Monotone Functions

\[ \text{def partial order } \leq \]
\[ x \leq y \iff \forall i \ x_i \leq y_i \]

monotone function \( f \)

\[ x \leq y \implies f(x) \leq f(y) \]

Learning algorithms for the class of monotone functions?

in homework we see \( 2^n \) random samples suffice for uniform distribution

\[ \mathcal{O}(n^2) \]

Why is this nontrivial?

we said poly samples is easy, but the problem is computation time? poly in what?

but need \( \text{poly}(\log |\mathcal{C}|) \) samples

there are \( 2^n \) functions \( = 2^{2^n} \) functions, so why so many monotone functions?

Consider slice functions:

\[ f = 1 \]
\[ f = 0 \]

\( \text{Set in all possible ways w/o violating monotonicity} \)

\( \implies \) learning needs \( \Omega\left(\frac{2^n}{n^2}\right) \) even with queries in PAC model

a hard distribution: uniform on middle row
Today's question:

What about learning monotone distributions, on uniform distribution, with queries?

Here we will get a very slight "win":

All monotone ftms have weak agreement with some dictator ftm.

Thm. \( f \) monotone, \( \forall g \in \{-1, x_1, x_2, \ldots, x_n \} \), s.t.

\[ \Pr_x [ f(x) = g(x) ] = \frac{1}{2} + \Omega \left( \frac{1}{n} \right) \]


Why does this give a learning algorithm? estimate agreement of \( f \) with all members of \( S \), output member with max agreement

Pf.

Case 1 \( f(x) \) has weak agreement with +1 or -1

Case 2 otherwise \( \Pr [ f(x) = 1 ] \in \left[ \frac{1}{4}, \frac{3}{4} \right] \)

First a break, before we prove case 2 ...

Let's define influence of monotone ftms?

- \# nodes = \( 2^n \)
- \# edges = \( \frac{n \cdot 2^n}{2} \)
- each level has \( \binom{n}{j} \) weight \( j \) nodes
- monotone \( \Rightarrow \) no blue above any red
- slicing cube in roughly half cuts many edges + many in same direction
- \( I(f) = \frac{\# red-blue edges}{2^n} \)
- \( I_A(f) = \frac{\# red-blue edges in 1^{st} \text{dim}}{2^n} \)
Another Definition

Influence of $i^{th}$ var on $f: \mathbb{Z}^n \to \mathbb{Z}^1$:

$$\text{Inf}_i(f) = \mathbb{P}_x [f(x) \neq f(x \oplus e_i)]$$

$\text{Inf}_i(f)$ is flipped

Total Influence:

$$\text{Inf}(f) = \sum_{i=1}^{n} \text{Inf}_i(f)$$

Thm: $f$ monotone $\Rightarrow \text{Inf}_i(f) = f(\exists x_i)$

Thm: Majority fn $f(x) = \text{sign}(\sum x_i)$ (n odd)

maximizes influence among monotone fn's.

Tests on hw.
Recall H.W.: 

If \( f \) is monotone
\[
\ln f^*_{\hat{f}}(f) = \hat{f}(\xi_{\hat{f}}) \uparrow \text{H.W.} \geq 2 \Pr[ f(x) = X_{\xi_{\hat{f}}}(x)] - 1
\]

Plan:

Show \( \ln f^*_{\hat{f}}(f) \geq \Omega(\frac{1}{n}) \)

\[
\Rightarrow \Pr[f(x) = X_{\hat{f}}] \geq \frac{1}{2} + \ln f^*_{\hat{f}}(f) \geq \frac{1}{2} + \Omega(\frac{1}{n})
\]

Will use following tool:

**Canonical Path Argument**

Plan 1) define canonical path for every red-blue pair of nodes (note such a path must cross at least one red-blue edge)

2) show upper bnd on # of c.p.l.s passing through any edge (in particular, any red-blue edge)

3) conclude lower bnd on # of red-blue edges
Part 1 of plan:

\[ \text{def: } \forall(x,y) \text{ s.t. } x \text{ red } \land y \text{ blue} \]

"canonical path from } x \text{ to } y" \text{ is: }

scan bits left to right, flipping where needed
each flip \( \Rightarrow \) step in path

\[
\begin{align*}
x &= \begin{bmatrix} -1 & +1 & +1 & +1 \\ +1 & -1 & +1 & +1 \\ +1 & +1 & -1 & +1 \\ +1 & +1 & +1 & -1 \\ +1 & +1 & +1 & +1 \end{bmatrix}
\end{align*}
\]

\(x \Rightarrow w \Rightarrow z \Rightarrow y\)
each step is Hamming distance 1

How many red-blue pairs have canonical paths?

recall \( \Pr[\|f(x)\|=1] \approx \frac{1}{4}, \frac{3}{4} \]

\( \geq \text{ paths } \geq \frac{1}{4} \cdot 2^n \cdot \frac{1}{4} \cdot 2^n = \frac{1}{16} \cdot 2^n \)
Part 2 of plan:

For any red-blue edge \( e \), how many \( x,y \) pairs can cross it with canonical \( x,y \)-path?

Main point:
all canonical paths crossing \( u,u^2 \) agree on \( y_1 \ldots y_i \) and \( x_{i+1} \ldots x_n \)

\( e = (u^j, u^{2k}) \)

red blue

Main point:
all canonical paths crossing \( u,u^2 \) agree on \( y_1 \ldots y_i \) and \( x_{i+1} \ldots x_n \)

Part 3 of plan:

\((\# \text{ red-blue edges}) \cdot (\# \text{ canonical paths that use it}) \geq (\# \text{ red-blue canonical paths})^{1.6} \cdot (\# \text{ red-blue pairs})\)

\[ \text{So} \]
\[ \# \text{ red-blue edges} \geq \frac{1}{16} \cdot 2^n = \frac{1}{16} \cdot 2^n \]

Since each \( v \) uses \( \geq 1 \) red-blue edge

So there exists \( i \) st. \[ \geq \frac{2^n}{16} \cdot \frac{1}{n} \] red-blue edges in direction \( i \)
So \( \inf_x f(x) = \frac{2^n}{16n} = \frac{1}{8n} = \frac{1}{\frac{1}{2} n} = \frac{n}{2} \cdot \Pr[f(x) = x_i] - 1 \)

\[
\text{total \# edges in dir } i
\]

\[
\Pr[f(x) = x_i] \geq \frac{1}{2} + \frac{1}{16n}
\]

Canonical Path argument also used in
- routing
- expansion / conductance of hypercube / other Markov Chains

What good is weak learning?

Unclear
- here only uniform distribution
- if can learn in all distributions, can do much more

(next result does not apply to monotone function learning... i.e. \( \frac{n}{2} \) agreement
- in particular, this weak notion of learning (i.e. \( \text{const} > \frac{1}{2} \) agreement probably doesn't give anything for stronger learning)