Today:

Learning large Fourier coefficients with queries

\[ \text{[Goldreich Levin]} \]
\[ \text{[Kushilevitz - Mansour]} \]

\[ x \quad \rightarrow \quad f \quad \rightarrow \quad f(x) \]

Given \( f, \theta \):

1) Output all \( S \) s.t. \( |\hat{f}(s)| \geq \theta \) \( \iff \) all close parity fits
2) All outputted \( S \) satisfy \( |\hat{f}(s)| \geq \frac{\theta}{2} \) \( \iff \) no junk

(Probably) can't do it with random examples
do queries help?

Last time: (warmup) only one big \( \hat{f}(s) \)

Algorithm: Find \( S \) bit-by-bit
General Case Main Idea:

exhaustive search with good pruning.

Idea: find way to use sampling to estimate "total energy" in subtree only go down high energy paths

How do we prune?

Define useful quantities:

Fix $0 \leq k \leq N$ current level of search

$s_i \in [K]$ current "node" of search

$s_i \sim r$ turns

$f_{k,s_i}: \mathbb{R}^{n-k} \to \mathbb{R}$

$s_{+} f_{k,s_i}(x) = \sum_{T_2 \in \mathcal{G}_{k+1\ldots n}} \hat{f}(s_i \cup T_2) X_{T_2}(x)$

will call with $y = x_{k+1\ldots n}$
Index 1 ~ prefix
Index 2 ~ suffix

Four coeffis \( \hat{f}(s) \)
\( s \) agrees with \( s_i \) on \( k \)

Sanity checks:
1) \( k=0 \) \( f_{0|0} = \sum_{T_2 \leq [n]} \hat{f}(T_2) \chi_{T_2}(n) = f(n) \)
2) \( k = n \) \( f_{n|s_i}(x) = \hat{f}(s_i) \)

Partition four coeff into \( 2^k \) subsets

Plan: Go down paths with \( E[f_{k|s}(x)] \geq \Theta^2 \)

1. Can compute?
2. Does it bring us to right leaves?
   - do we get all heavy leaves?
   - do we get too much junk?
      (light leaves)
3. How many paths do we take?
   lots of dead ends?
   runtime good?
Not too many paths (answer to 3)

\text{Lemma} "not too many" not just at end but also at any level of tree

1. Boolean

1. \( \frac{1}{\Theta^2} \) S's satisfy \( |\hat{f}(s)| \geq \Theta \)

2. A 0 \leq k \leq n, \leq \frac{1}{\Theta^2} \) fewer \( f_{k,s} \) have

\[ \sum_{x} |f_{k,s}(x)| \geq \Theta^2 \]

\text{BAD}

\text{Lemma says this doesn't happen}

\text{exponential partial paths}

\text{small # of S's}

\text{Pf, (1) Boolean Parseval's} \quad 1 = \mathbb{E}_{x}[f^2(x)] = \sum_{s} |\hat{f}(s)|^2

\text{so if } \frac{1}{\Theta^2} \text{ S's then } |\hat{f}(s)| \geq \Theta

\Rightarrow \sum_{s} |\hat{f}(s)|^2 > \frac{1}{\Theta^2} \quad \Theta^2 \rightarrow \leq
(2) For given $k^*$:

**Claim:** \( \forall k^* \in [K], s_i \in [K] \)

\[
E_x \left[ f_{k^* s_i}(x)^2 \right] = \sum_{T_2 \in \mathcal{K}_{k^*}, \ldots, n^2} \hat{f}(s_i \cup T_2)^2
\]

**Proof:**

\[
E_x \left[ f_{k^* s_i}(x)^2 \right] = E_x \left[ \left( \sum_{T_2} \hat{f}(s_i \cup T_2) \chi_{T_2}(x) \right)^2 \right]
\]

\[
= \sum_{T_2, T_2'} \hat{f}(s_i \cup T_2) \hat{f}(s_i \cup T_2') E_x \left[ \chi_{T_2}(x) \chi_{T_2'}(x) \right]
\]

\[
= \sum_{T_2} \hat{f}(s_i \cup T_2)^2
\]

Using **Claim**:

\[
1 = \sum_{s} \hat{f}(s)^2 = \sum_{s_i \in [K]} \sum_{T_2 \in \mathcal{K}_{k^*}, \ldots, n^2} \hat{f}(s_i \cup T_2)^2
\]

\[
= \sum_{s_i} E_x \left[ f_{k^* s_i}(x)^2 \right] \leq \text{claim}
\]

as before \( \leq \frac{1}{6} \) \( S_i \)'s can have \( \sum_{s} f_{k^* s_i}(x)^2 > \theta^2 \).
\[ \Rightarrow \leq \frac{1}{\Theta^2} \quad \text{"live" S's at each level of tree} \]

\[ \Rightarrow \leq \frac{n}{\Theta^2} \quad \text{paths are traversed in free} \]

\[ \leq 2^n \quad \text{Answer to 3} \]

Does it bring us to good leaves? (Answer to 2)

Fact "not missing out" \[ \Rightarrow \text{find all big Four coeffs} \]

for any \( S_1 \) if \( \exists T_2 \) s.t.

\[ \text{heavy F.C.'s} \{ | f(s_1 \cup T_2) | > \Theta \} \]

\[ \text{then } \forall k \quad E_X \left[ f_{k, S_1}^2(x) \right] = \sum_{T_2} \frac{f(s_1 \cup T_2)^2}{T_2} \geq \Theta^2 \]

\[ \Rightarrow E_X \left[ f_{k, S_1}^2(x) \right] \text{ is a good measure to use to find heavy F.C.'s} \]
Great we find good leaves.

But do we output bad leaves?

1) note we don't reach too many leaves in total
   → can't be too many "candidates" bad leaves

2) Can test a leaf to see if good

Given candidate $S$:
   is $f(s)$ big?

Previous lectures $S$ can estimate $f(s)$ for any specific $S$ quickly

⇒ can make sure not to output junk

---

Can we estimate $f_{KS}(x)$ (answer to (1))

Bad idea: estimate each $f(s, VT_2) \forall T_2$ (too many $T_2$'s)
Belkr idea:

"$f_{k,s_i}(x)$ estimation" lemma:

$f : \mathbb{Z}^n \rightarrow \mathbb{Z}^n$

$0 \leq k \leq n$

$s_i \subseteq [k]$

for $x \in \mathbb{Z}^n$

\[
\sum_{T_2 \subseteq \mathbb{Z}^{n-k}} \mathbb{E}_{y \in \mathbb{Z}^n} \left[ f_{k,s_i}(x) = E_{y \in \mathbb{Z}^n} \left[ f(y) \chi_{s_i}(y) \right] \right]
\]

Proof:

$f(y;x) = \sum_{T} \hat{f}(T) \chi_T(y;x)$

$T = T_1 \cup T_2 \quad T_1 \subseteq [k] \quad T_2 \subseteq \mathbb{Z}^{k+1,\ldots,n}$

$\chi_T(y;x) = \chi_{T_1}(y) \cdot \chi_{T_2}(x)$

$E_{y \in \mathbb{Z}^n} \left[ f(y) \chi_{s_i}(y) \right]$

$= E_{y \in \mathbb{Z}^n} \left[ \left( \sum_T \hat{f}(T) \chi_T(y;x) \right) \chi_{s_i}(y) \right]$

$= \mathbb{E}_{y \in \mathbb{Z}^n} \left[ \sum_{T_1} \left( \sum_{T_2} \hat{f}(T_1 \cup T_2) \chi_{T_1}(y) \cdot \chi_{T_2}(x) \right) \chi_{s_i}(y) \right]$
\[
\sum_{T_1} \sum_{T_2} \hat{f}(T_1, T_2) \cdot \chi_{T_1}(x) E_y[\chi_{T_1}(y) \cdot \chi_{S_1}(y)]
\]

= 0 \quad \text{if } T_1 \neq S_1;
= 1 \quad \text{if } T_1 = S_1;

= \sum_{T_2} \hat{f}(S_1, T_2) \chi_{T_2}(x)

= f_{S_1S_1}(x)

Can estimate \( E_y[\hat{f}(y) \chi_{S_1}(y)] \)

pick random queries to compute yourself

for several \( y \)'s, take average + Chernoff

\( O\left(\frac{1}{\epsilon^2 \log \frac{1}{\delta}}\right) \) queries

for add error \( \epsilon \)

prob of bad approx \( \leq \delta \)

\rightarrow Answer to question \( \ell \) is "yes"
Algorithm:

If $k = n$:
    test $s_i$ and output if good

Else
    if $\mathbb{E}_x \left[ f^2(x) \right] \geq \Theta^2/2$
        recurse on $(k+1, S_i \cup \mathcal{K}+k_i)$
        (else kill this subtree)

    if $\mathbb{E}_x \left[ f^2_{k, s_i}(x) \right] \geq \Theta^2/2$
        recurse on $(k+1, S_i)$
        (else kill this subtree)

Problem only get estimate of
$\mathbb{E}_x [f^2_{k, s_i}(x)]$

using $\Theta^2/2$ ensures we get all heavy coeffs
using same analysis as before, still won't have too many paths
+ still test all candidate $S$ before output $\hat{S}$
  no junk is output

Thm: $\exists \Theta \Rightarrow \exists \Theta$

$\forall \Theta > 0 \quad \text{KM-algorithm outputs set}$

$S = S_1, \ldots, S_{\Theta} \quad \text{st.} \quad l = O\left(\frac{1}{\Theta^2}\right)$

st. with prob $\geq 1 - \delta$

$\forall S_i \in S, \quad |\hat{f}(S)| \geq \Theta \sqrt{2}$

$\forall S_i \in S, \quad |\hat{f}(S) \leq \Theta$

What are values of $\hat{f}(S)$'s?

we can easily approximate
  (previous lectures)
Applications

- Decision trees of size $\leq t$

  Previous bound in terms of depth

- All functions of small $L_1$ norm

  $$ L_1(f) = \sum_s |f(s)| $$

  by setting $\Theta \leq \frac{2}{L_1(f)}$

  in time $\text{poly}(n, L_1(f), \frac{1}{\epsilon})$

  if don't know $L_1(f)$:

  guess 1, 2, 4, 8, ...

  run algorithm, get hypothesis

  test hypothesis & continue if not good