Lecture 19

Fourier-based learning algorithms

- Fourier Concentration via Noise sensitivity
- Learning heavy Fourier coeffs (with queries)

Recall Fourier Transform:

$$\chi_S(x) = \prod_{i \in S} x_i$$

$$\langle f, g \rangle = \frac{1}{2^n} \sum_x f(x)\, g(x)$$

lemma

$$\hat{f}(S) = \langle f, \chi_S \rangle \quad = 1 - 2 \cdot \Pr[f(x) \neq \chi_S(x)]$$

$$\forall f, \quad f(x) = \sum \hat{f}(S) \chi_S(x)$$

Plancherel $\langle f, g \rangle = \sum_S \hat{f}(S)\, \hat{g}(S)$

Learning via Fourier Representation

will look at learning algorithms that are based on estimating Fourier representation of fctn f (similar to polynomial interpolation)

Approximating one Fourier coefficient:

<u>lemma</u> for any $S \subseteq [n]$, can approx $\hat{f}(S)$ to within additive γ

(ie. $|\text{output} - \hat{f}(S)| \leq \gamma$)

with prob $\geq 1-\delta$ in $O(\frac{1}{\gamma^2} \log \frac{1}{\delta})$ samples.

no queries needed!

(Proved last time)

The low degree algorithm

definition of fctns for which low degree Fourier coeffs pretty much suffice to describe fctn:

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

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

for Boolean f, this implies

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

<u>Thm</u> if $\mathcal{C}$ has Fourier concentration $d = \alpha(\varepsilon, n)$

then there is a $q = O\left(\frac{n^d}{\varepsilon} \log \frac{n^d}{\delta}\right)$ sample uniform distribution learning algorithm for $\mathcal{C}$

i.e. algorithm gets q samples & with prob $\geq 1-\delta$ outputs h' s.t. $\Pr[f \neq h'] \leq 2 \cdot \varepsilon$

Applications

1) Bounded depth decision trees

2) Const depth ckts

3) halfspaces (linear threshold fctns)

Key idea:

Noise Sensitivity

← use to bound Fourier concentration

def. "Noise operator" $0 < \varepsilon < 1/2$

$$N_\varepsilon(x) = \text{randomly flip each bit of } x \text{ with prob } \varepsilon$$

def "Noise sensitivity"

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

Examples

1. $f(x) = x_1$ $\qquad ns_\varepsilon(f) = \varepsilon$

2. $f(x) = x_1 x_2 \dots x_k$ $\qquad ns_\varepsilon(f) = \Pr[f(x) = F \;\&\; f(N_\varepsilon(x)) = T]$

$$+ \Pr[f(x) = T \;\&\; f(N_\varepsilon(x)) = F]$$

by symmetry → $= 2 \cdot \Pr[f(x) = T \;\&\; f(N_\varepsilon(x)) = F]$

if $\varepsilon \ll \frac{1}{k}$, $\approx \frac{1}{2^{k-1}}(\varepsilon k)$ → $= \frac{2}{2^k}\left(1 - (1-\varepsilon)^k\right)$

if $\varepsilon \gg 1/k$, $\approx (1 - e^{-k\varepsilon})/2^{k-1}$

(End review)

6. Any f

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

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

for parity fcfns: $\frac{1}{2} - \frac{1}{2}(1-2\varepsilon)^{|S|}$

Pf. homework?

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

$$2 \cdot ns_\gamma(f) = 1 - \sum_S (1-2\gamma)^{|S|} \hat{f}(S)^2$$ previous thm

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

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

$$\geq \sum_{\substack{S \text{ s.t.} \\ |S| \geq 1/\gamma}} \left[1 - (1-2\gamma)^{1/\gamma}\right] \hat{f}(S)^2$$

$$> \sum_{|S| \geq 1/\gamma} (1 - e^{-2}) \hat{f}(S)^2$$

So $$\sum_{|S| \geq 1/\gamma} \hat{f}(S)^2 < \underbrace{\left(\frac{2}{1-e^{-2}}\right)}_{2.32} \cdot ns_\gamma(f)$$

∎

Corr for halfspace $h: \{\pm 1\}^n \to \{\pm 1\}$

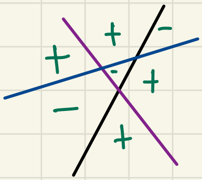
$$\sum_{|S| \geq O(\frac{1}{\varepsilon^2})} \hat{f}(S) \leq \varepsilon$$

(pf omitted - some calculations + bound on NS)

$\Longrightarrow$ Can learn any halfspace from $n^{O(1/\varepsilon^2)}$ random examples

(actually can do a lot better)

Corr any function of k halfspaces can be learned with $n^{O(k^2/\varepsilon^2)}$ samples



e.g. parity of k vars, $\wedge$ of $\frac{1}{2}$spaces

Pf idea noise sensitivity $\leq 8.8\, k\sqrt{\varepsilon}$ by union bound.

Learning Heavy Fourier Coeffs

[Goldreich Levin]
[Kushilevitz Mansour]

$$x \longrightarrow \boxed{\;f\;} \longrightarrow f(x)$$

not just low degree S
$\downarrow$
all close linear fctns

Given f, θ

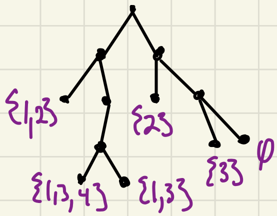
- Output all coeffs S s.t. $|\hat{f}(S)| \geq \theta$
- Only output S s.t. $|\hat{f}(S)| \geq \frac{\theta}{2}$ $\Leftarrow$ no junk

Probably can't do it with only random examples

What if can query f at any input?

Main Idea: "exhaustive search with good pruning"

leaves
$\sim$ Fourier coeffs



Var
x_1
x_2
x_3
x_4

left $\Rightarrow x_i \in S$
right $\Rightarrow x_i \notin S$

ONLY OUTPUT THOSE THAT REACH BOTTOM LEVEL

recursive algorithm:

- each node $\sim$ setting of $x_1 \cdots x_i$
- estimate "total energy" of subtrees $x_1 \cdots x_i \circ (x_{i+1} = +1)$ & $x_1 \cdots x_i \circ (x_{i+1} = -1)$
- only go down paths with high enough energy

How to prune?

Define quantity:

k		S_1
0		ϕ
1		$\{x_1\}$
2		$\{x_1\}$
3		$\{x_1, x_3\}$
4		$\{x_1, x_3, x_4\}$

Fix $0 \le k \le n$ current "level" of search

$S_1 \subseteq [k]$ current "node" of search

2^k such fctns (for each S_1)

$$f_{k,S_1} : \{\pm 1\}^{n-k} \to \mathbb{R}$$

all Fourier coeffs which agree on first k elements

$$\text{s.t. } f_{k,S_1}(x) = \underbrace{\sum_{T_2 \subseteq \{k+1..n\}}}_{\text{all extensions of } S_1 \text{ to indices in } \{k+1...n\}} \hat{f}(S_1 \cup T_2)\, \chi_{T_2}(x)$$

could be $S_1 \cup T_2$ but no need since $\chi_{S_1 \cup T_2} = \underbrace{\chi_{S_1}}_{\text{same for all}} \cdot \chi_{T_2}$

notation:
index 1 → prefix
2 → suffix

where are S_2 & T_1 in analysis?

Sanity Checks:

1) $k=0$

$$f_{0,\phi}(x) = \sum_{T_2 \subseteq [n]} \hat{f}(T_2)\,\chi_{T_2}(x) = f(x)$$

since $k=0$ since $S_1 = \emptyset$

2) $k=n$ $\quad f_{n,S_1}(x) = \hat{f}(S_1)$

sum over $T_2 = \emptyset$ since $T_2 = \emptyset$

Plan Only go down paths with $\mathbb{E}_{k,s}[f^2(x)] \geq \theta^2$

1. can we compute it?
2. does it bring us to right leaves?
   - do we get to all heavy leaves?
   - do we get junk? (light leaves)
3. how many paths do we take?

   lots of dead ends?
   is runtime good?

Not too many paths! (answer to 3)

at any stage in algorithm

Lemma "not too many"

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

(1) $\leq \frac{1}{\theta^2}$ S's satisfy $|\hat{f}(S)| \geq \theta$

(2) $\forall\ 0 \leq k \leq n$, $\leq \frac{1}{\theta^2}$ fctns f_{k,S_1}

have $E_x[f_{k,S_1}^2] \geq \theta^2$

Pf

(1) Parseval's $1 = E_x[f^2(x)] = \sum_S \hat{f}(S)^2$

so if $> \frac{1}{\theta^2}$ S's satisfy $|\hat{f}(S)| \geq \theta$

then $\sum_S \hat{f}(S)^2 > \frac{1}{\theta^2} \cdot \theta^2 > 1$

$\to\leftarrow$

(2) For given k:

<u>Claim</u>: $\forall\ k,\ S_1 \subseteq k$

$$\mathbb{E}_x\left[f_{k,S_1}(x)^2\right] = \sum_{T_2 \subseteq \{k+1,\dots,n\}} \hat{f}(S_1 \cup T_2)^2$$

<u>Pf of claim</u>:

$$\mathbb{E}_x\left[f_{k,S_1}(x)^2\right] = \mathbb{E}_x\left[\left(\sum_{T_2} \hat{f}(S_1 \cup T_2)\,\chi_{T_2}(x)\right)^2\right] \quad \text{def.}$$

$$= \mathbb{E}_x\left[\sum_{\substack{T_2, T_2' \\ \subseteq \{k+1 \dots n\}}} \hat{f}(S_1 \cup T_2)\cdot\hat{f}(S_1 \cup T_2')\,\chi_{T_2}(x)\,\chi_{T_2'}(x)\right]$$

$$= \sum_{T_2, T_2'} \hat{f}(S_1 \cup T_2)\,\hat{f}(S_1 \cup T_2')\,\mathbb{E}\underbrace{\left[\chi_{T_2}(x)\cdot\chi_{T_2'}(x)\right]}_{\substack{=1 \text{ if } T_2 = T_2' \\ =0 \text{ o.w.}}}$$

$$= \sum_{T_2} \hat{f}(S_1 \cup T_2)^2 \qquad \blacksquare$$

Using Claim:

Boolean Parseval's

expanding

$$1 = \sum_S \hat{f}(S)^2 = \sum_{S_1 \subseteq K} \sum_{T_2 \subseteq \{K+1 .. n\}} \hat{f}(S_1 \cup T_2)^2$$

$$= \sum_{S_1} E_x[f^2_{K,S_1}(x)] \qquad \text{claim}$$

so $\leq \frac{1}{\theta^2}$ S_1's can have $E_x[f^2_{K,S_1}(x)] > \theta^2$

$\blacksquare$