The PAC Learning Model

These notes are slightly edited from previous scribe notes (in Spring 2006) taken by Andrew Fox.

1 Recall Example: PAC Learning a Real Interval \([a,b]\)

**Examples** \(x \in \mathbb{R}\)

**Hypothesis** is an interval \([\hat{a}, \hat{b}]\)

**Learning Algorithm** We take the tightest fit interval around all the positive examples in the training set \([x_{\min}^+, x_{\max}^+]\)

**Result** (From previous lecture) With 3000 I.I.D. training examples we have the following result:

\[
\text{err}(\text{hyp}) < 0.01 \quad \text{with probability } \geq 1 - 2^{-20}
\]

A discussion of how this might be abstracted to yield a model for learning. This concluded in the PAC model below.

2 PAC Learning

**Definition:** Algorithm \(A\) PAC learns concept class \(C = \cup_n C_n:\)

\[
\forall n \\forall c \in C_n, \forall D \text{ distributions over } X_n, \forall 0 < \epsilon < 1 \text{ and } \forall 0 < \delta < 1
\]

If learning algorithm \(A\) is given \(\epsilon, \delta\) as input, and access to random training examples sampled from \(D\) and labeled by \(c\) (the true concept to be learned), then with probability at least \(1 - \delta\), \(A\) outputs a hypothesis \(h \in C\) such that:

\[
\text{Error}(h) = \text{Prob}_D[h(x) \neq c(x)] < \epsilon
\]

Moreover, \(A\)’s running time \(\leq p(n, \frac{1}{\epsilon}, \frac{1}{\delta}, \text{size}(c))\) (i.e. \(A\)’s running time is polynomial in those parameters.)

Note that we abstract the allowed error rate \(\epsilon\), and the failure probability \(\delta\). We also added two new parameters: a complexity parameter for the domain (typically number of features) and the size of the learned concept. All these can be used to quantify the complexity of the learning algorithm.
3 The Tightest Fit Learning Algorithm on Interval [a,b]

**Theorem:** The tightest fit learning algorithm PAC learns intervals when using $m \geq \frac{2}{\epsilon} \ln \frac{2}{\delta}$ examples.

**Proof:** The true concept interval we are trying to learn is [a,b]. We define two points $a+$ and $b-$ as follows:

$$a+ = \arg\min_x (Pr[a, x] \geq \epsilon/2)$$

$$b- = \arg\max_y (Pr[y, b] \geq \epsilon/2)$$

*Note: both points are defined relative to distribution, not relative to the learner.*

Due to our definition of $a+$ and $b-$, we have:

$$Pr(\text{one example does not hit } [a,a+]) = 1 - \epsilon/2$$

$$Pr(\text{all } m \text{ I.I.D. examples do not hit } [a,a+]) = (1 - \epsilon/2)^m$$

$$< e^{-mu/2} ; \quad \text{(using } 1 - x < e^{-x})$$

$$= e^{-\frac{\epsilon}{2} \cdot m} \ln \frac{2}{\delta}$$

$$= \frac{\delta}{2}$$

Also, $Pr(\text{all } m \text{ I.I.D. examples do not hit } [b-,b]) < \delta/2$, by the same argument.

Then it follows that $Pr(\text{sample hits both } [a,a+] \text{ and } [b-,b]) \geq 1 - \delta$.

In this case, $err(hyp) = Pr_D[h(x) \neq c(x)] < \epsilon$ is true since:

$$err(hyp) = Pr_D[a, x_{min}^+] + Pr_D[x_{max}^+, b] ; \quad (x_{min}^+ \text{ is the smallest } \oplus \text{ example, } x_{max}^+ \text{ is the largest})$$

$$\leq Pr[a, a+] + Pr[b-, b]$$

$$= \epsilon/2 + \epsilon/2$$

$$= \epsilon$$

*NB: The magnitude of $\delta$ is usually exponentially small, whereas the magnitude of $\epsilon$ is usually small only in a polynomial sense.*

4 Learning Over Logical Domains

**Examples** are elements of $\{0,1\}^n$ (i.e. a bit string of length n).

We have n boolean variables $x_1, ..., x_n$

$x_i$ takes values in $\{0,1\}$

*E.g.*

011010111 \ $\oplus$
110111011 \ $\oplus$
010110111 \ $\oplus$
000111000 \ $\ominus$
**Hypothesis**  A conjunction of literals (i.e. a variable or a negated variable)

- e.g. $x_1 \land \neg x_3 \land x_7$

**Observations For $\oplus$ examples:**

1. The set of literals that are true (=1) in all $\oplus$ examples include all literals that are in the true concept $c$.

2. If a literal is false in any $\oplus$ example, then it is surely not in the true concept.

### 4.1 The Elimination Algorithm

1. Initially, $hypothesis = x_1 \land \ldots \land x_n \land \neg x_1 \land \ldots \land \neg x_n$

2. Take $m = \frac{2n}{\epsilon} \ln \frac{2n}{\delta}$ I.I.D. examples

   - ignore $\ominus$ examples
   - remove any literal which is false in any positive example

**Theorem:** The Elimination Algorithm PAC learns the class of conjunctions.

**Proof:** analysis started; details in next lecture