CSE 513T Theory of Artificial Intelligence & Machine Learning February 3, 2015

Lecture 7

Lecturer: Brendan Juba *Scribe: SeungJu Lee*

1 From last class

Theorem (Occam's Razor) For any concept class $C = \cup_{n \in N} C_n$ (C_n is over n attributes), and any distribution D over χ_n, if we have

$$m \geq \frac{1}{\varepsilon \ln 2} \left(B(n,m) + \log_2 \frac{1}{\delta} + 1 \right) \text{ examples,}$$

then w. p. $1 - \delta$ <u>any</u> solution h to $Consis_H^B$ satisfies $Pr_{x \in D} [c(x) = h(x)] \geq 1 - \varepsilon$

Decision List
$(x_2, 1), (\neg x_3), (x_4, 0), (1, 1)$

Theorem (Decision List) There is an efficient (proper) PAC-learning algorithm for Deicision List.

2 Today's topic

1. Finish decision list learning
2. Agnostic vs. Realizable Learning
3. Empirical Risk Minimization
4. Uniform Convergence

3 Decision List learning (continued)

$$\begin{gathered} \mu = E\left[\frac{1}{m}\sum_i X_i\right] \\ = \frac{1}{m}\sum_i E[X_i] \\ \text{where } E[X_i] = p \cdot 1 + (1-p) \cdot 0 = p \\ = \frac{1}{m}\sum_{i=1}^{m} p \\ = \frac{1}{m} \cdot m \cdot p = p \end{gathered}$$

Max size of decision list with log is $2n$ literals
$\Rightarrow$ can represent Decision List in $O(nlogn)$ bits.

$$\begin{gathered} A^{EX}(n, s, \varepsilon, \delta) \text{ runs in time} \\ spoly\left(n, s, \frac{1}{\varepsilon}, \frac{1}{\delta}\right) \\ poly\left(n, 2^n, \frac{1}{\varepsilon}, \frac{1}{\delta}\right) \\ \Rightarrow \text{exceeds any poly time bound} \\ poly\left(n, 1, \frac{1}{\varepsilon}, \frac{1}{\delta}\right) \end{gathered}$$

1) How can you tell if s was guessed too small?

Using $m \geq O\left(\frac{1}{\varepsilon}\left(n \log n + log\, \frac{1}{\delta}\right)\right)$ examples, it suffices to find a list h consistent with c since then w.p. $1-\delta$.

$$Pr_{x \in D}\left[c(x) = h(x)\right] \geq 1 - \varepsilon$$

For a literal l, and a set of examples S, let

$$S_l = \{(x, b) \in S : l(x) = 1)\}$$

We say that l is useful for S if
1) S_l is nonempty
2) every $(x, b) \in S_l$ has the same label ($b = 0$ or $b = 1$)

Algorithm 1 FindList(S)

Let h be the empty list
while (Every example s does not have the same label) **do** ▷ $\leq m$ iteration
  Iterate over the literals l until we find a useful l ▷ $2n|s| \leq m$
  For the common label b of S_l, add (l, b)
  Set $S \leftarrow S \setminus S_l$ ▷ m
  Comment: each iteration takes $O(nm)$ time
end while
For the common label b of S_l, add (l, b)
Return h

Overall time: $O(nm^2) \rightarrow$ poly in $n, \frac{1}{\varepsilon}, log\, \frac{1}{\delta}$

1. Why is there always a useful literal?
2. Why is there final h consistent?

Inductively, S is the set of examples that reach the end of h we've constructed so far.
Empty List, full S trivial
By (2) of "usefulness," since every example reaching end of list and satisfiying l has same label

1. $s \leftarrow s \setminus s_l$ is the set of examples reaching the end of $h \rightarrow l \rightarrow$
2. $h \rightarrow l \rightarrow b$ is labeling every example in S_l correctly (b) (every example that has been classified by$h \rightarrow l \rightarrow b$ is also labeled correctly)

Final set of examples all had label b, Added $\rightarrow b$ to h

3.1 Why is there always a useful literal?

Consider the first literal of l^* of c (unknown list) that I haven't added to h yet. Let s' be the set of examples reaching the end of c, let S'' be the set of examples reaching the end of h. I know that all of the literals up to l^* of c are also tested by h. So $S''_l \subseteq S'$ So therefore, $S''_l \subseteq S'_l$ and since S'_l was useful for c, either $S''_l$is empty(advance in c) or S''_l is nonempty $\Longrightarrow S''_l$ is useful.

4 Agnostic vs. Realizable Learning

4.1 Recognizing "games"

Wittgenstein - ".. can call almost any "rule following" behavior a game."

But this (probably) isn't how we learn what games are

- "Simple pattern in behavior"
- "having fun"
- "score"

$\Rightarrow$ roughly, not perfectly indicative of "game"

We might hope to find a simple h s.t. $Pr_{x\in D}\left[h(x) = "game"(x)\right] \geq 1-\varepsilon$ where D is "scenes in life," but (presumably) no h^* satisfies $Pr_{x\in D}\left[h^*(x) = "game"(x)\right] = 1$

Definition The agnostic learning problem for a class of representations $C = U_{n\in N}C_n$(where C_n is over n attributes) is, given as inputs, n, $\frac{1}{\varepsilon}$, and $\frac{1}{\delta}$, and given access to examples drawn from an arbitrary distribution D and labeled by an arbitary Boolean f, produce as output, an efficiently evaluatable h such that it is within probability of $1-\delta$.

$$Pr_{x\in D}[h(x) \neq f(x)] \leq \min_{c\in C_n} Pr_{xinD}[c(x) \neq f(x))] + \varepsilon$$

More generally we may also consider an $\alpha(n, \varepsilon, \varepsilon^*)$ - approximate agnostic learning task where for $\varepsilon^* = \min_{c\in C_n} Pr[c(x) \neq f(x)]$. we (only) satisfy

$$Pr[h(x) \neq f(x)] \leq \alpha(n, \varepsilon, \varepsilon^*)$$
$$\text{e.g., } \alpha(n, \varepsilon, \varepsilon^*) = n\sqrt{\varepsilon^*} + \varepsilon$$
$$\Rightarrow \varepsilon^* \text{ possibily but still bounded}$$

Notice usual PAC-learning task is a special case of agnostic learning where $\varepsilon^* = 0$ We may refer to the case of $\varepsilon^* = 0$ as "realizable" learning (actually is some $c \in C_n$ such that $c(x) = f(x)$ for $\forall$ x)

5 Empirical Risk Minimization

Definition The empirical error of a hypothesis h on labeld ex's

$$(x^{(1)}, b^{(1)}), \cdots, (x^{(m)}, b^{(m)}) \text{ is } \hat{err}(h) = \frac{1}{m}\left|h : h(x^{(1)}) \neq b^{(1))}\right|$$

An Empirical Risk Minimizer (ERM) of a class H and a list of ex's is any $h^* \in$H such that $\hat{err}(h^*) = \min_{h\in H} \hat{err}(h)$ (realizable learning with $H \Rightarrow \exists h^*$ such that $\hat{err}(h^*) = 0 \Rightarrow$ an ERM is just a solution to $Consis_H^B$)

6 Uniform Convergence

Idea $\hat{err}(h)$ is a good estimate of $\Pr_{x\in D}[h(x) \neq f(x)]$ as long as we have enough examples (as long as we have "representative" data)

Definition For a concept f, a representation class H, a distribution D, and a list of examples, the examples are ε-representative if for every $h\in H$, we have

$$\left|err(h) - \hat{err}(h)\right| \leq \varepsilon$$
$$\Pr_{x\in D}[h(x) \neq f(x)]$$

Lemma For an $\frac{\varepsilon}{2}$-representative sample, any ERM h^* staisfies

$$err(h^*) \leq \min_{h \in H} err(h) + \varepsilon$$

(hence is a solution to the agnostic learning task for H)
Proof If the data is $\frac{\varepsilon}{2}$-representative, fix any h, we find

$$\begin{gathered} \text{for every } n \Rightarrow err(h) \leq \hat{err}(h^*) + \tfrac{\varepsilon}{2} \text{ and } \hat{err}(h^*) \leq err(h) + \tfrac{\varepsilon}{2} \\ \Rightarrow \min_{h \in H} \hat{err}(h) \leq \min_{h \in H} err(h) + \tfrac{\varepsilon}{2} \end{gathered}$$

so for an ERM

$$\begin{gathered} err(h^*) - \tfrac{\varepsilon}{2} \leq \hat{err}(h^*) \leq \min_{h \in H} \hat{err}(h) \\ err(h^*) \leq \min_{h \in H)} err(h) + \varepsilon \end{gathered}$$