MLT Homework 14

Ana Borovac
Bas Haver

January 7, 2019

Question 1

Boosting the Confidence: *Let A be an algorithm that guarantees the following: There exists some constant $\delta_0 \in (0,1)$ and a function $m_{\mathcal{H}} : (0,1) \to \mathbb{N}$ such that for every $\epsilon \in (0,1)$, if $m \geq m_{\mathcal{H}}(\epsilon)$ then for every distribution $\mathcal{D}$ it holds that with probability of at least $1 - \delta_0$, $L_{\mathcal{D}}(A(S)) \leq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$.*
Suggest a procedure that relies on A and learns $\mathcal{H}$ in the usual agnostic PAC learning model and has a sample complexity of

$$m_{\mathcal{H}}(\epsilon, \delta) \leq k m_{\mathcal{H}}(\epsilon/2) + \left\lceil \frac{8 \log(4k/\delta)}{\epsilon^2} \right\rceil,$$

where

$$k = \left\lceil \log\left(\frac{1}{2}\delta\right) / \log(\delta_0) \right\rceil.$$

Hint: Divide the data into $k+1$ chunks, where each of the first k chunks is of size $m_{\mathcal{H}}(\epsilon/2)$ examples. Train the first k chunks using A. Argue that the probability that for all of these chunks we have $L_{\mathcal{D}}(A(S)) > \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon/2$ is at most $\delta_0^k \leq \delta/2$.
Finally, use the last chunk to choose from the k hypotheses that A generated from the k chunks (by relying on Corollary 4.6).

Solution

As suggested by the hint, we divide the data into $k+1$ chunks, where each of the first k chunks is of size $m_{\mathcal{H}}(\epsilon/2)$. We nog apply algorithm A on the first k chunks. For every of the first k chunks we have that algorithm A guarantees that $L_{\mathcal{D}}(A(S)) > \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon/2$ with probability at most δ_0 when we have at least $m_{\mathcal{H}}$ data. Therefore the probability that this holds for all of these chunks is at most $\delta_0^k = \delta_0^{\lceil \log(\frac{1}{2}\delta)/\log(\delta_0) \rceil} \leq \delta_0^{\log_{\delta_0}(\frac{1}{2}\delta)} = \frac{1}{2}\delta$.
Now by Corollary 4.6 we have

$$m_{\mathcal{H}}(\epsilon/2, \delta/2) \leq m_{\mathcal{H}}^{UC}(\epsilon/4, \delta/2) \leq \lceil \frac{2 \log 4|\mathcal{H}|/\delta}{(\frac{1}{2}\epsilon)^2} \rceil.$$

Now when choosing from the outputted hypotheses as obtained from the first k chunks, we have $|\mathcal{H}| = k$, so we obtain

$$m_{\mathcal{H}}(\epsilon/2, \delta/2) \leq \left\lceil \frac{8 \log(4k/\delta)}{\epsilon^2} \right\rceil .$$

Since the error made in the first k chunks now is $\epsilon/2$ as well as in the last chunk, we can conclude that the procedure has a sample complexity of

$$m_{\mathcal{H}}(\epsilon, \delta) \leq k m_{\mathcal{H}}(\epsilon/2) + \left\lceil \frac{8 \log(4k/\delta)}{\epsilon^2} \right\rceil .$$

Question 2

Prove that the function h given in Equation (10.5) equals the piece-wise constant function defined according to the same thresholds as h.

Solution

We would like to prove that is holds if $x \in (\theta_{k-1}, \theta_k]$, then $h(x) = (-1)^k = g(x)$.

$$\begin{aligned}
h(x) &= \text{sign}\left(\sum_{t=1}^{T} w_t \text{sign}(x - \theta_{t-1}) \right) \\
&= \text{sign}(-\frac{1}{2}\text{sign}(x - \theta_0) + (-1)^2 \text{sign}(x - \theta_1) + \\
&\quad + \cdots + \\
&\quad (-1)^k \text{sign}(x - \theta_{k-1}) + (-1)^{k+1} \text{sign}(x - \theta_k) + \\
&\quad + \cdots + \\
&\quad + (-1)^T \text{sign}(x - \theta_{T-1}))
\end{aligned}$$

Since $x \in (\theta_{k-1}, \theta_k]$, it holds:

$$\text{sign}(x - \theta_j) = \begin{cases} 1; & j \in \{0, \ldots, k\} \\ -1; & j \in \{k+1, \ldots, T\} \end{cases}$$

It follows:

$$
\begin{aligned}
h(x) = \text{sign}(&-\frac{1}{2} + (-1)^2 + \\
&+ \cdots + \\
&(-1)^k + (-1)^{k+1}(-1) + \\
&+ \cdots + \\
&+ (-1)^T(-1)) \\
&= \text{sign}\left(-\frac{1}{2} + (-1)^2 + \cdots + (-1)^k + (-1)^{k+2} + \cdots + (-1)^{T+1}\right) \\
&= \text{sign}\left(-\frac{1}{2} + \sum_{i=2}^{T+1}(-1)^i - (-1)^{k+1}\right) \\
&= \text{sign}\left(-\frac{1}{2} + \sum_{i=2}^{T+1}(-1)^i + (-1)^k\right) \\
&= \text{sign}\begin{cases} -\frac{1}{2} + 1 + (-1)^k; & T \text{ is odd} \\ -\frac{1}{2} + 0 + (-1)^k; & T \text{ is even} \end{cases} \\
&= \text{sign}\begin{cases} \frac{1}{2} + (-1)^k; & T \text{ is odd} \\ -\frac{1}{2} + (-1)^k; & T \text{ is even} \end{cases} \\
&= \text{sign}\begin{cases} \frac{1}{2} + (-1); & T \text{ is odd}, k \text{ is odd} \\ \frac{1}{2} + 1; & T \text{ is odd}, k \text{ is even} \\ -\frac{1}{2} + (-1); & T \text{ is even}, k \text{ is odd} \\ -\frac{1}{2} + 1; & T \text{ is even}, k \text{ is even} \end{cases} \\
&= \text{sign}\begin{cases} -\frac{1}{2}; & T \text{ is odd}, k \text{ is odd} \\ \frac{3}{2}; & T \text{ is odd}, k \text{ is even} \\ -\frac{3}{2}; & T \text{ is even}, k \text{ is odd} \\ \frac{1}{2}; & T \text{ is even}, k \text{ is even} \end{cases} \\
&= \begin{cases} -1; & k \text{ is odd} \\ 1; & k \text{ is even} \end{cases} \\
&= (-1)^k
\end{aligned}
$$

Question 3

We have informally argued that the AdaBoost algorithm uses the weighting mechanism to "force" the weak learner to focus on the problematic examples in the next iteration. In this question we will find some rigorous justification for this argument.

Show that the error of ht w.r.t. the distribution $D^{(t+1)}$ is exactly $1/2$. That is, show that for every $t \in [T]$

$$\sum_{i=1}^{m} D_i^{(t+1)} \mathbb{1}_{[y_i \neq h_t(x_i)]} = \frac{1}{2}$$

Solution

From definition of $D_i^{(t+1)}$:

$$\begin{aligned}\sum_{i=1}^{m} D_i^{(t+1)} \mathbb{1}_{[y_i \neq h_t(x_i)]} &= \frac{\sum_{i=1}^{m} D_i^{(t)} e^{-w_t y_i h_t(x_i)} \mathbb{1}_{[y_i \neq h_t(x_i)]}}{\sum_{j=1}^{m} D_j^{(t)} e^{-w_t y_j h_t(x_j)}} \\ &= \frac{\sum_{i=1}^{m} D_i^{(t)} e^{-w_t y_i h_t(x_i)} \mathbb{1}_{[y_i \neq h_t(x_i)]}}{\sum_{j=1}^{m} D_j^{(t)} e^{-w_t y_j h_t(x_j)} \mathbb{1}_{[y_i \neq h_t(x_i)]} + \sum_{j=1}^{m} D_j^{(t)} e^{-w_t y_j h_t(x_j)} \mathbb{1}_{[y_i = h_t(x_i)]}}\end{aligned}$$

Now, we are going to use:

$$\begin{aligned} y_i = h_t(x_i) \quad &\Rightarrow \quad y_i h_t(x_i) = 1 \\ y_i \neq h_t(x_i) \quad &\Rightarrow \quad y_i h_t(x_i) = -1 \end{aligned}$$

It follows:

$$\begin{aligned}
&= \frac{\sum_{i=1}^{m} D_i^{(t)} e^{-w_t y_i h_t(x_i)} \mathbb{1}_{[y_i \neq h_t(x_i)]}}{\sum_{j=1}^{m} D_j^{(t)} e^{-w_t y_j h_t(x_j)} \mathbb{1}_{[y_i \neq h_t(x_i)]} + \sum_{j=1}^{m} D_j^{(t)} e^{-w_t y_j h_t(x_j)} \mathbb{1}_{[y_i = h_t(x_i)]}} \\
&= \frac{e^{w_t} \sum_{i=1}^{m} D_i^{(t)} \mathbb{1}_{[y_i \neq h_t(x_i)]}}{e^{w_t} \sum_{j=1}^{m} D_j^{(t)} \mathbb{1}_{[y_i \neq h_t(x_i)]} + e^{-w_t} \sum_{j=1}^{m} D_j^{(t)} \mathbb{1}_{[y_i = h_t(x_i)]}} \\
&= \frac{e^{w_t} \epsilon_t}{e^{w_t} \epsilon_t + e^{-w_t}(1 - \epsilon_t)}; \quad \epsilon_t = \sum_{i=1}^{m} D_i^{(t)} \mathbb{1}_{[y_i \neq h_t(x_i)]} \\
&= \frac{\epsilon_t}{\epsilon_t + e^{-2w_t}(1 - \epsilon_t)} \\
&= \frac{\epsilon_t}{\epsilon_t + e^{-2\frac{1}{2}\log\left(\frac{1}{\epsilon_t} - 1\right)}(1 - \epsilon_t)}; \quad w_t = \frac{1}{2}\log\left(\frac{1}{\epsilon_t} - 1\right) \\
&= \frac{\epsilon_t}{\epsilon_t + \frac{\epsilon_t}{1 - \epsilon_t}(1 - \epsilon_t)} \\
&= \frac{\epsilon_t}{\epsilon_t + \epsilon_t} \\
&= \frac{1}{2}
\end{aligned}$$

Question 4

In this exercise we discuss the VC-dimension of classes of the form $L(B,T)$. We proved an upper bound of $O(dTlog(dT))$, where $d = VCdim(B)$. Here we wish to prove an almost matching lower bound. However, that will not be the case for all classes B.

Subquestion 4.1

Note that for every class B and every number $T \geq 1$, $VCdim(B) \leq VCdim(L(B,T))$. Find a class B for which $VCdim(B) = VCdim(L(B,T))$ for every $T \geq 1$.

Solution

$L(B,T)$ is by definition:

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

If B equals to:

$$B = \{h_1, h_{-1}\}; \quad h_1(x) = 1,\ h_{-1}(x) = -1\ \forall x$$

Then it's VC-dimension equals to 1 (we can not label two points differently with h_1 or h_{-1}). In this case functions in $L(B,T)$ do not really depend on the input x. With different w_t we can get 1 or -1 for every x. Therefore, we again can not label two points differently and VC-dimension of $L(B,T)$ is also 1.

Subquestion 4.2

Let B_d be the class of decision stumps over $\mathbb{R}^d$. Prove that $\lfloor \log(d) \rfloor \leq VCdim(B_d) \leq 16 + 2\log(d)$.

Solution

We can write $B_d = \{x \mapsto \text{sign}(\theta - x_i) \cdot b : \theta \in \mathbb{R}, i \in [d], b \in \{+1, -1\}\}$ as:

$$B_d = \cup_{i=1}^{d} B_i$$

where $B_i = \{x \mapsto \text{sign}(\theta - x_i) \cdot b : \theta \in \mathbb{R}, b \in \{+1, -1\}\}$. $\text{VCdim}(B_i) = 2$, therefore (from exercise 6.11):

$$\text{VCdim}(\cup_{i=1}^{d} B_i) \leq 4 \cdot 2\log(2 \cdot 2) + 2\log(d) = 16 + 2\log(d)$$

For the lower bound, we follow the hint and assume $d = 2^k$. Let A be a $k \times d$ matrix whose columns are all the d binary vectors in $\{+1, -1\}^k$. The rows of A form a set of k vectors in $\mathbb{R}^d$. Show that this set is shattered by decision

stumps over $\mathbb{R}^d$. If columns represent all of 2^k possible labelings for k vectors, the following functions prove that a set of k vectors can be shattered:

$$x \mapsto \text{sign}(-x_i) \cdot (-1) \quad \Rightarrow \quad x \mapsto \text{sign}(x_i)$$

Therefor the lower bound equals to:

$$\lfloor \log(d) \rfloor \leq \text{VCdim}(B_d)$$

Subquestion 4.3

Let $T \geq 1$ be any integer. Prove that VCdim$(L(Bd, T)) \geq 0{,}5T \log(d)$.