Boosting

Peng Lingwei

April 11, 2019

Contents

10 Boosting

Boosting is an algorithm that grew out of a theoretical question and became a vary practical machine learning tool. The boosting approach uses a generalization linear approach to address two major issues:

- Bias-complexity tradeoff.
- Computational complexity of learning.The boosting algorithm amplifies the accuracy of weak learners.

AdaBoost(Adaptive Boost) stemmed from the theoretical question of whether an efficient weak learner can be "boosted" into an efficient strong learner.

10.1 WEAK LEARNABILITY

The fundamental theorem of learning theory characterizes the family of learnable classes and states that every PAC learnable class can be learned using any ERM algorithm by ignoring the computational aspect of learning.

Definition 10.1. *(γ-Weak-Learnability).*

- γ-weak-learner, $A : \exists m_{\mathcal{H}} : (0,1) \to \mathbb{N}$,*such that,* $\forall \delta \in (0,1), \forall$ *distribution* $\mathcal{D}$ *over* $\mathcal{X}$, $\forall f : \mathcal{X} \to \{\pm 1\}$, *if* $m \geq m_{\mathcal{H},\mathcal{D},f}(\delta)$,*we have,*

$$\mathbb{P}(L_{\mathcal{D},f}(A(S)) \leq 1/2 - \gamma) \geq 1 - \delta.$$

- $\gamma - weak - learnable,\ \mathcal{H} : \exists \gamma - weak - learner,\ A\ for\ \mathcal{H}.$

In chapter6, we have $m_{\mathcal{H}}(\epsilon, \delta) \geq C_1 \frac{d+log(1/\delta)}{\epsilon}$, so when $d = \infty$ then $\mathcal{H}$ is not $\gamma - weak - learnbale$. This implies that from the statistical perspective, weak learnbality is also characterized by the VC dimension of $\mathcal{H}$ and therefore is just as hard as PAC learning. (Ignoring computational complexity).

Considering computational complexity, we can get efficiently implemented weak learning. One possible approach is to take a "simple" hypothesis class, denoted B, and to apply ERM with respect to B as the weak learning algorithm.For this to work, we nned B with two properties:

- ERM_B is efficiently implementable.
- For every sample taht is labeled by some hypothesis from $\mathcal{H}$, any ERM_B hypothesis will have an error of at most $1/2 - \gamma$.

Example 10.1. Weak Learning of 3-Piece Classfiers Using Decision Stumps

- 3-Piece Classifiers $\mathcal{H} = \{h_{\theta_1,\theta_2,b:\theta_1,\theta_2 \in \mathbb{R}}, \theta_1 < \theta_2, b \in \{\pm 1\}\}$
  $h_{\theta_1,\theta_2,b}(x) = b \cdot 1\{x < \theta_1 \vee x > \theta_2\}$
- Decision Stumps *:* $B = \{x \mapsto b \cdot sign(x - \theta) : \theta \in \mathbb{R}, b \in \{\pm 1\}\}$
- *Proof.* (ERM_B is a $\gamma - weak - learner$ for $\mathcal{H}$)
  Since $\forall \mathcal{D}, \exists h \in B, L_{\mathcal{D}}(h) \leq 1/3$,
  In ch6, we have that when $m \geq \log \frac{2/\delta}{\epsilon}$:
  $$\mathbb{P}\{L_{\mathcal{D}}(ERM_B(S)) \leq minL_{\mathcal{D}}(h) + \epsilon\} \geq 1 - \delta.$$
  We set $\epsilon = 1/12$, then we obtain that the error of ERM_B is at most $1/3 + 1/12 = 1/2 - 1/12$. □

Theorem 10.1. ***Boosting the Confidence****. Let A be an algorithm that guarantees the following: There exist some constant* $\delta_0 \in (0, 1)$ *and a function* $m_{\mathcal{H}} \to \mathbb{N}$ *suchtaht for every* $\epsilon \in (0, 1)$*, if* $m \geq m_{\mathcal{H}}(\epsilon)$ *then for every distribution D it holds that with probability of at least* $1 - \delta_0$*,* $L_D(A(S)) \leq \min_{h \in \mathcal{H}} L_D(h) + \epsilon$*.We have*
$$m_{\mathcal{H}}(\epsilon, \delta) \leq km_{\mathcal{H}}(\epsilon/2) + \left\lceil \frac{8\log(4k/\delta)}{\epsilon^2} \right\rceil.$$
where $k = \lceil \log(\delta)/\log(\delta_0) \rceil$*.*

Proof. Pick k "chunks" of size $m_{\mathcal{H}}(\epsilon/2)$. Apply A on each of these chunks, to obtain $\hat{h_1}, \ldots, \hat{h_k}$. Note that the probability that $min_{i\in[k]}L_D(\hat{h_i}) \leq minL_D(h) + \epsilon/2$ is at least $1 - \delta_0^k \geq 1 - \delta/2$.Then, we need $k > \log(\delta)/\log(\delta_0)$

Now, apply an ERM over the class $\hat{\mathcal{H}} := \{\hat{h}_1, \ldots, \hat{h}_k\}$ with the training data being the last chunk of size $m_{\mathcal{H}}(\epsilon/2, \delta/2) = \left\lceil \frac{8\log(4k/\delta)}{\epsilon^2} \right\rceil$

Then we can guarantee that
$$\mathbb{P}\left\{L_D(\hat{h}) \leq \min_{i\in[k]} L_D(h_i) + \frac{\epsilon}{2} \leq \min_{h\in\mathcal{H}} L_D(h) + \epsilon\right\} \geq 1 - \delta.$$
□

10.1.1 Efficient Implementation of ERM for Decision Stumps

Let $\mathcal{X} = \mathbb{R}^d$ and consider the base hypothesis class of decision stumps over $\mathbb{R}^d$, namely,

$$\mathcal{H}_{DS} = \{\mathbf{x} \mapsto sign(\theta - x_i) \cdot b : \theta \in \mathbb{R}, i \in [d], b \in \{\pm 1\}\}.$$

Let $\mathbf{D}$ be a probability vector in $\mathbb{R}^m$ ($\sum_i D_i = 1$).

$$L_{\mathbf{D}}(h) = \sum_{i=1}^{m} D_i \mathbf{1}\{h(\mathbf{x}_i \neq y_i)\}.$$

ERM:

$$\min_{j \in [d]} \min_{\theta \in \mathbb{R}} \left(\sum_{i: y_i = 1} D_i \mathbf{1}\{x_{i,j} > \theta\} + \sum_{i: y_i = -1} D_i \mathbf{1}\{x_{i,j} \leq \theta\} \right) \tag{10.1}$$

Let training set is $x_{1,j} - 1 = x_{0,j} \leq x_{1,j} \leq x_{2,j} \leq \cdots \leq x_{m,j} \leq x_{m+1,j} = x_{m,j} + 1$, then define Θ:

$$\theta \in \Theta_j = \left\{ \frac{x_{i,j} + x_{x+1,j}}{2} : i \in x_{\cdot,j} \right\}.$$

We use following equation to calculate Equ(10.1) in $O(dm)$ instead of $O(dm^2)$.

$$F(\theta') = F(\theta) - D_i \mathbf{1}\{y_i = 1\} + D_i \mathbf{1}\{y_i = -1\} = F(\theta) - y_i D_i.$$

Algorithm 1 ERM for Decision Stumps

```
Require: S = {$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$}, distribution vector $\mathbf{D}$
Ensure: $F^* = \infty$
  for $j = 1, \ldots, d$ do
    sort S using the j'th coordinate, and denote
    $x_{1,j} \leq x_{2,j} \leq \cdots \leq x_{m,j} \leq x_{m+1,j} \overset{def}{=} x_{m,j} + 1$
    $F = \sum_{i: y_i = 1} D_i$
    if $F < F^*$ then
      $F^* = F, \theta^* = x_{1,j} - 1, j^* = j$
    end if
    for i=1,...,m do
      $F = F - y_i D_i$
      if $F < F^*$ and $x_{i,j} \neq x_{i+1,j}$ then
        $F^* = F, \theta^* = (x_{i,j} + x_{i+1,j}), j^* = j$
      end if
    end for
  end for
  return $j^*, \theta^*$
```

10.2 ADABOOST

AdaBoost constructs $\mathbf{D}^{(t)}$.The weak learner is assumed to return a "weak" hypothesis, h_t, whose error,

$$\epsilon_t \overset{def}{=} L_{\mathbf{D}^{(t)}}(h_t) \overset{def}{=} \sum_{i=1}^{m} D_i^{(t)} \mathbf{1}\{h_t(\mathbf{x}_i) \neq y_i\}.$$

is at most $\frac{1}{2} - \gamma$.

Algorithm 2 AdaBoost

```
Require: S = $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$, weak learner WL, number of rounds T.
Ensure: $\mathbf{D}^{(1)} = (\frac{1}{m}, \cdots, \frac{1}{m})$
  for $t = 1, \ldots, T$ do
    $h_t = WL(\mathbf{D}^{(1)}, S)$
    $\epsilon_t = \sum_{i=1}^{m} D_i^{(t)} \mathbf{1}\{y_i \neq h_t(\mathbf{x}_i)\}$
    $w_t = \frac{1}{2}\log(\frac{1}{\epsilon_t} - 1)$
    for $i = 1, \ldots, T$ do
      $D_i^{(t+1)} = \frac{D_i^{(t)} exp(-w_t y_i h_t(\mathbf{x}_i))}{\sum_{j=1}^{m} D_j^{(t)} exp(-w_t y_j h_t(\mathbf{x}_j))}$
    end for
  end for
  return $h_s((x)) = sign\left(\sum_{t=1}^{T} w_t h_t(\mathbf{x})\right)$
```

Theorem 10.2. *w.r.t. training set S, iteration of AdaBoost T, weak learner B with $\epsilon_t \leq 1/2 - \gamma$. Then,*

$$L_S(h_s) = \frac{1}{m}\sum_{i=1}^{m} \mathbf{1}\{h_S(\mathbf{x}_i \neq y_i)\} \leq exp(-2\gamma^2 T)..$$

Proof. Let $f_t(x) = \sum_{i=1}^{t} w_i h_i(x)$

$$\begin{aligned} D_i^{(T+1)} =& D_i^{(1)} \times \frac{e^{-y_i w_1 h_1(x_i)}}{Z_1} \times \cdots \times \frac{e^{-y_i w_T h_T(x_i)}}{Z_T} \\ =& \frac{D_i^{(1)} exp(-y_i \sum_{t=1}^{T} w_t h_t(x_i))}{\prod_{t=1}^{T} Z_t} \\ =& \frac{D_i^{(1)} exp(-y_i f_T(x_i))}{\prod_{t=1}^{T} Z_t} \end{aligned}$$

$$\begin{aligned}L_S(h_s) = L_{\mathbf{D}^{(1)}}(h_s) =& \sum_{i=1}^{m} D_i^{(1)} \mathbf{1}\{h_s(x_i) \neq y_i\} \\ \leq& \sum_{i=1}^{m} D_i^{(1)} exp(-y_i f_T(x_i)) \\ =& \sum_{i=1}^{m} D_i^{(T+1)} \prod_{t=1}^{T} Z_t \\ =& \prod_{t=1}^{T} Z_t\end{aligned}$$

$$\begin{aligned}Z_t =& \sum_{i=1}^{m} D_i^{(t)} e^{-w_t y_i h_t(x_i)} \\ =& \sum_{i:y_i=h_t(x_i)} D_i^{(t)} e^{-w_t} + \sum_{i:y_i \neq h_t(x_i)} D_i^{(t)} e^{w_t} \\ =& e^{-w_t}(1-\epsilon_t) + e^{w_t}\epsilon_t \\ =& 2\sqrt{\epsilon_t(1-\epsilon_t)} \leq \sqrt{1-4\gamma^2} \leq e^{-2\gamma^2}\end{aligned}$$

□

Theorem10.2 assumes that at each iteration of AdaBoost, the weak learner returns a hypothesis with weighted sample error of at most $1/2 - \gamma$ with probability greater than $1 - \delta$. Using the union bound, the probability that the weak learner will not fail at all is at least $1 - \delta T$, so we need small δ and large sample complexity.

10.3 LINEAR COMBINATIONS OF BASE HYPOTHESES

The output of AdaBoost will be a member of the following class:

$$L(B,T) = \left\{ x \mapsto sign\left(\sum_{t=1}^{T} w_t h_t(x)\right) : w \in \mathbb{R}^T, \forall t, h_t \in B \right\} \tag{10.2}$$

Example 10.2. *Consider the base class is Decision Stumps,*

$$\mathcal{H}_{DS1} = \{x \mapsto sign(x-\theta) \cdot b : \theta \in \mathbb{R}, b \in \{\pm 1\}\}.$$

Let g_r be a piece-wise constant function with at most r pieces; that is, there exist thresholds $-\infty = \theta_0 < \theta_1 < \theta_2 < \cdots < \theta_r = \infty$ such that:

$$g_r(x) = \sum_{i=1}^{r} \alpha_i \mathbf{1}\{x \in (\theta_{i-1}, \theta_i\} \quad \forall i, \alpha_i \in \{\pm 1\}.$$

Let $\mathcal{G}_r = \{g_r : \alpha_t = (-1)^t\}$, we will show that $\mathcal{G}_T \subset L(\mathcal{H}_{DS1}, T)$.
Now, the function

$$h(x) = sign\left(\sum_{t=1}^{T} w_t sign(x - \theta_{t-1})\right).$$

where $w_1 = -0.5$ and for $t > 1, w_t = (-1)^t$, is in $L(\mathcal{H}_{DS1}, T)$ and is equal to $g_r \in \mathcal{G}$.

The example 10.2 shows that $L(\mathcal{H}_{DS1}, T)$ can shatter any set of $T+1$ instances in $\mathbb{R}$; hence the VC-dimension of $L(\mathcal{H}_{DS1}, T)$ is at least $T+1$.

10.3.1 The VC-Dimension of $L(B, T)$

Lemma 10.1. *Let $L(B,T)$ be as defined in Equation(10.2).Assume that both T and VCdim(B) are at least 3. Then,*

$$VCdim(L(B,T)) \leq T(VCdim(B)+1)(3log(T(VCdim(B)+1))+1).$$

Proof. Denote $d = VCdim(B)$. Let $C = \{x_1, \ldots, x_m\}$ be a set that is shattered by $L(B,T)$. So $|B_C| \leq (em/d)^d$. We choose $h_1, \ldots, h_T \in B$, then there are at most $(em/d)^{dT}$ ways to do it. Next, for each such choice, we apply a linear predictor, which yields at most $(em/T)^T$ dichotomies.Therefore, the overall number number of dichotomies we can construct is upper bounded by

$$(em/d)^{dT}(em/T)^T < m^{(d+1)T} \quad (d, T \geq 3).$$

We assume C is shattered by $L(B,T)$, which yields

$$\begin{aligned} &2^m \leq m^{(d+1)T} \\ \Rightarrow &m \leq log(m)\frac{(d+1)T}{log2} \\ \Rightarrow &m \leq 2\frac{(d+1)T}{log2}\log\frac{(d+1)T}{log2} \\ \Rightarrow &m \leq (d+1)T(3log((d+1)T)+2) \end{aligned}$$

□

Theorem 10.3. ***VC of union**. Let $\mathcal{H}_1, \ldots, \mathcal{H}_r$ be the hypothesis classes over some fixed domain set $\mathcal{X}$. Let $d = max_i VCdim(\mathcal{H}_i)$ and assume for simplicity that $d \geq 3$.Then,*

$$VCdim(\cup_{i=1}^r \mathcal{H}_i) \leq 4d\log(2d) + 2\log(r) \tag{10.3}$$

Proof. Let $C = \{c_1, \ldots, c_m\}$ is shattered by $\mathcal{H}$, then $\tau_{\mathcal{H}}(m) = 2^k$ and

$$\tau_{\mathcal{H}}(m) \leq \sum_{i=1}^r \mathcal{H}_i(m) \leq rm^d.$$

by using Sauer's lemma and $d \geq 3$. Then, we have

$$2^m \leq rm^d \Rightarrow m \leq d\ln m + \ln r \Rightarrow m \leq 4d\log(2d) + 2\log(r).$$

□

Lemma 10.2. *Let $L(B,T)$ be as defined in Equation(10.2).Assume that both T and VCdim(B) are at least 3. Then,*

$$VCdim(L(B,T)) \geq 0.5T\log(d).$$

Proof. Firstly, we prove that when B_d be the class of decision stumps over $\mathbb{R}^d$, we have $log(d) \leq VCdim(B_d) \leq 16 + 2log(d)$.

Let, $B_d = \{h_{j,d,\theta} : j \in [d], b \in \{-1, 1\}, \theta \in \mathcal{R}\}$, where $h_{j,b,\theta}(\mathbf{x}) = b \cdot sign(\theta - x_j)$. For $B_d^j = \{h_{b,\theta} : b \in \{-1, 1\}, \theta \in \mathbb{R}\}$, where $h_{b,\theta}(\mathbf{x}) = b \cdot sign(\theta - x_j)$ Note that $VCdim(B_d^j) = 2$, so $VCdim(B_d) = VCdim(\cup_{j=1}^d B_d^j) \leq 16 + 2\log d$. (Not really used). The lower bound is trivial($log(d) \leq d$).

Secondly, we prove that $VCdim(L(B_d, T)) \geq 0.5T\log(d)$.

We pick every k instances in each $A, 2A, \ldots, \frac{T}{2}A$. So we have $\frac{T}{2}k$ instances, we can easily proof that these instances are shattered by $L(B_d, T)$ using :

$$h(x) = sign\left(h_{j_1,-1,1/2}(x) + h_{j_1,1,3/2}(x) + h_{j_2,-1,3/2}(x) + h_{j_2,1,5/2}(x) + \ldots\right).$$

□