Last time: *OLMB model. Toy alg. for init. int.
- elim. alg. for mon. disj. or conj. \( n, n+1. \)
- defined 1-OL.

Today: *OLMB alg. w/ MB O(\(n^2 \)) for \( C = \text{length-1 1-OL's} \)

\( \overset{\text{over } \{0,1\}}{\xrightarrow{\text{\# poss.}}} \)
- Winnow 1 alg. (k-sparse mon. disj.)
- Winnow 2 alg. for certain mon LTFs.

Readings:
- Blum survey.

\[ \begin{array}{c}
\neg x_4 \rightarrow x_2 \rightarrow x_3 \rightarrow 0 \\
\neg x_4 \rightarrow \neg x_2 \rightarrow \neg x_3 \rightarrow 0 \\
\end{array} \]

Every disj. is a 1-OL:
\[ x_2 \lor x_3 \lor x_4 \equiv 0 \]

But \( x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow 0 \)
\[
\begin{array}{cccc}
1 & 1 & 1 & 0 \\
\end{array}
\]
is not a disj. or conj.

\[ \overset{\text{WLOG, no var. occurs twice in a 1-OL. Need only}}{\xrightarrow{\text{\# poss. for each literal}} \text{1 poss. for each literal}} \]

\[ \overset{\text{\# poss.}}{\xrightarrow{\text{none.}}} \]

\[ \overset{\text{\text{Our alg: MB O(\(n^2 \)) for \( C = \text{all 1-OLs} \)}}}{\xrightarrow{\text{consider 1OLs of}}} \]

\[ \overset{\text{How many 1-OL's of length \( r \) are there?}}{\xrightarrow{\text{\( (4n)^r \) \#}}} \]

\[ \overset{\text{\( 4n+2 \) "rules" \( \overset{\text{\( \cdot \)}}{\xrightarrow{\text{\( b \)}}} \text{or } \overset{\text{\( \cdot \)}}{\xrightarrow{\text{\( b \)}}} \)}}{\xrightarrow{\text{\( 4n \)}}} \]

Our alg. uses hyp's which slightly extend 1OL's.

Hyp has several levels. Each hyp. we consider has exactly
1 copy of each rule. Rules are spread out across levels.
Within each level, view its rules as lexically ordered.
Initial hyp: \( h \) has all \( 4n+2 \) rules in level 1

(lex. ordered)

\[ h_{init} \equiv \begin{array}{c}
\begin{array}{cccc}
\neg x_1 & \rightarrow x_1 & \rightarrow \neg x_1 & \rightarrow x_2 & \rightarrow \ldots & \rightarrow x_n & \rightarrow 0 & \rightarrow 2 \\
\downarrow & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\
\end{array}
\end{array} \]

level 1.
On ex. x: look at level 1 then level 2, etc. Within each level: looks thru rules of that level in lex. order looking for 1st rule whose "if" cond. is sat; uses the corresp. output bit as h(x).

If no rule in current level has its "if" cond. sat, looks at next level.

* Given c(x) + h(x): if h(x) = c(x) no change.
* If h(x) ≠ c(x): take the rule that was just used to come up w/ the bit h(x), try move it down to next level.

Ex: Say 1st ex 
\[ x = \overline{101010} \]

\[ c(x) = \overline{1} \]

\[ h_{init} (x) = \overline{0} \]

| update rule |

level 2.

Thm: This alg. makes \( \leq (4n + 2) \cdot r = O(nr) \) mist. on any I-OC of length \( \leq r \).

Pf: Let c(x) be target I-OC. 

Claim: 1st rule in c is never pushed down to level 2; whenever its "if" cond. (xₙ) is sat, its pred. is.

So h always has in level 1. Given this, the 2nd rule in c is never pushed below level 2 in h: if its at level 2 then ex x causes level 2 of h to be consulted, no level-1 rule in h had its "if" cond. satisfied; in this case rule 2 of c is the actual rule labeling this x so doesn't cause mistake.

Inductively, no rule in c at level i is ever pushed below level i in h. This means no rule at ALL is ever pushed below level n, (some rule in c always is sat).

Each one of the \( 4n + 2 \) rules is moved \( \leq r \) levels. 

Each mist. moves one rule down one level. \( \leq (4n + 2) \cdot r \)

So \( \leq (4n + 2) \cdot r = O(nr) \) mist. in total.

Alg. is computationally efficient: \( O(n) \) time per stage.

Another specific alg/class: WINNOW alg. for K-sparse mon. disj. over \( \{0,1\} \).

A disj c is K-sparse if its an OR of \( \leq K \) literals.

\[ c(x) = x_2 \lor x_4 \lor x_{10} \lor x_{42856914} \]

3-sparse mon. disj. e.i.m.
Think if $\eta$ very small or $n$ very large. Choose $C$ mon disj. over $S_1, S_2$. Better MB than $\eta$

**WINNOW** I: m.b. $O(k \log n)$ attr. lea.

**(Read: Littlestone Winnow paper)**

Hyp. that $W_1$ uses: a LTF $h(x) = \prod [w_i x_i > \theta] \in \{0,1\}$

Quick LTF culture/facts: Any mon disj. can be expressed as an LTF:

$$x_1, x_2, x_3, x_4 \quad \equiv \quad x_1 + x_2 + x_3 + x_4 \geq \frac{1}{2}$$

$$\overline{x_1}, \overline{x_2}, \overline{x_3}, \overline{x_4} \quad \equiv \quad (1 - x_1) + x_2 + (1 - x_3) + (1 - x_4) \geq \frac{1}{2}$$

$$\overline{x_1}, \overline{x_2}, \overline{x_3}, \overline{x_4} \quad \equiv \quad (1 - x_1) + x_2 + x_3 + (1 - x_4) \geq \frac{3}{2}$$

Any $1$ ft. (HW problem)

LTFs can do more: "$r$-out-of-$n$" $h(x) = \prod [x_i \geq \frac{1}{2}]$

$$x_1 + x_2 + \ldots + x_n \geq \frac{r}{2}$$

**WINNOW** I: Init. $h$ is $w_1 = \ldots = w_n = 1, \theta = n$, $h(x) = \prod [w_i x_i > \theta]$. Fixed for $W_1$

Update: if $h(x) = c(x)$, no update.

If FP $(h(x) = 1, c(x) = 0)$

\[ \forall i \in [n] \text{ s.t. } x_i = 1 \]

Set $w_i = 0$. (demotion)

If FN $(h(x) = 0, c(x) = 1)$

\[ \forall i \in [n] \text{ s.t. } x_i = 1 \]

double $w_i$. $w_i \Leftarrow 2 \cdot w_i$. (promotion)

$w_i$'s always: $1, 2, 4, 8, \ldots$
Claim: \( \forall w, \text{ ever } < 0 \). \( \forall w, \text{ doubled } \) \( \forall w, \text{ always } \leq 2n \). (Only way \( w \) doubled is if \( w \cdot x \leq n \) \& \( x \text{ was } 1 \). so \( w \) was \( < n \).)

Lemma: Total number of prom. steps \( \leq k \cdot \log(2n) = k(1 + \log n) \). \( \forall w \cdot x \leq n \) \& \( \forall x \text{ was } 1 \). \( \forall w \text{ was } < n \).

Proof: \( \forall w \in c \) even is denoted. Each of the \( k \) wts is \( \leq \log(2n) \) times. So total number of prom. steps \( \leq k \cdot \log(2n) \).

Lemma: Let \( d = \# \text{ dem. steps, } p = \# \text{ prom. steps.} \)

Have \( d \leq p + n \) (always). \( W = \sum_{i=1}^{n} w_i \).

Proof: \( W = \text{ tot wt of all vars, } W = \sum_{i=1}^{n} w_i \).

\( W \) initially is \( n \).

Each dem step: \( w \cdot x \leq n \), set all \( w_i \) which had \( x_i = 1 \)
to 0; this decreases \( W \) by \( w \cdot x \); so each dem step decreases \( W \) by \( > n \).

Each prom step: had \( w \cdot x \leq n \), & double all \( w_i \)'s; this incr \( W \) by \( w \cdot x \); so prom step incr \( W \) by \( < n \).

So after \( d \) dem + \( p \) prom, have

\[ 0 \leq W \leq n - dn + pn \quad \text{so } d \leq p + n. \]

Proved Thm: W1 MB is

\[ \leq 2k \cdot \log(n) + 1 \quad \text{for } k\text{-sparse mon. disj.} \]
W1, like linear, brittle if noisy labels. Noisy tol. version: less extreme demotions.

Can we use W1 to learn other LTFs?
Consider \(-3x_1 + 4x_2 - 7x_3 + \ldots > 16\)

Can we mon LTFs (all \(w_i > 0\))?

Consider \(x_1 + 2x_2 + 3x_3 + \ldots + i \cdot x_i + \ldots + n \cdot x_n \geq n\)

W1: all its ws \(0, 1, 2, 4, 8, \ldots\)

Need to tweak \(\theta\) W1 to handle

Next time: W2. Can learn LTFs like
with pos. ws.

Perceptron?