{1}{n}$ $t=1,2,\dots,n$

- At boosting stage $m \geq 1$. find a base learn $h(\cdot\,; \hat{\theta}_m)$

S, K, θ_0

we can use other Base Learner as well!

that minimizes

$$\underline{\hat{\theta}}_m = \underset{\underline{\theta}_m}{\operatorname{argmin}} \underbrace{- \sum_{t=1}^n \tilde{W}_{m-1}(t)\, y_t\, h(\underline{x}_t ; \theta_m)}_{\text{weighted training error}}$$

probability distribution

- Choose $\hat{\alpha}_m \longrightarrow \hat{\alpha}_m = \frac{1}{2} \ln\left(\frac{1-\hat{\varepsilon}_m}{\hat{\varepsilon}_m}\right)$

- Update $$\tilde{W}_m(t) = \begin{cases} \dfrac{\tilde{W}_{m-1}(t)}{Z_m} e^{-\hat{\alpha}_m}, & \text{classify correctly} \\ \dfrac{\tilde{W}_{m-1}(t)}{Z_m} e^{\hat{\alpha}_m}, & \text{classify wrongly} \end{cases}$$

- Go to minimize Weighted Training Error To attain $\underline{\hat{\theta}}_{m+1}$

$\Downarrow$

$h(\cdot\,; \hat{\theta}_{m+1})$

Thm: If each base learner is slightly better than RANDOM GUESSING, i.e., $\hat{\varepsilon}_j < \frac{1}{2} \Rightarrow$ weighted trainging Error

Then We have Training Error Of $\boxed{\text{Ensemble Learner}}$ decrease to 0! (exponentially fast) //

$$h_m(x) = \sum_{j=1}^{m} \hat{\alpha}_j h(x; \hat{\theta}_j)$$

$$\frac{1}{n}\sum_{t=1}^{n} \mathbb{1}\{y_t h_m(x_t) \le 0\} \le \exp\left\{-2\sum_{j=1}^{m}\left(\tfrac{1}{2}-\hat{\varepsilon}_j\right)^2\right\} \quad \forall m$$

Weighted Training Error

$$\hat{\varepsilon}_j = \sum_{t=1}^{n} \tilde{W}_{j-1}(t)\, \mathbb{1}\{y_t \ne h(x_t, \hat{\theta}_j)\}$$

$\le \exp\{-2m\gamma^2\}$ if $\frac{1}{2} - \hat{\varepsilon}_j \ge \gamma$

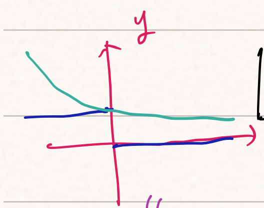
uniform Bound !

Objective of choosing $\hat{\theta}_j = \operatorname{argmin} \underline{2\varepsilon_j - 1}$

$\Leftrightarrow \operatorname{argmin} \varepsilon_j$

Pf: $\underline{\frac{1}{n}\sum \mathbb{1}\{y_t h_m(x_t) \le 0\} \le \exp\{-2\sum(\frac{1}{2}-\hat{\varepsilon}_j)^2\}}$

$z_t = y_t h_m(x_t)$



$\boxed{\text{Step 1}}$: $\frac{1}{n}\sum \mathbb{1}\{y_t h_m(x_t) \le 0\} \le \frac{1}{n}\sum_{t=1}^{n} \exp\{-y_t h_m(x_t)\}$

$\Downarrow$

exp loss is an upper Bound of 0-1 loss ! (convex & Smooth)

Can be very small !

$\boxed{\text{Step 2}}$: Claim: $\frac{1}{n}\sum \exp(-y_t h_m(x_t)) = \boxed{\prod_{j=1}^{m} Z_j}$

Where $\underline{Z_j} = \sum \tilde{W}_{j-1} \exp(-y_t \hat{\alpha}_t h(x_t; \hat{\theta}_j))$

$\hookrightarrow$ the normalization Factor

Pf: Note that $\tilde{W}_0(t) = W_0(t) = \frac{1}{n} \quad \forall t$

$$\tilde{W}_1(t) = \begin{cases} \dfrac{\tilde{W}_0(t)\exp\{\alpha_1\}}{Z_1} \\[2ex] \dfrac{\tilde{W}_0(t)\exp\{-\alpha_1\}}{Z_1} \end{cases}$$

$$\Leftrightarrow \tilde{W}_1(t) = \frac{1}{Z_1}\tilde{W}_0(t) \exp\{-\alpha_1 y_t h(x_t, \hat{\theta}_1)\}$$

$$\widetilde{W}_2(t) = \frac{1}{Z_2} \widetilde{W}_1(t) \exp\{-\alpha_2 y_t h(x_t ; \hat{\theta}_2)\}$$

$$= \frac{1}{n} \frac{1}{Z_1 Z_2} \exp\{-y_t [\alpha_1 h(x_t; \theta_1) + \alpha_2 h(x_t; \theta_2)]\}$$

$$\Rightarrow \widetilde{W}_m(t) = \frac{1}{n} \frac{1}{\prod Z_i} \exp\{-y_t h_m(x_t)\}$$

$$\left(\sum_{t=1}^{n} \widetilde{W}_m(t) = 1 \right)$$

$$\Rightarrow \prod_{i=1}^{m} Z_i = \frac{1}{n} \sum_{t=1}^{n} \exp\{-y_t h_m(x_t)\} \quad \square$$

$\boxed{\text{Seep 3}}$ <u>If $Z_j < 1$, training Error Decays Exp. FAST</u>

<u>Pf</u>:

$$Z_m = \sum_{t=1}^{n} \widetilde{W}_{m-1}(t) \exp(-\hat{\alpha}_m y_t h(x_t; \hat{\theta}_m))$$

$$= e^{-\hat{\alpha}_m} \sum \widetilde{W}_{m-1}(t) \cdot \mathbb{1}\{y_t = h(x_t, \hat{\theta}_m)\}$$

$$+ e^{\hat{\alpha}_m} \sum \widetilde{W}_{m-1}(t) \, \mathbb{1}\{y_t \neq h(x_t; \hat{\theta}_m)\}$$

In order to attain Training Error as small as possible, we minimize Z_m!

$$= e^{-\hat{\alpha}_m} \cdot (1 - \hat{\varepsilon}_m) + e^{\hat{\alpha}_m} \hat{\varepsilon}_m$$

Notice that Z_m is convex in $\hat{\alpha}_m$

$$\Rightarrow \frac{dZ_m}{d\hat{\alpha}_m} = 0 \Rightarrow e^{-\hat{\alpha}_m}(\hat{\varepsilon}_m - 1) + e^{\hat{\alpha}_m} \varepsilon_m = 0$$

$$\Rightarrow \hat{\alpha}_m = \frac{1}{2} \ln\left(\frac{1-\hat{\varepsilon}_m}{\hat{\varepsilon}_m}\right)$$

show the choice of $\hat{\alpha}_m$.

$\boxed{\text{Seep 4}}$: Substitute $\hat{\alpha}_m$ to Z_m !

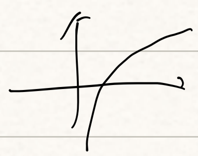
$$Z_m = e^{-\hat{\alpha}_m}(1-\hat{\varepsilon}_m) + e^{\hat{\alpha}_m}\hat{\varepsilon}_m \Bigg|_{\hat{\alpha}_m = \frac{1}{2}\ln\left(\frac{1-\hat{\varepsilon}_m}{\hat{\varepsilon}_m}\right)}$$

$$= \left(\frac{1-\hat{\varepsilon}_m}{\hat{\varepsilon}_m}\right)^{-\frac{1}{2}}(1-\hat{\varepsilon}_m) + \left(\frac{1-\hat{\varepsilon}_m}{\hat{\varepsilon}_m}\right)^{\frac{1}{2}}\hat{\varepsilon}_m$$

$$= 2\sqrt{\hat{\varepsilon}_m(1-\hat{\varepsilon}_m)}$$

Putting Everything together: Training Error $\leq \prod_{i=1}^{m} Z_i$

Best Possible $\hat{\alpha}_m$



$\ln u < u-1$

$\boxed{\left(\text{for the choice of } \hat{\alpha}_m\right)}$ $= \prod_{i=1}^{m} 2\sqrt{\hat{\varepsilon}_i(1-\hat{\varepsilon}_i)}$

$(1-u)^{\frac{1}{2}} = \exp\left(\ln(1-u)^{\frac{1}{2}}\right)$

$= \exp\left(\frac{1}{2}\ln(1-u)\right) \Rightarrow$

$\ln u < u-1$

$\Leftrightarrow \ln(1-u) < -u$ $\Rightarrow$ $< \exp\left(-\frac{1}{2}u\right)$

$$= \prod_{i=1}^{m}\sqrt{1-(1-2\hat{\varepsilon}_i)^2}$$

$$\leq \prod_{i=1}^{m}\exp\left(-\frac{1}{2}(1-2\hat{\varepsilon}_i)^2\right)$$

$$= \exp\left\{-\frac{1}{2}\sum_{i=1}^{m}(1-2\hat{\varepsilon}_i)^2\right\}$$

$$= \exp\left\{-2\sum_{i=1}^{m}\left(\frac{1}{2}-\hat{\varepsilon}_i\right)^2\right\}$$

Counter Example : (Decision Stump CANNOT Do very well)

$\Rightarrow \hat{\varepsilon}_j = \frac{1}{2}$ for $\forall j$

since the weight will never update.

Other types of Error

① Weighted Error Relative to Update weights. $\boxed{\tilde{W}_m(t)}$

Claim: $\sum_{t=1}^{n} \tilde{W}_m(t)\, \mathbb{1}\{y_t \neq h(x_t;\hat{\theta}_m)\} = \frac{1}{2}$

Rmk: The Base Learner $\hat{\theta}_m$ $(h(\cdot;\hat{\theta}_m))$ is useless in the next boosting iteration.

only NEXT ONE

since $h(\cdot;\hat{\theta}_m)$ is attain by:

$\hat{\theta}_m = \arg\min_{\theta} \sum \hat{W}_{m-1}(t)\, \mathbb{1}\{y \neq h(x_t;\theta)\}$

$\Downarrow$

Pf: $2\,\mathbb{1}\{y_t \neq h(x_t;\hat{\theta}_m)\} - 1 = -y_t\, h(x_t;\hat{\theta}_m)$

$\Rightarrow$ Our conclusion $\Leftrightarrow$ $\underbrace{\sum \hat{W}_m(t)\, y_t h(x_t;\hat{\theta}_m)}_{\text{weighted agreement}} = 0$

$\hat{\theta}_{m+1} \neq \hat{\theta}_m$

$\Downarrow$

choose a different decision stump.

However, it only guarantees that we don't choose the same decision stump IN TWO CONSECUTIVE ITERATION!

$\Downarrow$

10th decision stump can be the same with 20th decision stump.

$= \sum_{\text{correct}} \tilde{W}_m(t)\, y_t h(x_t;\hat{\theta}_m)$

$+ \sum_{\text{wrongly}} \hat{W}_m(t)\, y_t h(x_t;\hat{\theta}_m)$

$= \sum_{\text{correct}} \frac{\tilde{W}_{m-1} e^{-\alpha_m} y_t h(x_t;\hat{\theta}_m)}{Z_m}$

$$+ \sum_{\text{wrongly}} \frac{\tilde{W}_{m-1}(t)\, e^{\alpha_m} y_t\, h(x_t; \hat{\theta}_m)}{Z_m}$$

$$= \frac{1}{Z_m} \left(\sum_{\text{correct}} e^{-\alpha_m} \tilde{W}_{m-1}(t) - \sum_{\text{wrongly}} e^{\alpha_m} \tilde{W}_{m-1}(t) \right)$$

$$= \frac{1}{Z_m} \left(e^{-\alpha_m} (1 - \hat{\varepsilon}_m) - e^{\alpha_m} \hat{\varepsilon}_m \right) = 0$$

$\frac{d Z_m}{d\alpha_m} = 0 \Rightarrow$ choose Best Possible $\hat{\alpha}_m$ for Z_m

#

EMPHASIZE THE SEQUENCE

$$\{\tilde{W}_{m-1}(t)\}_{t=1}^{n}$$

$$\Downarrow$$

$$\hat{\theta}_m \rightarrow h(x; \hat{\theta}_m)$$

$$\Downarrow$$

$$\hat{\varepsilon}_m = \sum \tilde{W}_{m-1}(t)\, \mathbb{1}\{y_t \neq h(x_t; \hat{\theta}_m)\}$$

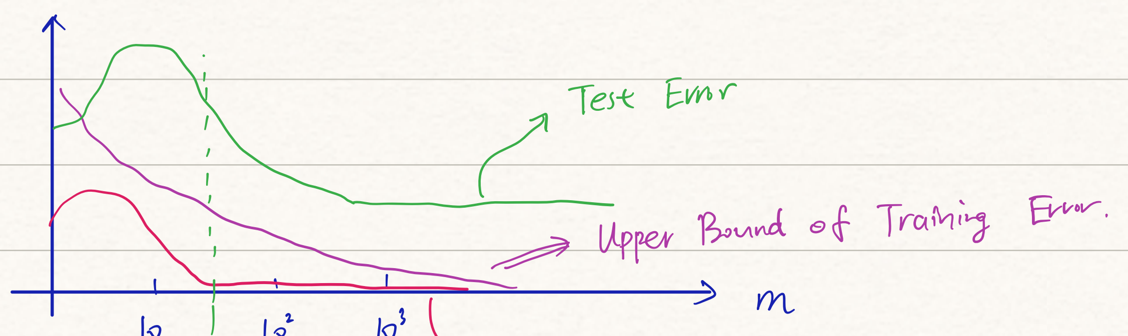
$$\Downarrow$$

$$\hat{\alpha}_m = \ln\left(\frac{1 - \hat{\varepsilon}_m}{\hat{\varepsilon}_m}\right)$$

$$\Downarrow$$

$$\{\tilde{W}_m(t)\}_{t=1}^{n} = \frac{1}{Z_m} \begin{cases} e^{\alpha_m}, & \text{wrong} \\ e^{-\alpha_m}, & \text{correct} \end{cases}$$

Test Error



# of iteration / Boosting Rounds.

Training Error

Rmk: 1. Test error continue ↓ even after training error = 0

2. Test Error Does NOT ↑ even after many Boosting iteration! ⇒ Very Robust!

⇓

Simulation! → 1 vs 7/8 Pattern Recognition

Heuristic Explanation: $\boxed{(\text{for } ①)}$ Normalized Ensemble Classifier

$$\tilde{h}_m(x) = \frac{\sum \hat{\alpha}_j h(x; \hat{\theta}_j)}{\sum \alpha_i} \in [-1, 1]$$

Define Voting Margin for each t:

↓

$$\text{margin}(t) := y_t \tilde{h}_m(x_t) \in [-1, 1]$$

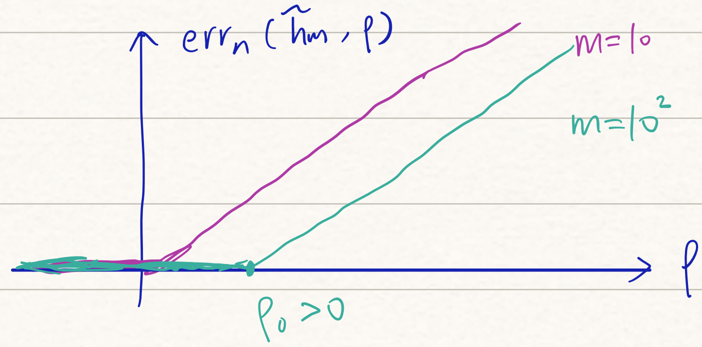
Rmk: $\text{margin}(t) > 0 \Leftrightarrow$ example $\vec{x}_t$ is classified correctly by $h_m(\cdot)$ ($\tilde{h}_m(\cdot)$)

Def: ρ - training error: $\text{err}_n(\tilde{h}_m; \rho) = \frac{1}{n} \sum_{t=1}^{n} \mathbb{1}\{\text{margin}(t) \leq \rho\}$

Rmk: Training Error = $\text{err}_n(\tilde{h}_m, 0)$

From traing error theorem: $\text{err}_n(\tilde{h}_m; 0) = 0$ after finitely m

But even after $err_n(\tilde{h}_m, 0) = 0$, boosting still decreases $err_n(\tilde{h}_m, \rho)$ for some $\rho > 0$



Why? Because we are implicitly minimize the 0-1 loss!

$\Downarrow$

Training Error Thm:

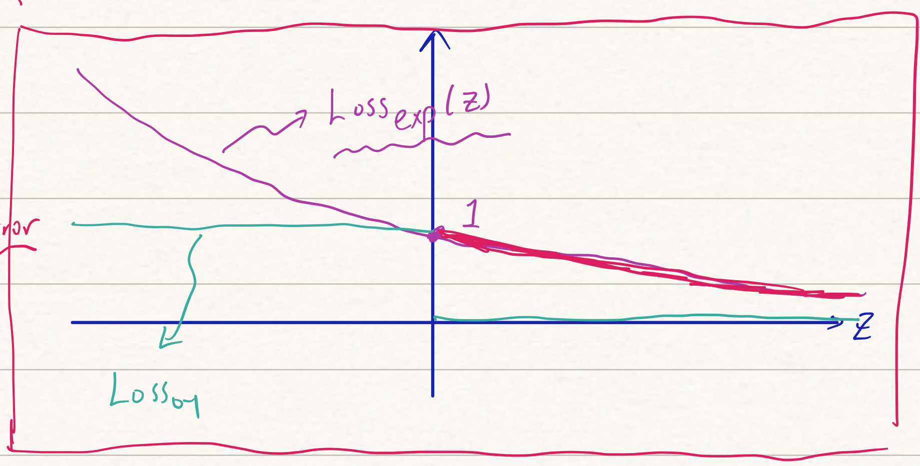
we actually minimize this!

0-1 loss $\leq$ exp. loss $= \exp(-\text{margin}(t))$

☆ When we do update, we actually minimize $Loss_{exp}$ (exp. Error)

$\Downarrow$ 0-1 Training Error decrease

$\Downarrow$ ρ-Training Error decrease



Resistance to overfitting

1. "Complexity" of ensemble classifier h_m $\nearrow$ rapidly with m
2. Sequential optimizing instead of Joint (Greedy)

Define: Expected Risk.

$$R(f) = \mathbb{E}\left[\text{loss}_{0\text{-}1}(y, f(x))\right] = \Pr(y \neq f(x))$$

Empirical Risk

$$\hat{R}_n(f) = \frac{1}{n}\sum_{t=1}^{n} \text{Loss}_{0\text{-}1}(y_t, f(x_t))$$

<u>Thm</u>: Freund & Schapire (1995)

With prob $\geq 1-\delta$, $R(h_m) \leq \hat{R}_n(h_m) + \underbrace{\tilde{O}\left(\sqrt{\frac{md}{n}}\right)}_{\Downarrow \text{ very loose}}$

according to m

- m: # of boosting rounds
- n: # of training sample
- d: VC dimension of base learner

→ a measure of complexity.

Rmk: This Bound is not predictive of Good (Robust) test error because of its dependence on m!

⇓

Not Good Thm!

☆☆☆

<u>Thm</u>: With Prob $\geq 1-\delta$, $R(h_m) \leq \text{err}_n(h_m; \rho) + \tilde{O}\left(\sqrt{\frac{d}{n\rho^2}}\right)$

Not related to m!

fixed with m grows?

$$\text{err}_n(h_m;\rho) = \frac{1}{n}\sum_{t=1}^{n} \mathbb{1}\{y_t \tilde{h}_m(x_t) \leq \rho\}$$

$\Rightarrow$ if we choose $\forall \rho < \gamma = \min\left(\frac{1}{2} - \varepsilon_j\right)$

then $err_n(\tilde{h}_m, \rho) \to 0$ expentially fast !

☆

Rmk: Essentially this bound says that there is no more dependence of the Expected Risk on m

$\Downarrow$

exp. Loss

implies that Adaboost (Boosting is Robust to overfitting)