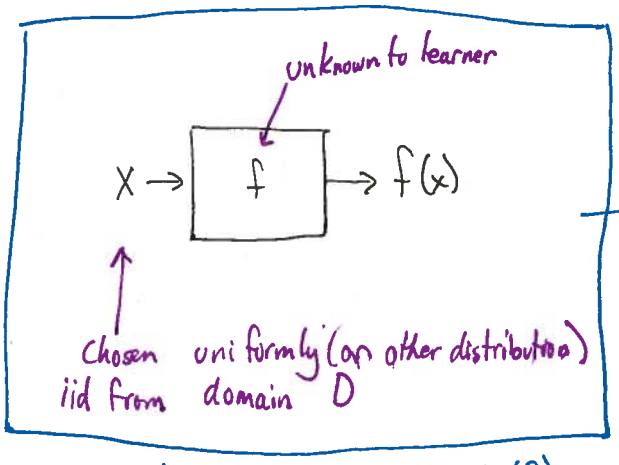
Learning via Fourier Coeffs

- Some fctns & their Fourier representation
- the low degree algorithm
- applications

Learning

Learn from random, uniform examples: How do we formalize?



$X_1, f(X_1)$
$X_2, f(X_2)$
⋮

m random labelled examples

Example oracle $Ex(f)$

After seeing several examples, learner should output hypothesis h.

- hopefully h=f
- is that asking too much?
- how about $dist(h,f) < \varepsilon$?

what is distance?

e.g. $\Pr_{x \in D}[h(x) \neq f(x)]$?

but then what distribution on D?

Today: uniform

In general: match distribution of examples

Valiant's PAC model

"Probably Approximately Correct"

def. given hypothesis h, error of h wrt f is $\text{error}(h) = \Pr_{x \in_u D}[f(x) \neq h(x)]$

Note: this is defn wrt uniform. In general, this is the same as distribution on D from example ora

often will use:
f is ε-close to h wrt D if $\Pr_{x \in_u D}[f(x) \neq h(x)] \leq \varepsilon$

Note if f is arbitrary, there is nothing you can do! (ie can't learn a random fctn)
However, if you know something about f, there may be hope!

What if you know that f is from a family of functions

def. uniform distribution learning algorithm for concept class, C is algorithm A s.t.

- A is given $\varepsilon, \delta > 0$ access to $Ex(f)$ for $f \in C$
- A outputs h s.t. with prob $\geq 1-\delta$ error(h) wrt. f is $\leq \varepsilon$

h is ε-close to f

Parameters of Interest

- m — # samples used by A — "Sample complexity"
- ε — accuracy parameter
- δ — confidence parameter
- runtime? — hope for $\text{poly}(\log(\text{domain size}), \frac{1}{\varepsilon}, \frac{1}{\delta})$
- description of h?
  - should it be similar to description of f?
    (proper learning)
  - at least should be relatively compact
    $O(\log |C|)$ + efficient to evaluate

Remarks

- as before, dependence on δ needn't be more than $O(\log(1/\delta))$.
  why?
- Uniform case is special case of PAC-model:
  Given $EX_{\mathcal{D}}(f)$ for unknown $\mathcal{D}$
  output h with small error according to same $\mathcal{D}$ (some $\mathcal{D}$ can be harder than others)

Ignoring Runtime

Occam's Razor

learning is easy!

i.e. can easily achieve small sample complexity

Brute Force Algorithm

- Draw $M = \frac{1}{\varepsilon}\left(\ln |\mathcal{C}| + \ln \frac{1}{\delta}\right)$ uniform examples
- search over all $h \in \mathcal{C}$ until find one that labels all examples correctly & output it.
  (choose arbitrarily if ≥ 1 such h works)

Behavior:

What should behavior be?

- f is a good thing to output ✓
- what is a bad thing to output?

h is "bad" if error(h) wrt $f \geq \varepsilon$

$\Pr[\text{bad } h \text{ consistent with examples}]$

$\leq (1-\varepsilon)^M$

$\Pr[\text{any bad } h \text{ consistent with examples}]$

$\leq |\mathcal{C}|\,(1-\varepsilon)^M$ ← union bnd

$\leq |\mathcal{C}|\,(1-\varepsilon)^{\frac{1}{\varepsilon}\left(\ln |\mathcal{C}| + \ln \frac{1}{\delta}\right)}$

$\leq \delta$

∴ unlikely to output any bad h

[Does the Bible really predict JFK's assasination?]

Comments

- proof didn't use anything special about uniform distribution

  works for any $\mathcal{D}$,
  as long as error defined w.r.t. same $\mathcal{D}$ as sample generator

- once we have a good h

  1) can predict values of f on new inputs since $\Pr_{x \in \mathcal{D}}[f(x)=h(x)] \geq 1-\delta$

     random ~ according to $\mathcal{D}$

  2) can compress description of samples

     $(x_1, f(x_1))(x_2, f(x_2)) \dots (x_m, f(x_m))$ $\quad m(\log|D| + \log|R|)$

     (ranged of f)

     $\Downarrow$

     $x_1 \dots x_m$, description of h $\quad m \cdot \log|D| + \log|C| k$

So learning, prediction & compression are related.

learning $\Rightarrow$ prediction & compression

formal relations in other direction too

Occam's Razor: simplest explanation is best

An efficient learning algorithm

C = conjunctions over $\{0,1\}^n$

ie. $f(x) = x_i x_j \bar{x}_k$

- can't hope for 0-error from subexponential # of random examples

  eg. how to distinguish $f(x) = x_1 \cdots x_n$ from $f(x) = 0$?

- Brute force: $M = \frac{1}{\varepsilon}\left(\ln(2^n) + \ln\frac{1}{\delta}\right)$ examples & as much time

- Poly time algorithm:

  - draw poly$(1/\varepsilon)$ random examples to estimate $\Pr[f(x)=1]$ to additive error $\pm \frac{\varepsilon}{4}$

    if estimate $< \varepsilon/2$, output "$h(x)=0$"

  - since estimate $\geq \varepsilon/2$ & error $\leq \varepsilon/4$

    $\Pr[f(x)=1] \geq \varepsilon/4$

    so, every $O(1/\varepsilon)$ examples see new random "positive" example (expected) } just look at these

  - in set of positive examples

    let V = {vars set same way in each example}

    output $h(x) = \bigwedge_{i \in V} X_i^{b_i}$ ← b_i tells us if i complemented or not

behavior of algorithm:

for i in conjunction:

must be set same way in each positive example $\Rightarrow$ in V

for i not in conjunction:

$$\Pr[i \in V] \leq \Pr[i \text{ set same in each of } k \text{ positive examples}]$$

$$\leq \frac{1}{2^{k-1}}$$

$$\Pr[\text{any } i \text{ that not in conjunction manages to survive}]$$

$$\leq \frac{n}{2^{k-1}}$$

$$\leq \delta \quad \text{if pick } k = \log \frac{n}{\delta}$$

So $\Omega(\log \frac{n}{\delta})$ positive examples

+ $\Omega(\frac{1}{\varepsilon} \log \frac{n}{\delta})$ total examples suffice!

Learning via Fourier Representation

learning algorithms based on estimating Fourier representation of fctn f (similar to poly interpolation)

Approximating one Fourier coefficient:

lemma can approx any specific Fourier coeff S to w/in additive γ (i.e. $|\text{output} - \hat{f}(S)| \leq \gamma$) with prob $\geq 1-\delta$ in $O(\frac{1}{\gamma^2}\log\frac{1}{\delta})$ samples

Note no queries needed!!

Pf. Chernoff + $\hat{f}(S) = 2\underbrace{\Pr_x[f(x)=\chi_S(x)]}_{\text{estimate this}} - 1$ ∎

Can we find any or all heavy coefficients?

there are exponentially many coefficients.

Can use same samples for all coeffs, but must union bnd prob of error on any of them

Using $\delta = \frac{1}{2^n}$, gives $O(\frac{1}{\gamma^2}\cdot n)$ samples, but exp runtime.

queries can help a lot!

What if we "know where to look" for heavy coefficients?

e.g. all heavy coeffs are in "low degree" coeffs?

If so, can search!

Fourier Representations of Important Examples

Two examples

1) $\overline{\text{AND}}$ on $T \subseteq N$ s.t. $|T| = k$

$$\overline{\text{AND}}(x_{i_1} \cdots x_{i_k}) = 1 \quad \text{if} \quad \forall i_j \in T = \{i_1 \cdots i_k\}$$
$$x_{i_j} = -1$$

-1 0.w.

define $\overline{\text{AND}}$: $f(x) = \begin{cases} 1 & \text{if } \forall i \in T \quad x_i = -1 \\ 0 & \text{o.w.} \end{cases}$ } corresponds to AND fctn over $\{0,1\}$

$$= \frac{(1-x_{i_1})}{2} \cdot \frac{(1-x_{i_2})}{2} \cdot \cdots \cdot \frac{(1-x_{i_k})}{2}$$

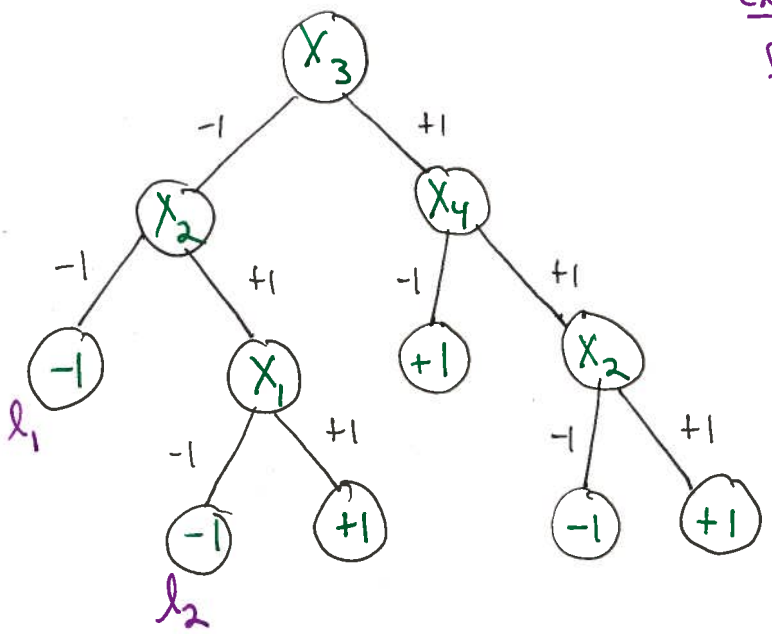
$$= \sum_{S \subseteq T} \frac{(-1)^{|S|}}{2^k} \chi_S$$

* so $\overline{\text{AND}}(x) = 2f(x) - 1$

$$= -1 + \frac{2}{2^{|T|}} + \sum_{\substack{S \in T \\ |S| > 0}} \frac{(-1)^{|S|}}{2^{k-1}} \chi_S$$

Note: all Fourier coeffs containing vars not in T are 0

2) Decision trees



examples

$f_{l_1}(x) = \frac{(1-X_3)}{2} \cdot \frac{(1-X_2)}{2}$

$f_{l_2}(x) = \frac{(1-X_3)}{2} \frac{(1+X_2)}{2} \frac{(1-X_1)}{2}$

First, consider path fctns:

left or right

$$f_l(x) = \prod_{i \in V_l} \frac{(1 \pm X_i)}{2}$$

vars visited on path to leaf l

(-1)^# left turns taken in S

$$= \frac{1}{2^{|V_l|}} \sum_{S \subseteq V_l} (\pm 1) \chi_S$$

$f_l(x) = \begin{cases} 1 & \text{if } X \text{ takes p} \\ 0 & \text{o.w.} \end{cases}$

so $f(x) = \sum_{l \in \text{leaves of } T} f_l(x) \, val(l)$

exactly one of these is 1, others are 0

Comment only coeffs corresponding to S s.t. $|S| \leq$ max path length can be nonzero.

Low degree algorithm

def $f: \{\pm 1\}^n \to \mathbb{R}$ has $\alpha(\varepsilon, n)$- Fourier concentration

if $\sum_{S \subseteq [n] \text{ s.t. } |S| > \alpha(\varepsilon,n)} \hat{f}(S)^2 \leq \varepsilon \quad \forall\ 0 < \varepsilon < 1$

for Boolean f, this implies $\sum_{S \subseteq [n] \text{ s.t. } |S| \leq \alpha(\varepsilon,n)} \hat{f}(S)^2 \geq 1 - \varepsilon$

examples

1) fctn f which depends on $\leq k$ vars has $\sum_{S \text{ s.t. } |S| > k} \hat{f}(S)^2 = 0$

} if f doesn't depend on x_i then all $\hat{f}(S)$ for which $i \in S$ satisfy $\hat{f}(S) = 0$

2) f = AND on $T \subseteq \{1..n\}$ has $\log(\frac{1}{\varepsilon})$-F.C.

- all $\hat{f}(S)^2 = 0$ for $|S| > |T|$
- if $|T| \leq \log\frac{1}{\varepsilon}$ then ✓
- if $|T| \geq \log\frac{1}{\varepsilon}$ then:

$$\hat{f}(\phi)^2 = \left(1 - 2\Pr(f(x) \neq \chi_\phi(x))\right)^2 = \left(1 - \frac{2}{2^{|T|}}\right)^2 > 1 - \varepsilon$$

so $\sum_{S \neq \phi} \hat{f}(S)^2 \leq \varepsilon$ + f has 0-F.C. ∎

Now, let's approximate fctns with $d = \alpha(\varepsilon, n)$ F.C.:

Low Degree Algorithm

Given
- d degree
- τ accuracy
- δ confidence

Algorithm
- Take $m = O\left(\frac{n^d}{\tau} \ln \frac{n^d}{\delta}\right)$ samples
- $C_S \leftarrow$ estimate of $\hat{f}(S)$ (for each S s.t. $|S| \leq d$)
- output $h(x) = \sum_{|S| \leq d} C_S \chi_S(x)$

$\leq \binom{n}{d}$ of these

Can reuse same samples for each!

† use $\text{sign}(h(x))$ as hypothesis!

Why does this work?

Two stages:

1) show that if f has low F.C. then $E_x[(f(x) - h(x))^2]$ small (L_2 dist / 2^n)

2) show that $\Pr[f(x) \neq \text{sign}[h(x)]] \leq E_x[(f(x) - h(x))^2]$

Hamming dist

Thm if f has $d = \alpha(\varepsilon, n)$ – F.c. then h satisfies $E_x[(f(x)-h(x))^2] \le \varepsilon + \tau$ with prob $\ge 1-\delta$

Pf

Claim with prob $\ge 1-\delta$, $\forall\, S$ s.t. $|S| \le d$, $|C_S - \hat{f}(S)| \le \gamma$ for $\gamma \leftarrow \sqrt{\frac{\tau}{n^d}}$

Pf of claim

note, $\frac{1}{\gamma^2} = \frac{n^d}{\tau}$

Chernoff bnd $\Rightarrow O\left(\frac{n^d}{\tau} \ln \frac{n^d}{\delta}\right) = O\left(\frac{1}{\gamma^2} \ln \frac{n^d}{\delta}\right)$ samples

yields $\Pr\left[|C_S - \hat{f}(S)| > \gamma\right] < \frac{\delta}{n^d}$

union bnd $\Rightarrow \Pr[\exists\, S \text{ s.t. } |C_S - \hat{f}(S)| > \gamma] < \delta$ ∎

↑ only $\binom{n}{d} < n^d$ such S's of size $\le d$

Assume $\forall\, S$ s.t. $|S| \le d$, $|C_S - \hat{f}(S)| \le \gamma$

define $g(x) \equiv f(x) - h(x)$

Fourier transform is linear $\Rightarrow \forall S \quad \hat{g}(S) = \hat{f}(S) - \hat{h}(S)$

by defn, $\forall S$ s.t. $|S| > d$, $\hat{h}(S) = 0 \Rightarrow \hat{g}(S) = \hat{f}(S)$

$|S| \le d$, $\hat{h}(S) = C_S \Rightarrow \hat{g}(S) = \hat{f}(S) - C_S$

so $\hat{g}(S)^2 \le \gamma^2$

so $E[(f(x)-h(x))^2] = E[g(x)^2]$

$= \sum_S \hat{g}(S)^2$ Parseval

$= \sum_{|S|\le d} \hat{g}(S)^2 + \sum_{|S|>d} \hat{g}(S)^2$

$\le \gamma^2$, $\le \varepsilon$ by F.C.

$\le n^d \cdot \gamma^2$

$\le \tau + \varepsilon$ ∎

Thm $f : \{\pm 1\}^n \to \{\pm 1\}$

$h : \{\pm 1\}^n \to \mathbb{R}$

then $\Pr[f(x) \neq \text{sign}(h(x))] \le E[(f(x)-h(x))^2]$

Pf. $E[(f(x)-h(x))^2] = \frac{1}{2^n} \sum_x (f(x)-h(x))^2$ defn

$\Pr[f(x) \neq \text{sign}(h(x))] = \frac{1}{2^n} \sum_x 1_{\{f(x) \neq \text{sign}(h(x))\}}$

} show term by term

But if $f(x) = \text{sign}(h(x))$

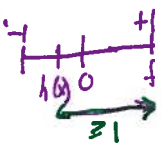
$(f(x)-h(x))^2 \ge 0$

$1_{f(x) \neq \text{sign}(h(x))} = 0$

if $f(x) \neq \text{sign}(h(x))$

$(f(x)-h(x))^2 \ge 1$

$1_{f(x) \neq \text{sign}(h(x))} = 1$

eg. 

So $\forall x$, $(f(x)-h(x))^2 \ge 1_{f(x) \neq \text{sign}(h(x))}$ ∎

Correctness of learning algorithm:

Thm. if $\mathcal{C}$ has Fourier concentration $d = \alpha(\varepsilon, n)$
then there is a $q = O(\frac{n^d}{\varepsilon} \log \frac{n^d}{\delta})$ sample
uniform distribution learning algorithm for $\mathcal{C}$
ie. algorithm gets q samples & with prob $\geq 1-\delta$
outputs h' s.t. $\Pr[f \neq h'] \leq 2\varepsilon$

Pf. run low degree alg with $\hat{\tau} = \varepsilon$
get h s.t. $E[(f-h)^2] \leq \varepsilon + \varepsilon = 2\varepsilon$
output $\operatorname{sign}(h)$ ∎

Applications

1) Bounded depth decision trees

$$f(x) = \sum_{\ell \in \text{leaves of } T} f_\ell(x) \cdot \text{val}(\ell)$$

$f_\ell(x)$: fctn which depends on $\leq$ depth many vars
val(ℓ): const

by linearity, $\hat{f}(S) = \sum \text{val}(\ell) \cdot \hat{f}_\ell(S)$ which is 0 if $|S| >$ depth

2) Constant depth ckts:

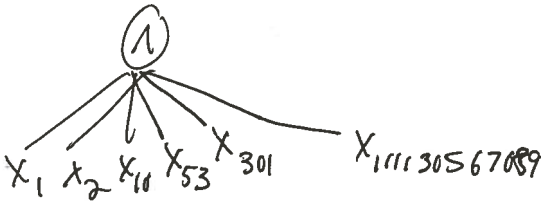
Def. "Boolean Ckt C" is a DAG

gates: $\wedge, \vee, \neg, 1, 0, X_1 \cdots X_n$

and, or, not, or ± 1, vars

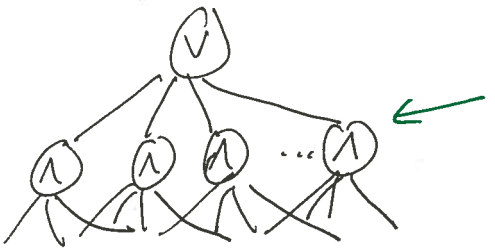
how many inputs?

const? poly? unbounded?



Can we compute parity of n bits in const depth?

yes! can compute any n-bit fctn in constdepth



← each "∧" picks an arbitrary sat setting & checks if input matches

Can we compute parity of n bits in const depth, poly size?

No! [Furst Saxe Sipser] } lemon

Switching lemma

Lemons ⇒ Lemonade

Thm [Hastad, Linial, Mansour, Nisan]

$\forall f$ computable via size s depth d ckts

$$\sum_{|S|>t} \hat{f}^2(S) \leq \alpha \quad \text{for} \quad t = O\left(14 \log \frac{2s}{\alpha}\right)^{d-1}$$

prove via random restrictions as in [Fss] parity result

Take Advanced Complexity!

take $s = \text{poly}(n)$, $d = \text{const}$, $\alpha = O(\varepsilon)$ $\Rightarrow t = O(\log^d(\frac{n}{\varepsilon}))$

Gives $n^{O(\log^d(\frac{n}{\varepsilon}))}$ sample query algorithm

(note: can improve to $n^{O(\log\log n)}$ [Jackson])

Application 2

Learning halfspaces (linear threshold fctns)

Def. $h(x) = \text{sign}(w \cdot x - \theta)$ is a "halfspace function"

$$\text{sign}(x) \begin{cases} +1 & \text{if } x \geq 0 \\ -1 & \text{o.w} \end{cases}$$

Thm Let h be a halfspace over $\{\pm 1\}^n$
then h has fourier concentration $\alpha(\varepsilon) = \frac{c}{\varepsilon^2}$

(ie. $\sum_{|S| \geq \frac{c}{\varepsilon^2}} \hat{h}(S)^2 \leq \varepsilon$)

(Will prove soon)

Corr low degree algorithm learns halfspaces under uniform distribution with $n^{O(1/\varepsilon^2)}$ uniform samples.

(Actually can learn in $O(n^5)$ but we'll get a "big win" from this approach soon...)

Key idea: Noise sensitivity

def.

(1) $0 < \varepsilon < 1/2$ "Noise operator"

$N_\varepsilon(x)$ = randomly flip each bit of X with prob ε

(2) "Noise Sensitivity"

$$NS_\varepsilon(f) = \Pr_{\substack{X \in \{\pm 1\}^n, \\ \text{noise}}} [f(x) \neq f(N_\varepsilon(x))]$$

Examples

1) $f(x) = X_1$ $\quad NS_\varepsilon(f) = \varepsilon$

2) $f(x) = X_1 X_2 \cdots X_k$ $\quad NS_\varepsilon(f) = \Pr[f(x) = F \text{ \& } f(N_\varepsilon(x)) = T]$
$+ \Pr[f(x) = T \text{ \& } f(N_\varepsilon(x)) = F]$

$= 2 \cdot \Pr[f(x) = T \text{ \& } f(N_\varepsilon(x)) = F]$ ← symmetry since $N_\varepsilon(x)$

$= 2 \cdot \frac{1}{2^k} \cdot (1 - (1-\varepsilon)^k)$

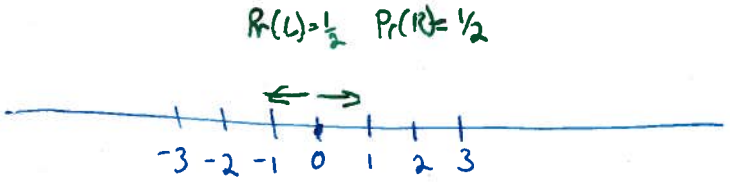
$= \begin{cases} \frac{1}{2^{k-1}}(\varepsilon k) & \text{if } \varepsilon \ll \frac{1}{k} \\ \frac{1 - e^{-k\varepsilon}}{2^{k-1}} & \text{if } \varepsilon \gg 1/k \end{cases}$ } small but fctn is rarely 1

3) $f(x) = Maj(X_1 \cdots X_n)$

$NS_\varepsilon(f) = O(\sqrt{\varepsilon})$

Sketch

- $Maj(x)$ ~ random walk on line starting at 0



$X_i = 1 \Rightarrow R \times 2$

$X_i = -1 \Rightarrow L \times 2$

$\sum X_i$ = endpt of walk (if start at 0)

e.g. $x = (1\ 1\ -1\ 1\ -1\ -1)$

ends at 0

equivalent process:

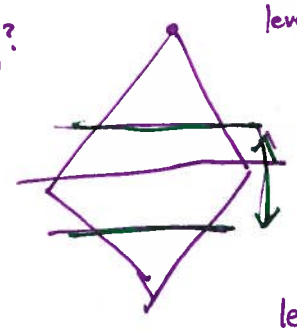
pick random pt on hypercube & output the level

Well known fact:

expected distance from startpt $(|X_1 + X_2 + \cdots + X_n|)$ after n steps is $\sqrt{n}$

& likely to be close to expectati

why?



- $N_\varepsilon(x)$ ~ rando m walk on εn bits (each has twice value later since chan +1 → -1, -1 → +1)

expected displacement $2\sqrt{\varepsilon n}$

Another view:
- take walk according to X
- then take walk according to $N_\varepsilon(x)$

- heuristic argument:
  pretend 1st walk leaves us at $\sqrt{n}$
  $\Pr[\text{2nd walk takes us across 0}] = \frac{1}{2}\Pr[\text{2nd displacement} > \sqrt{n}]$
  $= \frac{1}{2\sqrt{\varepsilon}} \cdot 2\cdot\sqrt{\varepsilon n}$
  $< 2\sqrt{\varepsilon}$ by Markov's $\neq$

4) any LTF (½space)

Thm [Peres] $NS_\varepsilon(LTF) < 8.8\sqrt{\varepsilon}$ ← best possible since Maj is LTF & has $NS_\varepsilon(MAJ) = \Theta(\sqrt{\varepsilon})$

5) parity fctns $\chi_S(x)$ for $|S|=k$ ← we've done $k=1$ (dictators)

$NS_\varepsilon(f) = \Pr[\text{odd \# bits in } S \text{ flipped by } N_\varepsilon]$
$= \binom{k}{1}(1-\varepsilon)^{k-1}\varepsilon + \binom{k}{3}\cdot(1-\varepsilon)^{k-3}\varepsilon^3 + \dots$

but $1^k = ((1-\varepsilon)+\varepsilon)^k = (1-\varepsilon)^k + \binom{k}{1}(1-\varepsilon)^{k-1}\varepsilon + \binom{k}{2}(1-\varepsilon)^{k-2}\varepsilon^2 + \dots$

$-\ (1-2\varepsilon)^k = ((1-\varepsilon)-\varepsilon)^k = (1-\varepsilon)^k - \binom{k}{1}(1-\varepsilon)^{k-1}\varepsilon + \binom{k}{2}(1-\varepsilon)^{k-2}\varepsilon^2 + \dots$

$\div 2$

$\frac{1-(1-2\varepsilon)^k}{2} = NS_\varepsilon(f)$

↖ high for large k (if S likely to get "hit")

6) any f:

Thm $f: \{\pm 1\}^n \to \{\pm 1\}$

$$NS_\varepsilon(f) = \frac{1}{2} - \frac{1}{2} \sum_S (1-2\varepsilon)^{|S|} \hat{f}(S)^2$$

note for parity fctns, this gives

$$NS_\varepsilon(\chi_S) = \frac{1}{2} - \frac{1}{2}\cdot(1-2\varepsilon)^{|S|} \quad \checkmark$$

Pf.

$$NS_\varepsilon(f) = \Pr_{\substack{x \\ y \in N_\varepsilon(x)}} [f(x) \neq f(y)]$$

$$= E_{\substack{x, \\ y \in N_\varepsilon(x)}} [1_{f(x)\neq f(y)}]$$

$$= E_{xy}\left[\frac{(f(x)-f(y))^2}{4}\right] = \frac{1}{4} E[\underbrace{f(x)^2 + f(y)^2}_{\text{always } 1} - 2 f(x) f(y)]$$

$$= \frac{1}{2} - \frac{1}{2} E[f(x) f(y)]$$

$$= \frac{1}{2} - \frac{1}{2} E\left[\sum_S \hat{f}(S) \chi_S(x) \sum_T \hat{f}(T) \chi_T(y)\right]$$

$$= \frac{1}{2} - \frac{1}{2} \sum_{S,T} \hat{f}(S)\hat{f}(T) E[\chi_S(x)\chi_T(y)]$$

$= 0$ if $S \neq T$ } show using standard techniques using pairn on $i \in S\Delta$

but, if $S = T$?

$$E_{xy}[\chi_S(x)\chi_T(y)] = 1 \cdot \Pr[\chi_S(x) = \chi_S(y)] + (-1)\Pr[\chi_S(x) \neq \chi_S(y)]$$

$$= 1 - 2\Pr[\chi_S(x) \neq \chi_S(y)]$$

$NS_\varepsilon(\chi_S) = \frac{1}{2} - \frac{1}{2}(1-2\varepsilon)^{|S|}$

$$= (1-2\varepsilon)^{|S|}$$

$$= \frac{1}{2} - \frac{1}{2} \sum_S (1-2\varepsilon)^{|S|} \hat{f}(S)^2$$

□

Noise Sensitivity vs. Fourier Concentration:

Thm $\forall f : \{\pm 1\}^n \to \{\pm 1\}$ $\quad 0 < \gamma < 1/2$

$$\sum_{|S| \geq \frac{1}{\gamma}} \hat{f}(S)^2 < 2.32\, NS_\gamma(f)$$

Pf

$$2\, NS_\gamma(f) = 1 - \sum_S (1-2\gamma)^{|S|} \hat{f}(S)^2 \qquad \text{previous thm}$$

$$= \sum_S \hat{f}(S)^2 - \sum_S (1-2\gamma)^{|S|} \hat{f}(S)^2 \qquad \text{Boolean Parseval's}$$

$$= \sum_S \left[1 - (1-2\gamma)^{|S|}\right] \hat{f}(S)^2$$

$$\geq \sum_{S \text{ s.t. } |S| \geq \frac{1}{\gamma}} \left[1 - (1-2\gamma)^{1/\gamma}\right] \hat{f}(S)^2 \qquad \text{drop lower terms, raise to lower power}$$

$$> \sum \left(1 - e^{-2}\right) \hat{f}(S)^2$$

so $$\sum_{|S| > \frac{1}{\gamma}} \hat{f}(S)^2 < \underbrace{\frac{2}{1-e^{-2}}}_{< 2.32}\, NS_\gamma(f)$$ ∎

Which gives:

Corr 1 $f: \{\pm 1\}^n \to \{\pm 1\}$. $\beta : [0, 1/2] \to [0, 1/2]$ is fctn s.t. $NS_\gamma(f) \leq \beta(\gamma)$

Then $\sum_{|S| \geq m} \hat{f}(S)^2 \leq \varepsilon$ for $m = \frac{1}{\beta^{-1}(\varepsilon/2.32)}$ } f has Fourier concentration m

Pf

$$\sum_{|S| \geq m} \hat{f}(S)^2 \leq \sum_{|S| \geq \frac{1}{\beta^{-1}(\varepsilon/2.32)}} \hat{f}(S)^2 \leq 2.32\, NS_{\frac{1}{\beta^{-1}(\varepsilon/2.32)}}(f) \leq 2.32\, \beta\left(\beta^{-1}\left(\frac{\varepsilon}{2.32}\right)\right) = \varepsilon$$

(thm; $\frac{1}{\beta^{-1}(\varepsilon/2.32)}$ set to "γ"; def of β)

Back to ½ spaces:

Corr 2 for ½ space $h: \{\pm 1\}^n \to \{\pm 1\}$

$$\sum_{|s| \geq O(\frac{1}{\varepsilon})} \hat{f}(s)^2 \leq \varepsilon$$

Pf

for ½ space $ns_\varepsilon(h) \leq 8.8\sqrt{\varepsilon}$ so use $\beta(\varepsilon) = 8.8\sqrt{\varepsilon}$

$\beta^{-1}(X) = \left(\frac{X}{8.8}\right)^2$

$\beta^{-1}\left(\frac{\varepsilon}{2.32}\right) \leq \left(\frac{\varepsilon}{2.32 \times 8.8}\right)^2$ ($2.32 \times 8.8 \approx 20.416$)

$\leq \left(\frac{\varepsilon}{20}\right)^2$

so for $m \geq \frac{1}{\left(\frac{\varepsilon}{20}\right)^2}$

□

⇒ Can learn any ½ space over uniform with $n^{O(1/\varepsilon^2)}$ random examples

but, can do a lot better!

still, technique extends ...

Learn any fctn of k ½ spaces:

$h_1 \ldots h_k$ are k halfspaces

$g: \{\pm 1\}^k \to \{\pm 1\}$ Boolean fctn

$f: \{\pm 1\}^n \to \{\pm 1\}$ st $f(x) = g(h_1(x), \ldots, h_k(x))$

Thm $ns_\varepsilon(f) = 8.8k\sqrt{\varepsilon}$

Pf

$$ns_\varepsilon(f) = \Pr[f(x) \neq f(N_\varepsilon(x))]$$

$$= \Pr[g(h_1(x) \dots h_k(x)) \neq g(h_1(N_\varepsilon(x)), \dots h_k(N_\varepsilon(x)))]$$

$$\leq \Pr[h_1(x) \neq h_1(N_\varepsilon(x)) \text{ or } h_2(x) \neq h_2(N_\varepsilon(x)) \text{ or} \dots]$$

$$\leq k \cdot 8.8 \cdot \sqrt{\varepsilon} \qquad \text{union bnd}$$

■

Take $\beta(\varepsilon) = 8.8\, k \sqrt{\varepsilon}$

so $\beta^{-1}(x) = \left(\frac{x}{8.8\,k}\right)^2$

+ corr 1 $\Rightarrow$

Thm $\sum_{|S| \geq \Omega\left(\frac{k^2}{\varepsilon^2}\right)} \hat{f}(S)^2 \leq \varepsilon$

Thm low degree alg learns any Boolean fctn of k ½ spaces with $n^{O(k^2/\varepsilon^2)}$ samples

e.g. $\wedge$ of 2 ½ spaces

parity of k vars