Notes 8: Online to PAC conversion

1. Online to PAC

**Theorem 1.1.** If Algorithm $A$ learns $C$ in Online Mistake Bound model with $M$ mistakes. Then some algorithm PAC learns $C$ using

$$m = \frac{M + 1}{\varepsilon} \ln \frac{M}{\delta}$$

samples

**Proof.** Can assume $A$ only updates its hypothesis after making a mistake (homework)

<table>
<thead>
<tr>
<th>PAC Learning Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keep feeding to $A$ independent samples from $EX(c, D)$</td>
</tr>
<tr>
<td>Until $A$ correctly classifies $\frac{1}{\varepsilon} \ln \frac{M}{\delta}$ samples in a row</td>
</tr>
<tr>
<td>Then output $A$’s current (i.e. last) hypothesis $h$</td>
</tr>
</tbody>
</table>

$A$’s predictions:

\[
\begin{aligned}
\sqrt{\checkmark} \times 
\times 
\times 
\times 
\times 
\times 
\times 
\times 
\times 
\times 
\times 
\times 
\end{aligned}
\]

(repeat $\leq M$ times)

$\leq M + 1$ blocks, each with $\leq \frac{1}{\varepsilon} \ln \frac{M}{\delta}$ samples

#samples used $\leq \frac{M + 1}{\varepsilon} \ln \frac{M}{\delta}$

We now argue final hypothesis $h_{last}$ has error $\leq \varepsilon$ with prob. $\geq 1 - \delta$

If $\text{err}_D(h_i, c) \geq \varepsilon$:

$$P\left[ h_i \text{ correct } k \text{ def } \frac{1}{\varepsilon} \ln \frac{M}{\delta} \text{ times} \right] \leq (1 - \varepsilon)^k \leq e^{-\varepsilon k} = \frac{\delta}{M}$$

$A$ uses $\leq M + 1$ hypotheses $h_1, \ldots, h_{last}$

$$P[\text{any of them has error } \geq \varepsilon \text{ and correct } k \text{ times}] \leq M \cdot \frac{\delta}{M} = \delta$$

Union bound over $M$ (not $M + 1$) because if $h_{last} = h_{M+1}$ then $h_{last}$ has zero error for otherwise $A$ may make $M + 1$ mistakes

If $A$ efficient, so is its PAC version

Implies PAC learning algorithms for

- e.g. (sparse) conjuctions/disjunctions, short decision lists, well-seperated LTFs
- e.g. monotone disjunctions: Elimination Algorithm makes $\leq n$ mistakes
  - its PAC version uses $O\left(\frac{n}{\varepsilon} \ln \left(\frac{n}{\delta}\right)\right)$ samples

2. PAC to Online? No

$X =$ unit interval $= [0, 1]$ 

$C =$ initial intervals $= \{[0, b] \mid 0 \leq b \leq 1\}$

Can be PAC learned with $(1/\varepsilon) \ln (1/\delta)$ samples (same idea as axis-aligned rectangles)

**Claim 2.1.** Any algorithm $A$ for learning closed intervals over $[0, 1]$ in the Online model makes an arbitrarily large number of mistakes

**Proof.** The adversary below forces $A$ to always err

<table>
<thead>
<tr>
<th>Adversary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initially $I = [0, 1]$</td>
</tr>
<tr>
<td>Repeat</td>
</tr>
<tr>
<td>Set $x =$ midpoint of $I$</td>
</tr>
<tr>
<td>Feed $x$ to $A$ and gets $A$’s prediction</td>
</tr>
<tr>
<td>Label $x$ opposite to $A$’s prediction</td>
</tr>
<tr>
<td>If $x$’s correct label is 0, shrinks $I$ by keeping only its left half, else keep only its right half</td>
</tr>
</tbody>
</table>

1
e.g. 1st round $x^1 = 1/2$, if $A$ predicts $x^1$ as 0, then label $x^1$ as 1, update $I$ as $[1/2, 1]$

All positive samples to the left of all negative samples
Some initial interval correctly classifies all labelled samples so far

$X$ above is infinite
How about finite $X$?
Efficient PAC algorithm for $C$ over finite $X$ implies efficient online algorithm with few mistakes?
Previous example of initial intervals (now over $X = \{1, 2, \ldots, n\}$) has efficient online algorithm
namely Halving algorithm with $\leq \log n$ mistakes

In fact Halving algorithm has very efficient implementation in this case (binary search)
Under reasonable cryptographic assumptions, still no PAC-to-online conversion for finite $X = \{0, 1\}^n$