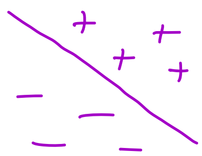
Last time:

examples of $\mathcal{C}$'s we'll consider:
- s-term DNFs, s-clause CNFs, k-DNFs, k-CNFs
- LTFs $\text{sign}(w \cdot x - \theta)$



- OLMB model
- Elimination alg. for mon. conj., variants

mistake bound : n

$\{0,1\}^n$

- 1-decision lists (def.)

Today: More examples of OLMB algs for specific $\mathcal{C}$'s.

- $O(nr)$-mistake-bound alg for 1-DL of length r over $\{0,1\}^n$ $O(n^2)$ for any 1-DL
- Winnow1 alg. for sparse monotone disj:
  $O(k \cdot \log n)$-mistake bound for k-sparse monotone disj. over $\{0,1\}^n$
- (maybe start) Winnow2 alg. for certain LTFs

Admin. stuff:

- PS1 out, due in 2 weeks.
- Rocco's OH today 2–4 (not Fri.)

Questions?

Recall:

$$\begin{array}{ccc} \text{true} & x_7 & \bar{x}_3 \\ \downarrow & \downarrow & \downarrow \\ 1 & 0 & 1 \end{array}$$

$\boxed{\text{Total \# of possible rules, including the two "default rules", is } 4n+2}$

Here's a learning alg:

Hyp is a 1-DL, "with structure":

- rules in list are partitioned into layer 1, layer 2, ...;
- all $4n+2$ always present in rule;
- within a level, rules are lex. ordered.

initial h: all rules in level 1.

initial h.

$$\underbrace{\begin{array}{ccccccccccc} x_1 & \rightarrow & x_1 & \rightarrow & \bar{x}_1 & \rightarrow & x_1 & \rightarrow & x_2 & \rightarrow & \dots \\ \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \\ 0 & & 1 & & 0 & & 1 & & 0 & & \end{array}}_{\text{level 1}}$$

Alg: • if $h(x) = c(x)$ (after example x), no update.

• if $h(x) \neq c(x)$, take the rule that was used to make the (incorrect) pred. on x, move to next level.

: move to level 3

Thm: Above alg makes $O(n \cdot r)$ mist.

when c is a 1-DL of length r.

Pf:

Target:
$$\begin{tikzcd} \ell_1 \arrow[r] \arrow[d] & \ell_2 \arrow[r] \arrow[d] & \cdots \arrow[r] & \ell_r \arrow[r] \arrow[d] & b_{r+1} \\ b_1 & b_2 & & b_r & \end{tikzcd}$$

- 1st obs: the 1st rule in c, never goes below level 1 (if its lit ℓ_1 holds, b_1 is the c(x) value)

- Conseq, the 2^{nd} rule in c, never goes below level 2: (for this rule to get pushed below level 2 by an example x, would need to be the case that no rules in level 1 applied; so didn't apply; but then if applies, b_2 is c(x) - no mistake.)

- Inductively: i^{th} rule in c never moved below i.

So no rule in c ever pushed below level r+1. (including default "true" rule).

"true" $\to b_{r+1}$

Hence no rule at all ever goes below r+2 (b/c in level ≤ r+1 always).

☆ Each mistake moves a rule down a level.

4n+2 rules, each moves ≤ r+1 : ≤(4n+2)·(r+1) "movements", +)

So tot # mistakes ever made must be $\leq (4n+2)\cdot(r+1)$
$= O(n \cdot r)$.

 Comput. efficient!

(Each stage: $O(n)$ time.)

Learning Sparse Disjunctions:

Winnow1

(Mon. disj.)

Recall: "elim" alg over $\{0,1\}^n$: m.b. $= n$.

What if $n = 10^6$ + target disj is of length $k = 5$? Can we do better?

Winnow1: $O(k \cdot \log n)$ m.b. for length-k mon. disj.

Alg's hyp.: a LTF

$h(x) = \mathbb{1}[w \cdot x \geq \theta]$

$w_1, \dots, w_n, \theta \in \mathbb{R}$.

Sane? Yes:

$x_1, \dots, x_n \in \{0,1\}$ vars.

Here are some LTFs:

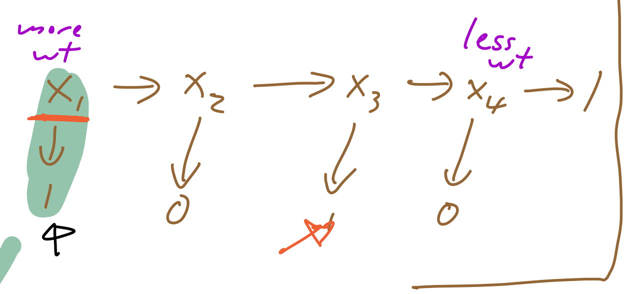
$\mathbb{1}\{\underline{x_1 + \dots + x_k} \geq \frac{1}{2}\}$: disjunction. OR

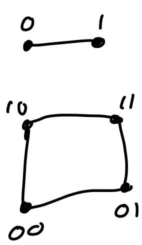
$\mathbb{1}[\underline{x_1 + \dots + x_k} \geq k - \frac{1}{2}]$: conj AND

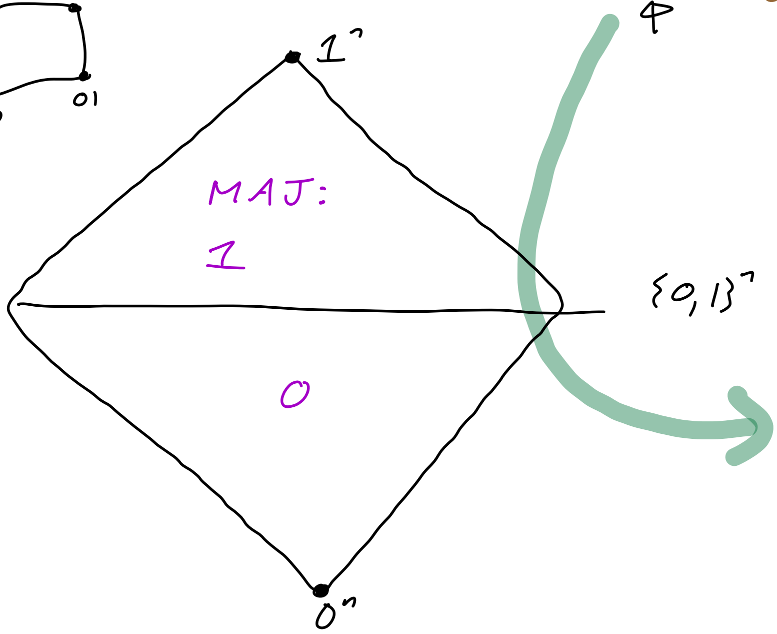
$\mathbb{1}[\underline{x_1 + \dots + x_k} \geq \frac{k}{2}]$: Majority

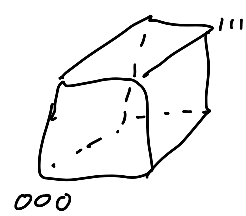
Can also express any 1-DL as an LTF.

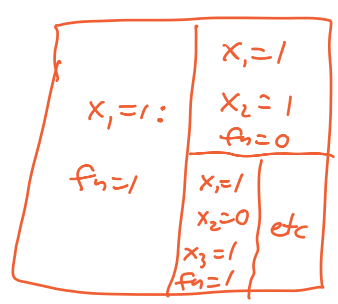
(think about it...)











Here's (a) Winnow1 (variant):

Init h: $h(x) = \mathbb{1}[x_1 + \dots + x_n \geq n]$

$w_i = 1 \ \forall i$

$\theta = n$ (always!)

Updates: on example x,

as usual, • if $h(x) = c(x)$, no change to h.

• if $h(x)=1$ & $c(x)=0$ (f. pos.), "demotion step": for each i s.t. $x_i = 1$, set $w_i = 0$. [Like elim. alg.]

• if $h(x)=0$ & $c(x)=1$ (f. neg.), "promotion step": for each i s.t. $x_i = 1$, set $w_i = 2 \cdot w_i$.

[Makes sense: $w \cdot x$ was $< n$, should have been $\geq n$.]

<u>Lemma 1:</u>

1) No weight w_i ever negative. ✓

2) In each prom. step, at least one var x_i in c will get promoted (its w_i doubled) ✓

3) $\forall i = 1, \dots, n$, always have $w_i \leq 2n$.

Pf: w_i only promoted if $w_i < n$ (if $w_i \geq n$ & $x_i = 1$, $w \cdot x$ is $\geq n$ & no false neg mistake), so never $> 2n$. ∎

<u>Lemma 2:</u> Total # <u>prom.</u> (false neg) $\leq k \cdot \log(2n)$.

<u>Pf</u>: Observe no x_i that's in c ever gets

denoted (demotion only happens if no var in c is 1).

So for an x_i in c, w_i is

never reaches $2^{\ell+1} > 2n$

$$\underbrace{1,\dots,1,\ 2,\dots,2,\ 4,\dots,4,\ \dots\ 2^{\ell},\dots\dots}_{\le \log(2n) \text{ times when it's promoted.}}$$

By this (*2), total # prom. steps $\le k \cdot \log(2n)$. ∎

<u>Lemma 3:</u> Let d = demot. steps

p = promot. steps.

Always have $d \le p+1$.

<u>Pf</u>: Let $W = w_1 + \dots + w_n$

Initially $W = n$

- At each demot. step on example x:

  $w \cdot x = w_1 x_1 + \dots + w_n x_n$ was $\ge n$

  $= \sum_{i:\, x_i = 1} w_i$ + these w_i's are set to 0; so $W \downarrow$ by $\ge n$.

- At each promot. step on example x:

  $w \cdot x = w_1 x_1 + \dots + w_n x_n$ was $< n$

  $= \sum_{i:\, x_i = 1} w_i$ + these w_i's are doubled; so $W \uparrow$ by $< n$.

So after p promotions + d demotions, recalling that W is n initially + $W \geq 0$ always, have

$$0 \leq W \leq n + pn - dn$$

$$dn \leq n + pn$$

$$d \leq 1 + p. \quad \blacksquare$$

Combining lemmas:

<u>Thm:</u> Winnow1 makes

$$\leq 2k \log(2n) + 1 = 2k \log n + 2k + 1$$
$$= O(k \log n)$$

mistakes to learn class of length-k mon. disj.

Comput. efficient!	Attribute-efficient:
:)	$\log n$ dep. on n.

? Is there an alg that's comput. efficient + attribute efficient for length-k DLs?

Next time: learning LTFs:

- Winnow2
- Perceptron