Boosting From Different Perspectives

Navid Ardeshir

Department of Statistics
Columbia University

May 2020

Outline

Outline

Review

Setting Up some Notation:

Data $\mathcal{S} = \{(X_i, Y_i) \in \mathcal{X} \times \{\pm 1\} : 1 \leq i \leq n\}$ represents learner's observed data where X is generated from an unknown distribution $\mathcal{D}$ and $Y = f(X)$ for some mapping $f : \mathcal{X} \mapsto \{\pm 1\}$.

$X =$  , $Y = 1$

Review

Setting Up some Notation:

Data $\mathcal{S} = \{(X_i, Y_i) \in \mathcal{X} \times \{\pm 1\} : 1 \leq i \leq n\}$ represents learner's observed data where X is generated from an unknown distribution $\mathcal{D}$ and $Y = f(X)$ for some mapping $f : \mathcal{X} \mapsto \{\pm 1\}$.

$X =$  $, \ Y = 1$

Output Prediction rule from hypothesis class $\mathcal{H}$ which contains certain mappings from $\mathcal{X}$ into $\{\pm 1\}$. For instance, truncated linear functions $\{x \mapsto \text{sign}(\langle a, x \rangle) \text{ for } a \in \mathbb{R}^d\}$

Review

Setting Up some Notation:

- Accuracy can be measured by $L_{\mathcal{D}}(h) = \mathbb{P}[h(X) \neq Y]$ which is the true error rate of a hypothesis $h \in \mathcal{H}$. Goal of the learner is try to minimize this.

Review

Setting Up some Notation:

- Accuracy can be measured by $L_{\mathcal{D}}(h) = \mathbb{P}[h(X) \neq Y]$ which is the true error rate of a hypothesis $h \in \mathcal{H}$. Goal of the learner is try to minimize this.
  - Learner does not have enough information to compute the loss!

Review

Setting Up some Notation:

- Accuracy can be measured by $L_{\mathcal{D}}(h) = \mathbb{P}[h(X) \neq Y]$ which is the true error rate of a hypothesis $h \in \mathcal{H}$. Goal of the learner is try to minimize this.
  - Learner does not have enough information to compute the loss!
  - Instead, estimates it in the most natural way and minimizes that (considering it's computationally feasible). This is called expected risk minimization (ERM):

$$L_S(h) = \mathbb{P}_S[h(X) \neq Y] = \frac{|\{i \in [n] : h(X_i \neq Y_i)\}|}{n}$$

Review

Setting Up some Notation:

- Accuracy can be measured by $L_{\mathcal{D}}(h) = \mathbb{P}[h(X) \neq Y]$ which is the true error rate of a hypothesis $h \in \mathcal{H}$. Goal of the learner is try to minimize this.
  - Learner does not have enough information to compute the loss!
  - Instead, estimates it in the most natural way and minimizes that (considering it's computationally feasible). This is called expected risk minimization (ERM):

$$L_S(h) = \mathbb{P}_S[h(X) \neq Y] = \frac{|\{i \in [n] : h(X_i \neq Y_i)\}|}{n}$$

Intuition Law of Large Numbers ensures that the estimate is close to the true rate for large enough number of samples.

Outline

Is LLN Enough?

- Suppose $\hat{h} \in \arg\min_{h \in \mathcal{H}} L_{\mathcal{S}}(h)$. We want $L_{\mathcal{D}}(\hat{h})$ to be small and close to optimum. It is enough to control $\sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)|$:

$$\begin{aligned} L_{\mathcal{D}}(\hat{h}) &\leq L_{\mathcal{S}}(\hat{h}) + \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \\ &\leq L_{\mathcal{S}}(h^*) + \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \\ &\leq L_{\mathcal{D}}(h^*) + 2 \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \end{aligned}$$

- Note that $L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{h(X_i) \neq Y_i\}} - \mathbb{P}_{\mathcal{D}}[h(X) \neq Y]$

Is LLN Enough?

- Suppose $\hat{h} \in \arg\min_{h \in \mathcal{H}} L_{\mathcal{S}}(h)$. We want $L_{\mathcal{D}}(\hat{h})$ to be small and close to optimum. It is enough to control $\sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)|$:

$$\begin{aligned} L_{\mathcal{D}}(\hat{h}) &\leq L_{\mathcal{S}}(\hat{h}) + \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \\ &\leq L_{\mathcal{S}}(h^*) + \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \\ &\leq L_{\mathcal{D}}(h^*) + 2 \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \end{aligned}$$

- Note that $L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{h(X_i) \neq Y_i\}} - \mathbb{P}_{\mathcal{D}}[h(X) \neq Y]$
- Chernoff's Inequality controls argument this difference but we have a sup. This is where Empirical Process Theory kicks in!

Is LLN Enough?

- Suppose $\hat{h} \in \arg\min_{h \in \mathcal{H}} L_{\mathcal{S}}(h)$. We want $L_{\mathcal{D}}(\hat{h})$ to be small and close to optimum. It is enough to control $\sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)|$:

$$
\begin{aligned}
L_{\mathcal{D}}(\hat{h}) &\leq L_{\mathcal{S}}(\hat{h}) + \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \\
&\leq L_{\mathcal{S}}(h^*) + \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)| \\
&\leq L_{\mathcal{D}}(h^*) + 2 \sup_{h \in \mathcal{H}} |L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h)|
\end{aligned}
$$

- Note that $L_{\mathcal{S}}(h) - L_{\mathcal{D}}(h) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{h(X_i) \neq Y_i\}} - \mathbb{P}_{\mathcal{D}}[h(X) \neq Y]$
- Chernoff's Inequality controls argument this difference but we have a sup. This is where Empirical Process Theory kicks in!
- Some sort of Uniform Law of Large Number is required...

Chernoff-Hoeffding Bound

Theorem

Let $(Z_i)_{1\leq i\leq n} \in \{0,1\}^n$ be the result of n trials of random coin tossing. Then we have the follwing concentration inequality:

$$\mathbb{P}[|\frac{1}{n}\sum_{i=1}^{n} Z_i - \mathbb{E}[Z_1]| \geq \epsilon] \leq 2e^{-2n\epsilon^2}$$

Remark

The tail bound is asymptotically sharp due to Central Limit Theorem since tail of a gaussian decays exponentially quadratic.

Proof

Let $p = \mathbb{E}[Z_1]$. Using Markov's Inequality $\mathbb{P}[X \geq \alpha] \leq \alpha^{-1}\mathbb{E}[X]$ for a positive random variable X:

$$\begin{aligned}
\mathbb{P}[\frac{1}{n}\sum_{i=1}^{n} Z_i - \mathbb{E}[Z_1] \geq \epsilon] &= \mathbb{P}[e^{\lambda(\sum_{i=1}^{n} Z_i - n\mathbb{E}[Z_1])} \geq e^{n\lambda\epsilon}] \\
\text{(Markov'x Inequality)} &\leq e^{-n\lambda\epsilon}\mathbb{E}[e^{\lambda(\sum_{i=1}^{n} Z_i - n\mathbb{E}[Z_1])}] \\
\text{(By Independence)} &= e^{-n\lambda\epsilon}(\mathbb{E}[e^{\lambda(Z_1 - \mathbb{E}[Z_1])}])^n \\
&= e^{-n\lambda\epsilon}(pe^{\lambda(1-p)} + (1-p)e^{-\lambda p})^n \\
&= e^{-n\lambda\epsilon - n\lambda p + n\log(1-p+pe^{\lambda})} \\
\text{(Hoeffsing's Lemma)} &\leq e^{-n\lambda\epsilon + n\frac{\lambda^2}{8}} \\
\text{(Optmizie over } \lambda \geq 0\text{)} &= e^{-2n\epsilon^2}
\end{aligned}$$

Outline

Notions of Learnability

Probably Approximately Correct (PAC) Learnability

A hypothesis class $\mathcal{H}$ is called PAC learnable if for every $\epsilon, \delta, \mathcal{D}$, and f which satisfies realizibility assumption provided with enough number of samples (polynomial function of $1/\epsilon, 1/\delta$) learner can return hypothesis $h \in \mathcal{H}$ such that $L_{\mathcal{D}}(h) \leq \epsilon$ holds with probability at least $1 - \delta$.

Notions of Learnability

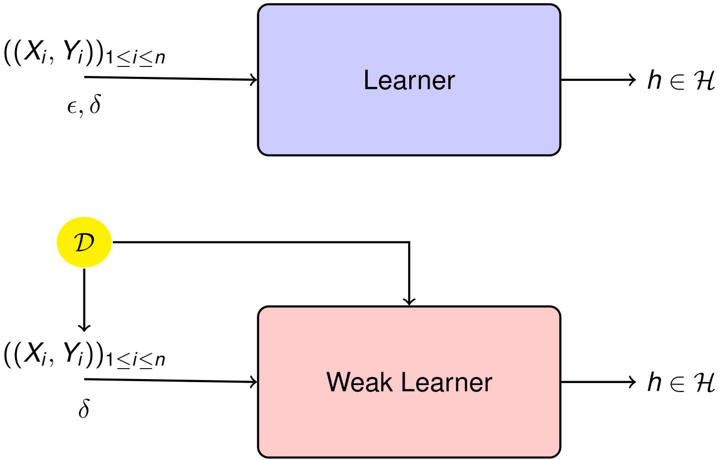
Probably Approximately Correct (PAC) Learnability

A hypothesis class $\mathcal{H}$ is called PAC learnable if for every $\epsilon, \delta, \mathcal{D}$, and f which satisfies realizibility assumption provided with enough number of samples (polynomial function of $1/\epsilon, 1/\delta$) learner can return hypothesis $h \in \mathcal{H}$ such that $L_{\mathcal{D}}(h) \leq \epsilon$ holds with probability at least $1 - \delta$.

γ-Weak-Learnability

A hypothesis class $\mathcal{H}$ is called γ-Weak-learnable if for every $\delta, \mathcal{D}$, and f which satisfies realizibility assumption provided with enough number of samples (polynomial function of $1/\delta$) learner can return hypothesis $h \in \mathcal{H}$ such that $L_{\mathcal{D}}(h) \leq 1/2 - \gamma$ holds with probability at least $1 - \delta$.

Notions of Learnability



Outline

Is PAC Learning Stronger Than Weak Learning?

- Suppose hypothesis class $\mathcal{H}$ is γ Weak learnable. Denote, $A = [Y_i h(X_I)]_{i,h}$ then for every $p \in \triangle([n])$ there exists $h \in \mathcal{H}$ such that:

$$\sum_{i=1}^{n} p_i \mathbf{1}_{\{h(X_i) \neq Y_i\}} \leq \frac{1}{2} - \gamma$$

Is PAC Learning Stronger Than Weak Learning?

- Suppose hypothesis class $\mathcal{H}$ is γ Weak learnable. Denote, $A = [Y_i h(X_I)]_{i,h}$ then for every $p \in \triangle([n])$ there exists $h \in \mathcal{H}$ such that:

$$\sum_{i=1}^{n} p_i \mathbf{1}_{\{h(X_i) \neq Y_i\}} \leq \frac{1}{2} - \gamma$$

$$\Updownarrow$$

$$p^\top A e_h = \sum_{i=1}^{n} p_i Y_i h(X_i) \geq 2\gamma$$

Is PAC Learning Stronger Than Weak Learning?

- Suppose hypothesis class $\mathcal{H}$ is γ Weak learnable. Denote, $A = [Y_i h(X_l)]_{i,h}$ then for every $p \in \triangle([n])$ there exists $h \in \mathcal{H}$ such that:

$$\sum_{i=1}^{n} p_i \mathbf{1}_{\{h(X_i) \neq Y_i\}} \leq \frac{1}{2} - \gamma$$

$$\Updownarrow$$

$$p^\top A e_h = \sum_{i=1}^{n} p_i Y_i h(X_i) \geq 2\gamma$$

$$\Updownarrow$$

$$\boxed{\min_{p \in \triangle([n])} \max_{h \in \mathcal{H}} p^\top A e_h \geq 2\gamma}$$

Existence of an Ideal Booster

If we assume $\mathcal{H}$ is finite then this can be considered as a zero-sum game between learner and booster. By Von Neumann's Minimax Theorem:

- Booster's Strategy

$$\min_{p \in \triangle([n])} \max_{h \in \mathcal{H}} p^\top A e_h = \max_{w \in \triangle(\mathcal{H})} \min_{i \in [n]} e_i^\top A w$$

Existence of an Ideal Booster

If we assume $\mathcal{H}$ is finite then this can be considered as a zero-sum game between learner and booster. By Von Neumann's Minimax Theorem:

- Booster's Strategy

$$\min_{p \in \triangle([n])} \max_{h \in \mathcal{H}} p^\top A e_h = \max_{w \in \triangle(\mathcal{H})} \min_{i \in [n]} e_i^\top A w$$

- (Learner's) Payoff Matrix

Existence of an Ideal Booster

If we assume $\mathcal{H}$ is finite then this can be considered as a zero-sum game between learner and booster. By Von Neumann's Minimax Theorem:

- Booster's Strategy

$$\min_{p \in \triangle([n])} \max_{h \in \mathcal{H}} p^\top A e_h = \max_{w \in \triangle(\mathcal{H})} \min_{i \in [n]} e_i^\top A w$$

- (Learner's) Payoff Matrix
- Learner's Strategy

Existence Continued

Preceeding argument implies existence of a weighted majority vote classifier which has zero training error.

$$\max_{w \in \triangle(\mathcal{H})} \min_{i \in [n]} e_i^\top A w \geq 2\gamma > 0$$

$$\Updownarrow$$

$$\forall i \in [n] \quad Y_i\Big(\sum_{h \in \mathcal{H}} w_h^* h(X_i)\Big) > 0$$

Existence Continued

Preceeding argument implies existence of a weighted majority vote classifier which has zero training error.

$$\max_{w \in \triangle(\mathcal{H})} \min_{i \in [n]} e_i^\top A w \geq 2\gamma > 0$$

$$\Updownarrow$$

$$\forall i \in [n] \quad Y_i\Big(\sum_{h \in \mathcal{H}} w_h^* h(X_i)\Big) > 0$$

Is it computationally tractable to find $g(X) = \text{sign}(\sum_{h \in \mathcal{H}} w_h^* h(X))$, though? How should we find the weights?

Outline

Roadmap

- It is promising to learn about Booster's Minimax strategy by playing the game multiple times and learn from your mistakes.

Roadmap

- It is promising to learn about Booster's Minimax strategy by playing the game multiple times and learn from your mistakes.
- The idea is to change the effective distribution $p \in \triangle([n])$ (Booster's strategy) at each round so that we can trick the learner into spreading out the error.

Roadmap

- It is promising to learn about Booster's Minimax strategy by playing the game multiple times and learn from your mistakes.
- The idea is to change the effective distribution $p \in \triangle([n])$ (Booster's strategy) at each round so that we can trick the learner into spreading out the error.
- Now by taking a majority vote over the hypotheses produced by Weak learner we can make training error zero!

Boosting Repeated Game

- Initialize: $s_0 = 0 \in \mathbb{R}^n$
- For $t = 1, \cdots, T$:
  1- Booster picks a strategy $p_t \in \triangle([n])$.
  2- Weak learner picks $z_t \in \{\pm 1\}^n$ where $z_{t,i} = Y_i h_t(X_i)$ which satisfies $p_t^\top z_t \geq 2\gamma$.
  3- Update state $s_t = s_{t-1} + z_t$.
- Final majority vote rule is $g(X) = sign(\sum_{t=1}^{T} h_t(X))$.
- Loss for Booster is RHS and his Goal is to minimize training error (make it zero):

$$\sum_{i=1}^{n} \mathbf{1}_{\{g(X_i) \neq Y_i\}} = \sum_{i=1}^{n} \mathbf{1}_{\{s_{T,i} \leq 0\}} \leq \sum_{i=1}^{n} e^{-\eta s_{T,i}}$$

Analysis

Suppose s is the state after first $T-1$ rounds. How should the Booster choose p_T in round T?

- Denote, $\Lambda_T(s) := \sum_{i=1}^{n} \phi_T(s_i)$ where $\phi_T(s_i) = e^{-\eta s_i}$. He should pick p which attains the min below:

$$\Lambda_{T-1}(s) := \min_{p \in \triangle([n])} \max_{\substack{z \in \{\pm 1\}^n \\ p^\top z \geq 2\gamma}} \Lambda_T(s+z)$$

Analysis

Suppose s is the state after first $T-1$ rounds. How should the Booster choose p_T in round T?

- Denote, $\Lambda_T(s) := \sum_{i=1}^{n} \phi_T(s_i)$ where $\phi_T(s_i) = e^{-\eta s_i}$. He should pick p which attains the min below:

$$\Lambda_{T-1}(s) := \min_{p \in \triangle([n])} \max_{\substack{z \in \{\pm 1\}^n \\ p^\top z \geq 2\gamma}} \Lambda_T(s+z)$$

- By the same argument if we assume s is the state after $t-1$ rounds of play we can define total incurred loss of the booster as:

$$\Lambda_{t-1}(s) := \min_{p \in \triangle([n])} \max_{\substack{z \in \{\pm\}^n \\ p^\top z \geq 2\gamma}} \Lambda_t(s+z)$$

Value of the Game

- The minimum possible total loss ahievable by Booster against an optimal Learner becomes:

$$\min_{p_1 \in \triangle([n])} \max_{\substack{z_1 \in \{\pm\}^n \\ p_1^\top z_1 \geq 2\gamma}} \min_{p_2 \in \triangle([n])} \max_{\substack{z_2 \in \{\pm\}^n \\ p_2^\top z_2 \geq 2\gamma}} \cdots \min_{p_T \in \triangle([n])} \max_{\substack{z_T \in \{\pm\}^n \\ p_T^\top z_T \geq 2\gamma}} \Lambda_T\left(\sum_{t=1}^{T} z_t\right)$$

- Booster tries to make this value less than one in order to obtain zero training error.
- Unfortunately this expression is unwieldy and it's not clear there exists an efficient algorithm to compute the best strategy.
- Instead, we work with a tractable upper bound.

Toward Decomposition on States

- The trick is to somehow rid of intertwined coordinates.

$$\begin{aligned}
\Lambda_{t-1}(s) &= \min_{p\in\triangle([n])} \max_{\substack{z\in\{\pm1\}^n\\ p^\top z\geq 2\gamma}} \Lambda_t(s+z) \\
&= \min_{p\in\triangle([n])} \max_{z\in\{\pm1\}^n} \min_{\lambda\geq 0} \Lambda_t(s+z)+\lambda(p^\top z-2\gamma) \\
&\leq \min_{p\in\triangle([n])} \min_{\lambda\geq 0} \max_{z\in\{\pm1\}^n} \Lambda_t(s+z)+\lambda(p^\top z-2\gamma) \\
&= \min_{q\in\mathbb{R}_+^n} \max_{z\in\{\pm1\}^n} \Lambda_t(s+z)+q^\top(z-2\gamma)
\end{aligned}$$

- Define recursively:

$$\phi_{t-1}(s_i) := \min_{q_i>0} \max_{z_i\in\{\pm1\}} \phi_t(s_i+z_i)+q_i(z_i-2\gamma)$$

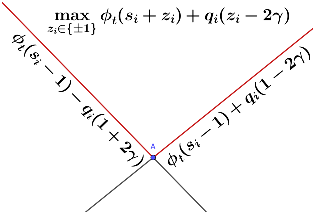
Claim. $\forall t \quad \Lambda_t(s) \leq \sum_{i=1}^{n} \phi_t(s_i)$

Proof. Backward induction on t:

$$\begin{aligned}
\Lambda_{t-1}(s) &\leq \min_{q \in \mathbb{R}_+^n} \max_{z \in \{\pm 1\}^n} \Lambda_t(s+z) + q^\top (z - 2\gamma) \\
&\leq \min_{q \in \mathbb{R}_+^n} \max_{z \in \{\pm 1\}^n} \sum_{i=1}^{n} \phi_t(s_i + z_i) + q_i(z_i - 2\gamma) \\
&= \sum_{i=1}^{n} \min_{q_i \geq 0} \max_{z_i \in \{\pm 1\}} \phi_t(s_i + z_i) + q_i(z_i - 2\gamma) \\
&= \sum_{i=1}^{n} \phi_{t-1}(s_i)
\end{aligned}$$

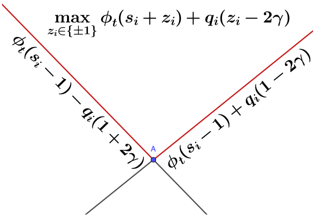
Achieving The Bound

- $\phi_t(s_i + z_i) + q_i(z_i - 2\gamma)$ is linear in q_i



Achieving The Bound

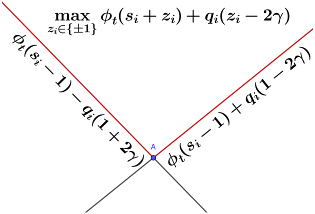
- $\phi_t(s_i + z_i) + q_i(z_i - 2\gamma)$ is linear in q_i
- Intersection point achieves the Minimax.



Achieving The Bound

- $\phi_t(s_i + z_i) + q_i(z_i - 2\gamma)$ is linear in q_i
- Intersection point achieves the Minimax.

$$q_i = \frac{\phi_t(s_i + 1) - \phi_t(s_i - 1)}{2}$$

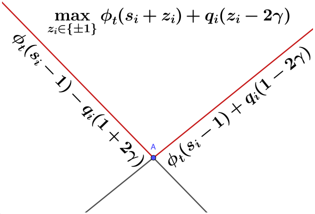


Achieving The Bound

- $\phi_t(s_i + z_i) + q_i(z_i - 2\gamma)$ is linear in q_i
- Intersection point achieves the Minimax.

$$q_i = \frac{\phi_t(s_i + 1) - \phi_t(s_i - 1)}{2}$$

$$\phi_{t-1}(s_i) = (\frac{1}{2}+\gamma)\phi_t(s_i+1)+(\frac{1}{2}-\gamma)\phi_t(s_i-1)$$



Booster's Strategy

- Solution to the recursion formula becomes:

$$\phi_t(s_i) = ((\frac{1}{2} + \gamma)e^{-\eta} + (\frac{1}{2} - \gamma)e^{+\eta})^{T-t} e^{-\eta s_i}$$

Booster's Strategy

- Solution to the recursion formula becomes:

$$\phi_t(s_i) = ((\frac{1}{2} + \gamma)e^{-\eta} + (\frac{1}{2} - \gamma)e^{+\eta})^{T-t} e^{-\eta s_i}$$

- Thus, we obtain an explicit formula for Booster's strategy on round t:

$$\boxed{p_{t,i} \propto q_i \propto e^{-\eta s_{t-1,i}}}$$

Booster's Strategy

- Solution to the recursion formula becomes:

$$\phi_t(s_i) = \left(\left(\frac{1}{2} + \gamma\right)e^{-\eta} + \left(\frac{1}{2} - \gamma\right)e^{+\eta}\right)^{T-t} e^{-\eta s_i}$$

- Thus, we obtain an explicit formula for Booster's strategy on round t:

$$\boxed{p_{t,i} \propto q_i \propto e^{-\eta s_{t-1,i}}}$$

Intuition Booster tries to weigh more on hard samples to force the Weak learner to learn that sample...

Suppose Booster plays the proposed strategy and encounters states $s_0, s_1, \cdots, s_T$.

Claim. $$\sum_{i=1}^{n} \phi_T(s_{T,i}) \leq \sum_{i=1}^{n} \phi_{T-1}(s_{T-1,i}) \leq \cdots \leq \sum_{i=1}^{n} \phi_0(s_{0,i})$$

Suppose Booster plays the proposed strategy and encounters states $s_0, s_1, \cdots, s_T$.

Claim. $\sum_{i=1}^{n} \phi_T(s_{T,i}) \leq \sum_{i=1}^{n} \phi_{T-1}(s_{T-1,i}) \leq \cdots \leq \sum_{i=1}^{n} \phi_0(s_{0,i})$

Proof.

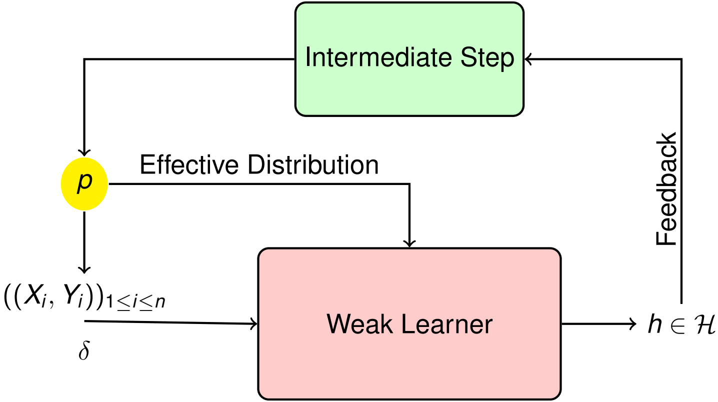
$$
\begin{aligned}
\sum_{i=1}^{n} \phi_{t-1}(s_{t,i}) &= \sum_{i=1}^{n} \min_{q_i \geq 0} \max_{z_i \in \{\pm 1\}} \phi_t(s_{t,i} + z_i) + q_i(z_i - 2\gamma) \\
&= \sum_{i=1}^{n} \max_{z_i \in \{\pm 1\}} \phi_t(s_{t,i} + z_i) + q_{t,i}(z_i - 2\gamma) \\
&\geq \sum_{i=1}^{n} \phi_t(s_{t,i} + z_{t,i}) + \underbrace{\sum_{i=1}^{n} q_{t,i}(z_{t,i} - 2\gamma)}_{\geq 0}
\end{aligned}
$$

Suppose Booster plays the proposed strategy and encounters states $s_0, s_1, \cdots, s_T$.

Claim. $\sum_{i=1}^{n} \phi_T(s_{T,i}) \leq \sum_{i=1}^{n} \phi_{T-1}(s_{T-1,i}) \leq \cdots \leq \sum_{i=1}^{n} \phi_0(s_{0,i})$

$$\begin{aligned}
\text{Training Error } &= \sum_{i=1}^{n} \mathbf{1}_{\{s_{T,i} \leq 0\}} \leq \sum_{i=1}^{n} e^{-\eta s_{T,i}} = \sum_{i=1}^{n} \phi_T(s_{T,i}) \\
&\leq \sum_{i=1}^{n} \phi_0(s_{0,i}) = n\phi_0(0) \\
\text{(Optimize over } \eta) &= n((\frac{1}{2} + \gamma)e^{-\eta} + (\frac{1}{2} - \gamma)e^{+\eta})^T \\
\text{(Setting } \eta = \frac{1}{2}\log(\frac{1/2+\gamma}{1/2-\gamma})) &= n(1-4\gamma^2)^{\frac{T}{2}} \xrightarrow{T\to\infty} 0
\end{aligned}$$

Boosting Algortithm Flowchart



Outline

AdaBoost

- As we saw Booster was capable of making the loss very small. However, the caveat is T, γ should be known in advance which is an impractical assumption.

AdaBoost

- As we saw Booster was capable of making the loss very small. However, the caveat is T, γ should be known in advance which is an impractical assumption.
- AdaBoost rectify this by setting:

$$\underbrace{\gamma_t = \frac{1}{2}\sum_{i=1}^{n} Y_i h_t(X_i)}_{\text{Advantage of hypothesis } h_t}, \quad \underbrace{\eta_t = \frac{1}{2}\log\left(\frac{1/2+\gamma_t}{1/2-\gamma_t}\right)}_{\text{Amount of trust should be put onto } h_t}$$

AdaBoost

- As we saw Booster was capable of making the loss very small. However, the caveat is T, γ should be known in advance which is an impractical assumption.
- AdaBoost rectify this by setting:

$$\underbrace{\gamma_t = \frac{1}{2}\sum_{i=1}^{n} Y_i h_t(X_i)}_{\text{Advantage of hypothesis } h_t} , \quad \underbrace{\eta_t = \frac{1}{2}\log\left(\frac{1/2+\gamma_t}{1/2-\gamma_t}\right)}_{\text{Amount of trust should be put onto } h_t}$$

- Booster's strategy at round t becomes $p_{t,i} \propto e^{-\sum_{\tau=1}^{t-1} \eta_\tau z_{\tau,i}}$ as opposed to $p_{t,i} \propto e^{-\eta s_{t,i}} = e^{-\eta \sum_{\tau=1}^{t-1} z_{\tau,i}}$.

AdaBoost

- As we saw Booster was capable of making the loss very small. However, the caveat is T, γ should be known in advance which is an impractical assumption.
- AdaBoost rectify this by setting:

$$\underbrace{\gamma_t = \frac{1}{2}\sum_{i=1}^{n} Y_i h_t(X_i)}_{\text{Advantage of hypothesis } h_t} , \qquad \underbrace{\eta_t = \frac{1}{2}\log(\frac{1/2+\gamma_t}{1/2-\gamma_t})}_{\text{Amount of trust should be put onto } h_t}$$

- Booster's strategy at round t becomes $p_{t,i} \propto e^{-\sum_{\tau=1}^{t-1}\eta_\tau z_{\tau,i}}$ as opposed to $p_{t,i} \propto e^{-\eta s_{t,i}} = e^{-\eta \sum_{\tau=1}^{t-1} z_{\tau,i}}$.
- Final majority vote becomes $g(X) = \text{sign}(\sum_{t=1}^{T} \eta_t h_t(X))$

AdaBoost

Theorem

Suppose the weak learning algorithm, when called by AdaBoost, generates hypotheses with advantages $\gamma_1, \cdots, \gamma_T$. Then the final bound on number of misclassified examples by the majority vote becomes:

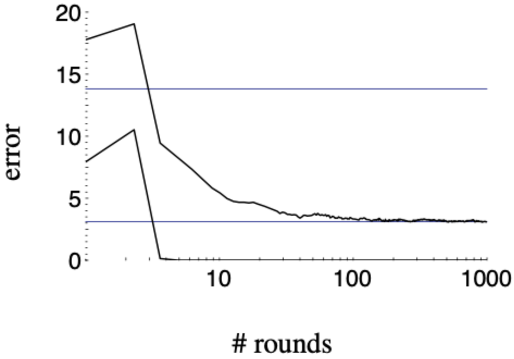
$$n \prod_{t=1}^{T} \sqrt{1 - 4\gamma_t^2}$$

Remark

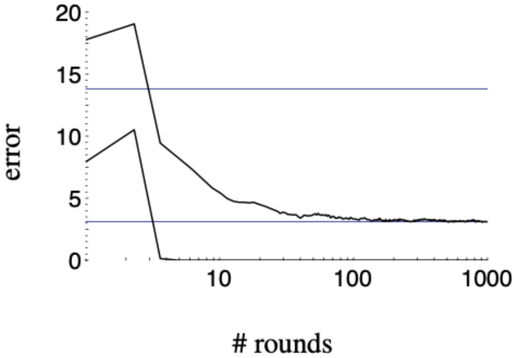
γ_t does not require to be positive which corresponds to a classifier better than random guessing and the bound still holds.

Outline

Paradox

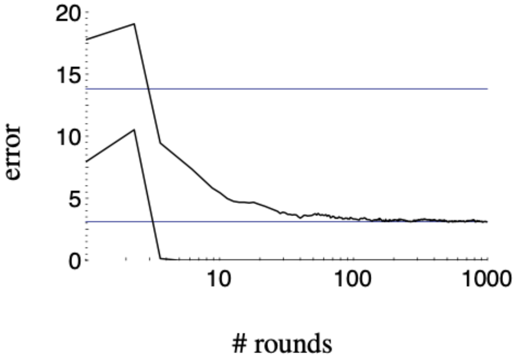


Paradox



- How can it be that complex combined classifiers are performing well? Why test error flatens?!

Paradox



A How can it be that complex combined classifiers are performing well? Why test error flatens?!

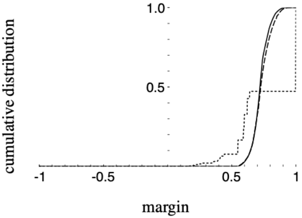
B How come training error is zero but test error is still reducing?

Is a simpler classifier a better one?

- One might say η_t are rapidly converging to zero so the number of classifiers combined is effectively bounded.
  - This is not true since if $\eta_t = \frac{1}{2}\log(\frac{1/2+\gamma_t}{1/2-\gamma_t})$ goes to zero then γ_t must go to zero but it stays around 44-45% in this dataset.
  - This indicates resistance to overfitting! Don't get me wrong, though, there are cases which AdaBoost overfits. This happens when we use very weak base classifiers...

Margin Theory

- Additional information lies in the confidence of our prediction, i.e., $|g(X)|$ which is the margin corresponding to that sample.
- The confidence in our predictions increases significantly with additional rounds of AdaBoost
- There is a Generalization theorem by Schapire and other peers which relates true error with emipirical distribution of the margin...



Conclusion

- We showed boosting had its roots in a purely theoretical question.
- Proved existence of an ideal Majority Vote Booster and then attempted to give an algorithm to find such classifiers.
- We proved training error can be very small (even zero) after enough number of iterations.
- We Introduced AdaBoost which was basically an adaptation from the boosting algorithm stated.
- We gave some intuition on how Boosting resist to overfit.

References I

Daniel Hsu.
Weak Versus Strong Learning.

Yoav Freund. Robert E. Schapire.
A Decision-Theoric Generalization of Online Learning and an Application in Boosting
European conference on computational learning theory. Springer, Berlin, Heidelberg, 1995

Robert E. Schapire. Yoav Freund.
Boosting. Foundations and Algorithms.
MIT Press, 2012