Last time: 0clay
  • review w1, w2, P; Finish Perceptron application
  • Dual Perceptron, "kernelization"

Today:  • Start generic bounds for online learning
  - Halving alg
  - Randomized HA for oblivious adversaries
  - VC Dimension negative results

Questions?

**GENERIC BDS FOR OLMB LEARNING**

lower bounds

Goal: alg, l.b.'s that hold for any finite l1L.

**Halving Algorithm** (H.A.)

• Let CONSIST be the subset of C consisting of all c ∈ C that were consistent with all l.b. ex.
(x, c(x)) seen so far.
  • Given ex. x: H.A. predicts according to maj
vote over c ∈ CONSIST. (if |CONSIST| = 2000 +
1150 say 1 on x 3  H.A. predicts 1 on x.
850 say 0 on x )
• On a mistake update CONSIST. (If c(x) was
0, next |CONSIDER| is 850.)

Theorem: For any finite $C$, m.b. of H.A. $\leq \log_2 |C|$.

Proof: Each mistake of H.A. causes size of $|CONSIDER|$ to be mlt. by some $p$, $0 < p \leq \frac{1}{2}$.
Once $|CONSIDER| = 1$, no more mist. So...

Example: $C =$ all length-$r$ DL's over $\{0,1\}^n$.

$$|C| \approx (4n)^{r/2}$$

$$\log |C| = O(r \log n).$$

So H.A. makes $O(r \cdot \log n)$ mist. $\Rightarrow$ But...

Our "levels" alg: $O(n \cdot r)$ m.b. $\Rightarrow$ poly$(n)$ runtime per trial

Winnow: $O(2^r \cdot \log n)$ $\Rightarrow$ ""

Q: Is there an alg w/ $O(r \cdot \log n)$ m.b. $\Rightarrow$ poly$(n)$ time per trial?

???

Back to H.A. in general: good m.b. ! ʕ•ᴥ•ʔ
1. But slow to run: on each trial, presumably need time \( \geq |\text{CONSIST}| \) (to eval. \( c \) on \( x \) for each) & this is \( |e| \) — very slow.

2. Uses weird hyp's: \( \text{MAJ}(c_1, c_2, \ldots, c_n) \)

2. Brittle if noisy examples. (We'll do a noise-tolerant version soon.)

\[
x \in X
\]

\[
\begin{array}{c}
{c_1} \quad {c_2} \\
{c_1'} \quad {c_2'} \\
\vdots \\
{c_n} \quad {0}
\end{array}
\]

For some \( e_i \)'s, H.A.'s n.o. is best possible!

\[ C = \text{all } Z^{1 \times 1} \text{ with } x \mapsto 0,1 \]

H.A. makes \( |X| = \log |e| \) mistakes.

No alg. can do better.

H.A. sometimes does even better than \( \log |e| \) mistakes.

Ex: \( X = \{1, \ldots, N\} \)

\(|e| = N, \quad c_i(j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \)

\[ e \in X \]
MAJ vote over $c$, $0$
After 1st mistake: no
more mistakes.

Randomized H.A. (for "oblivious adversary")

M.B. criterion: very worst-case; $\Xi$ to assuming
see of lab. ex. gen. by omniscient online adversary.
At trial $i$, adv. knows everything about learner;
can use any $c$ consistent w/ first $i-1$ labels.
Too paranoid?

Can relax model: assume target $c$ &
seq. of ex. $x', x^2, \ldots$ are chosen "once & for all" at
start of process. "oblivious adversary"

Makes it possible for learner to fruitfully
use randomness to her advantage.

(RHA)

Randomized H.A. (for obliv. adversaries)

- Updates hyp. $h$ only on mistakes
- Each time mistake made: update $\text{CONSIST}$
  &
pick $h$ to be a randomly selected $c \in \text{the new CONSIST}$.
(Initial $h$: a uniform random $c \in c$.)
Then

\[ \mathbb{E}[\text{#mist. of RHA}] \leq \ln|\mathcal{E}| + O(1) \]

Proof: Have

- Fixed \( c, x, x', \ldots \)

Consider point in RHA execution where \(|\text{CONSIST}| \leq r\).
Let \( M_r = \mathbb{E}[\text{#mist. RHA makes from then on, with the fixed } c \text{ & ex seq.}] \)

Note \( M_{|\mathcal{E}|} = \text{our goal} = \mathbb{E}[\text{#mist. of RHA}] \).

Conceptual step for analysis:
Order concepts in \( \text{CONSIST} \) by when they'll be eliminated by remaining mistakes:

\[ c_1, c_2, c_3, c_4, \ldots, c_{13}, \ldots, c_r = c \quad \Rightarrow \quad r = |\text{CONSIST}| \]

This is well-def. by oblivious adversary assumption.

Consider RHA chooses rand. \( h \in \{c_{r-1}, \ldots, c_r\} \).

- If \( h = c_r \): 0 more mistakes
- If \( h = c_e \) some \( t < r \): 1 mistake eliminates \( c_1, \ldots, c_t \) ( & maybe more). Then (at most) \( r - t \) concepts in new \( \text{CONSIST} \), \( \mathbb{E}[\text{#mist. from then on}] \leq M_{r-1} \).
So

\[ M_r \leq \sum_{t=1}^{r-1} \frac{1}{r} \left(1 + M_{t-r}\right) \]

We pretend is = (okay). Get

\[ M_r = \frac{r-1}{r} + \frac{M_1 + M_2 + \ldots + M_{r-1}}{r} \]

\[ r \cdot M_r = (r-1) + M_1 + \ldots + M_{r-1} \]

\[ (r-1) M_{r-1} = (r-2) + M_1 + \ldots + M_{r-2} \]

\[ r(M_r - M_{r-1}) + M_{r-1} = 1 + M_{r-1} \]

so

\[ rM_r - rM_{r-1} = 1 \quad \Rightarrow \quad M_r = M_{r-1} + \frac{1}{r} \]

\[ M_1 = 0, \quad M_2 = \frac{1}{2}, \quad M_3 = \frac{1}{2} + \frac{1}{3}, \ldots \]

\[ M_r = \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \ldots + \frac{1}{r} = \ln r + O(1) \]

\[ r = \|e\| \quad M_{\|e\|} \leq \ln \|e\| + O(1). \]

"\|e\|" better than \(\log_2 \|e\|\)

"simpler alg.

"hyp. nicer

"still insufficient \"more assumptions, etc.\"
Generic

Lower bound for LMB learning: VC Dimensions

Vapnik-Chervonenkis Dimension: a numerical parameter of a concept class $C$. $VCOIM(C)$ $VC(C)$.

A way of measuring "how complex" $C$ is.
Important in learning theory (next model too)!

**Def:** Fix $C$ over $X$. Let $S \subseteq X$.

$S$ is shattered by $C$ if $\forall U \subseteq S$, $\exists c \in C$ s.t. $c \cap S = U$ (subset p.o.v.)

Equivalently,

$c(x) = 1$ \hspace{1cm} $x \in U$
$c(x) = 0$ \hspace{1cm} $x \in (S \setminus U)$

"No matter how you want to label the pts in $S$, some $c \in C$ does the job"

$VCOIM(C) =$ size of largest $S \subseteq X$ that is shattered by $C$. 