Last time: some more examples of $\mathcal{C}$'s (DNFs, LTFs)

- FIRST LEARNING MODEL:

  Online Mistake-Bound Learning (OLMB)

- Elimination alg. for monotone disjunctions (+ more)
- Decision lists: what they are

Today: More OLMB alg's for specific concept classes.

- OLMB alg. for learning DL's: $O(nr)$ mist. bound for learning 1-DL's of length $r$ over $\{0,1\}^n$
- Winnow1 alg. for learning length-$k$ monotone disj. over $\{0,1\}^n$ with $O(k \cdot \log n)$ mist. bound
- (maybe start) Winnow2 alg: OLMB alg. for certain LTF's

$\boxed{\text{LTFs}}$

Admin: - PS1 out today, due in 2 weeks
- See web page for TA OH's

$\boxed{\text{Questions?}}$

Here's an alg. to learn length-$r$ DL's with $O(n \cdot r)$ mist. bd.

Alg's hyp: a (redundant) DL with "extra structure".

Hyp. DL always has all $4n+2$ rules, grouped into levels.

Each level can have $\geq 1$ rules.

Within a level, rules are lex. ordered.

Init. h: only 1 level. It's level 1.

$$\overbrace{\begin{array}{cccccccccccccccc} x_1 & \to & x_1 & \to & \bar{x}_1 & \to & \bar{x}_1 & \to & x_2 & \to & \cdots & \bar{x}_n & \to & 0 & \to & 1 \\ \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & & \downarrow & & & & \\ 0 & & 1 & & 0 & & 1 & & 0 & & & 1 & & & & \end{array}}^{\text{level 1.}}$$

[ level 1 ] → [ level 2 ] → . . .

That's our h: (initially) the above DL.

Update rule: Given $x \in \{0,1\}^n$,
recall $h(x)$ outputs output bit of first rule "if $\ell \to b$" whose $\ell$ is satisfied. Say that rule lived in level $i$ in $h$.

- if $h(x) = c(x)$: :) no change to $h$
- if $h(x) \neq c(x)$: move the rule used (that caused mistake) down to level $i+1$. (demotion)

---

Ex:

$$h = \overbrace{\begin{array}{ccccc} x_3 & \to & \bar{x}_4 & \to & x_7 \\ \downarrow & & \downarrow & & \downarrow \\ 1 & & 0 & & 1 \end{array}}^{\text{level 1}} \to \overbrace{\begin{array}{ccccc} x_2 & \to & x_4 & \to & \cdots \\ \downarrow & & \downarrow & & \\ 1 & & 0 & & \end{array}}^{\text{level 2}}$$

$x = 0000111\ldots$ ; $c(x) = 1$

nope! $h(x) = 0$ $\quad$ $h(x) = 0$;

$\boxed{\begin{array}{c}\bar{x}_4 \\ \downarrow \\ 0\end{array}}$ made us wrong !!

Move it to level 2...

New h:

$$\overbrace{\begin{array}{ccc} x_3 & \to & x_7 \\ \downarrow & & \downarrow \\ 1 & & 1 \end{array}}^{\text{level 1}} \to \overbrace{\begin{array}{ccccc} x_2 & \to & x_4 & \to & \bar{x}_4 \\ \downarrow & & \downarrow & & \downarrow \\ 1 & & 0 & & 0 \end{array}}^{\text{level 2}} \to$$

---

<u>Thm:</u> Above alg. has MB $O(n \cdot r)$ when target c is a length-r DL.

<u>Pf</u>: Let c = target DL.

$$\overbrace{\begin{array}{ccccccc} x_3 & \to & x_4 & \to & x_2 & \to \cdots & x_7 \\ \downarrow & & \downarrow & & \downarrow & & \downarrow \\ 0 & & 1 & & 1 & & 0 \end{array}}^{r \text{ rules.}} \to 1$$

- 1st rule $\begin{array}{c} x_3 \\ \downarrow \\ 0 \end{array}$ (in target concept c) never moved below level 1:
  whenever its lit. is sat, its b is the right value, so no demotion for it.

- 2nd rule $\begin{array}{c} x_4 \\ \downarrow \\ 1 \end{array}$ (in target c) never moved below level 2:
  if x "reaches" level 2, 1st rule of c didn't apply, so if 2nd rule's lit is sat, its b is the right value, so can't cause a mistake.

- induct., $i^{th}$ rule in c never moved below level i.

So no rule in c – including final $\to 1$ – will ever go below level $(r+1)$.

So no rule at all ever go below level $r+2$. (never use rule at level $r+2$ or greater to predict).

Every mist: a rule moves down a level.

$4n+2$ rules, each moves $\le r+1$

$\Downarrow$

$$\text{tot \# mist.} \le (4n+2)(r+1)$$
$$= \boxed{O(nr).} \to poly(n)$$

$\Downarrow$ Computationally efficient

each trial poly(n)

Learning Sparse (Mon) Disj: Winnow 1 alg.

Elim. alg. for mon disj over $\{0,1\}^n$:
mist. bd. $= n$.

What if target mon disj is only over $k \ll n$ vars?

Can we do better?

$\mathcal{C}$ = all length-$k$ mon disj.

Winnow1: $O(k \cdot \log n)$.

Hyp. of Winnow1: an LTF.

$\boxed{\mathbb{1}[w \cdot x \geq \theta]}$ $\quad w \in \mathbb{R}^n,\ \theta \in \mathbb{R},$
$x \in \{0,1\}^n.$

Quick LTF interlude:

- LTFs can express conj nonmon mon disj:

"$x_1 \vee x_2 \vee \dots \vee x_k$" $\qquad \ell_1 \vee \ell_2 \vee \dots \vee \ell_k$

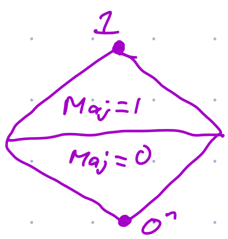
$\mathbb{1}[x_1 + \dots + x_k \geq \frac{1}{2}]$ $\qquad \mathbb{1}[\tilde{\ell}_1 + \dots + \tilde{\ell}_k \geq \frac{1}{2}]$

$$\tilde{\ell}_i = \begin{cases} x_i & \ell_i = x_i \\ 1 - x_i & \ell_i = \overline{x_i} \end{cases}$$

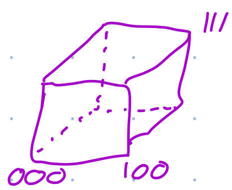
$x_1 \wedge \dots \wedge x_k$ : $\mathbb{1}[x_1 + \dots + x_k \geq k - \frac{1}{2}]$

- Maj fn: $\quad \mathrm{Maj}: \{0,1\}^n \to \{0,1\}$

$$\mathbb{1}[x_1 + \dots + x_n \geq \tfrac{n}{2}]$$




$\{0,1\}^n$



- Any 1-DL can be expressed as an LTF:

$$\begin{array}{ccccc} x_1 & \longrightarrow & x_2 & \longrightarrow & \cdots \\ \downarrow & & \downarrow & & \\ 1 & & 0 & & \end{array}$$

$$2^n \cdot x_1 + 2^{n-1} \cdot x_2 + \cdots \qquad \text{etc.}$$

Diff. variants of Winnow 1; here's a simple one:

- $h_{init}$: $\mathbb{1}[w \cdot x \geq \theta]$, with $w = (1, \dots, 1)$ & $\theta = n$ (always $n$)
- Updates:
  - if $h(x) = c(x)$, ☺ no update
  - if $h(x) = 1$ & $c(x) = 0$ (false pos):
    "demotion step": $\forall\, i \in [n]$ s.t. $x_i = 1$,
    set $w_i = 0$.
  - if $h(x) = 0$ & $c(x) = 1$ (false neg):
    ✓ "promotion step": $\forall\, i \in [n]$ s.t. $x_i = 1$,
    set $w_i \leftarrow 2 \cdot w_i$ (double it).

$[n] = \{1, \dots, n\}$

---

$w$ init. is $(1, 1, 1, \dots, 1)$

Eventually, could be $(0, 0, 4, 1, 0, 8, \dots)$

---

Intuition: • in dem. step, had $c(x) = 0$;
↳ non disj.

all $i$ st $x_i = 1$ can't be in $c$, makes sense to kill those $w_i$'s.

• in prom. step: $c(x) = 1$,

but our $w \cdot x$ was $< n$ $(h(x) = 0)$.

Makes sense to <u>increase</u> wts $w_i$ s.t. $x_i = 1$.

---

<u>Lem:</u> 1) Every $w_i \geq 0$ always. ✓

2) In each prom. step, at least one $w_i$ s.t. <u>$x_i \in c$</u> gets promoted. ✓

3) No $w_i$ is ever $> 2n$. ✓

<u>Pf</u>: if $w_i \geq n$, it'll never again be prom.

(prom. only if $h(x) = 0$ + $x_i = 1$, but if $w_i \geq n$ + $x_i = 1$, $h(x)$ will be 1 not 0).

---

<u>Lem:</u> Tot # prom. steps is $\leq k \cdot \log(2n)$

$p \leq k(1 + \log n)$

<u>Pf</u>: Each prom. doubles $\geq 1$ of $\leq k$ weights in $c$;

each weight doubled $\leq \log(2n)$ times. ∎

Lem: Let $d = \#$ dem. steps

$p = \#$ prom. steps.

Then $d \le p + 1$.

Pf: Let $W = w_1 + \dots + w_n$. Initially $W = n$.

- Each dem. step: $w \cdot x$ was $\ge n$
  + we "zero out" exactly $w \cdot x$ of $W$.
  So $W$ decr. by $\ge n$ on each dem.

$w_1 = 2 \quad x_1 = 1$

$w_3 = 8 \quad x_3 = 1$

$\Downarrow$ prom

$w_1 = 4,$

$w_3 = 16$

- Each prom. step: $w \cdot x$ was $< n$
  + we increasing $W$ by $w \cdot x$
  So $W$ incr. by $< n$ on each prom.

After $p$ prom. + $d$ dem., $W_{init} = n$, $W$ never $< 0$,

have

$$0 \le W \le n - dn + pn.$$

So $d\cancel{n} \le \cancel{n} + p\cancel{n}$

$$d \le p + 1.$$

---

So $d + p \le 2p + 1 \le 2k(1 + \log n) + 1$

$= O(k \cdot \log n).$

Next time: learning LTFs with
Winnow2 + Perceptron