Notation

- Recall $\langle f, g \rangle$ is the expected value of $f(x)g(x)$ over uniformly random $x \in \{0, 1\}^n$
- The dot-product of $x, y \in \{0, 1\}^n$ is, represented by $x \cdot y$, equal to $\bigoplus_{i \in [n]} x_i y_i$
- A function $H: \{0, 1\}^n \rightarrow \mathbb{R}$ can be interpreted as a $2^n$ long string of $\mathbb{R}$ entries. On querying $H$ at $r$, we obtain the $r$-th entry of the string.
- Two functions $f$ and $g$ are close, if the strings corresponding to the function $f$ and $g$ differ only at a small number of positions (we will make this more quantitative later in the notes)
Problem Statement: Given an oracle $H : \{0, 1\}^n \rightarrow \{+1, -1\}$ that is close to some $\chi_A$, we are interested in querying $H$ multiple times and explicitly finding $\chi_S$.

Perspective (1): Recall, Hadamard code: The encoding of $S \subseteq [k]$ is the string corresponding to the function $\chi_S$. This linear code has block-length $n = 2^k$ and distance $d = 2^{k-1}$. So, $H$ is an erroneous codeword and we are interested in finding the nearest codeword, i.e. the decoding problem for Hadamard Code.

Perspective (2): Given a function $H$, we are interested in learning its heavy Fourier Coefficients. Restricting to these Fourier coefficients, we can compute a function $\tilde{H}$ that approximates $H$. That is, we approximately learn the function $H$ by querying it.
Assumption: $H$ completely agrees with some $\chi_S$

Algorithm: We query $H$ at $e_i$

If $H(e_i) = +1$, then we know that $i \not\in S$; and, if $H(e_i) = -1$, then we know that $i \in S$

By querying $H$ at all $e_i$, $i \in [n]$, we can always recover the set $S$
First Non-trivial Decoding Result

- Assumption: $H$ agrees with some $\chi_S$ with probability $3/4 + \varepsilon$, i.e. $H$ agrees with $\chi_S$ at some $(3/4 + \varepsilon)2^n$ inputs.

- Algorithm: We compute $a_i = H(r) \cdot H(r + e_i)$ for $r \leftarrow \{0, 1\}^n$.

- Note that: if “$H(r)$ and $H(r + e_i)$ both agree with $\chi_S(r)$ and $\chi_S(r + e_i)$” or “$H(r)$ and $H(r + e_i)$ both disagree with $\chi_S(r)$ and $\chi_S(r + e_i)$” then $a_i = \chi_S(e_i)$; otherwise, $a_i = -\chi_S(e_i)$.

- So, we have the following:

$$
\Pr[a_i \neq \chi_S(e_i)] = \Pr[H(r) \neq \chi_S(e_i) \land H(r + e_i) = \chi_S(r + e_i)] + \\
\Pr[H(r) = \chi_S(e_i) \land H(r + e_i) \neq \chi_S(r + e_i)] \\
\leq \Pr[H(r) \neq \chi_S(e_i)] + \Pr[H(r + e_i) \neq \chi_S(r + e_i)] \\
\leq 2(1/4 - \varepsilon) = 1/2 - 2\varepsilon
$$

- By sampling $k$ independent $r$s, we obtain $a_i$s that agree with $\chi_S(e_i)$ with probability $1/2 + 2\varepsilon$. 

Lecture 27: Goldreich-Levin Theorem
By taking the majority of the $a_i$s we recover $\chi_S(e_i)$ correctly, except with probability $\exp(-\Theta(k/\varepsilon^2))$ (Chernoff Bound). So, we recover $\chi_S(e_i)$ correctly with probability $1 - \Theta(1/n^2)$ by choosing $k = \Theta(\varepsilon^2 \log n)$.

We recover all $\chi_S(e_i)$, for all $i \in [n]$, with probability $1 - n \cdot \Theta(1/n^2) = 1 - 1/n$, if we choose $k = \Theta(\varepsilon^2 \log n)$ (by Union Bound).

Conditioned on recovering all $\chi_S(e_i)$, we recover $S$ (using the idea of reconstruction of $S$ when $H$ agrees with $\chi_S$ always).
There exists $S$ such that:

- $H$ agrees with $\chi_S$ with probability 1: We can recover $S$ with probability 1 by querying $H$ exactly $2n$ times.

- $H$ agrees with $\chi_S$ with probability $3/4 + \varepsilon$: We can recover $S$ with $1 - 1/n$ probability by querying $H$ exactly $\Theta(\frac{1}{\varepsilon^2} n \log n)$ times.
What if there is only 3/4 Agreement?

- Consider two distinct non-empty subsets $S$ and $S'$ and let $H(x) = \max\{\chi_S(x), \chi_{S'}(x)\}$
- Note that $H(x)$ agrees with each of $\chi_S(x)$ and $\chi_{S'}(x)$ exactly at 3/4 positions
- So, given $H$ if we decode it to $S$, then considering the witness “$H$ agrees with $\chi_{S'}$ with probability 3/4” we always fail to recover $S'$!
- Thus, “Unique Decoding” is impossible if $H$ agrees with (some) $\chi_S$ with probability in the range $(1/2, 3/4]$
- We do the next best thing: “List Decoding”
- Given $\varepsilon > 0$, the decoding procedure (probabilitically) outputs a list of subsets $L \subseteq 2^{[n]}$ such that if $H$ agrees with $\chi(S)$ with probability $1/2 + \varepsilon$ then $S \in L$ with constant probability (say, 1/2)
Size of $L$

**Lemma**

Given $H$ and $\varepsilon > 0$, let

$$L_\varepsilon = \{S : H \text{ agrees with } \chi_S \text{ with probability } 1/2 + \varepsilon\}$$

Then, $|L_\varepsilon| \leq 1/4\varepsilon^2$.

- Note that if $H$ and $\chi_S$ agree with probability at least $1/2 + \varepsilon$ then $\langle H, \chi_S \rangle = \hat{H}(S) \geq 2\varepsilon$
- By Parseval’s, we have $\sum_S \hat{H}(S)^2 = \|H\|_2^2 = 1$
- Therefore, we have $|L_\varepsilon| \leq 1/4\varepsilon^2$
Goal: Given $\varepsilon > 0$, (probabilistically) output a list $L$ such that for all $S \in L_\varepsilon$, we have $S \in L$ with probability at least $1/2$.
We will set ourselves an alternate goal: If $H$ agrees with $\chi_S$ with probability $1/2 + \varepsilon$ we will construct a new oracle $\tilde{H}$ that agrees with $\chi_S$ with probability $7/8$ (i.e. $3/4 + 1/8$)

Given access to $\tilde{H}$ we can recover $S$ (we have already seen how to recover $S$ if the agreement probability is $3/4 + \varepsilon$)
A Hypothetical Setting

- Suppose $\tilde{H}$ is queried at $r$. We compute the answer as follows.
- Let $\{r_1, \ldots, r_k\}$ be $k$ uniformly random string drawn from $\{0, 1\}^n$
- Suppose (hypothetically) we are given $\{b_1, \ldots, b_k\}$ such that
  $b_i = \chi_S(r + r_i)$, for all $i \in [k]$

- Now, $\chi_S(r_i) \cdot b_i$ always agrees with $\chi_S(r)$, for $i \in [k]$
- Therefore, $H(r_i) \cdot b_i$ agrees with $\chi_S(r)$ with probability $1/2 + \varepsilon$, for $i \in [k]$
- The majority of $\{H(r_1) \cdot b_1, \ldots, H(r_k) \cdot b_k\}$ agrees with $\chi_S(r)$ with probability $31/32$, for $k = \Theta(1/\varepsilon^2)$
- We output this majority ans as the answer $\tilde{H}(r)$
Analysis of Hypothetical Setting

- Over random \( r, r_1, \ldots, r_k \), (and conditioned on guessing \( b_1, \ldots, b_k \) correctly), we have:

\[
\Pr_{r,r_1,\ldots,r_k} \left[ \text{ans} = \tilde{H}(r) \right] \geq 31/32
\]

- Using an averaging argument:

\[
\Pr_{r_1,\ldots,r_k} \left[ \Pr_r \left[ \text{ans} = \tilde{H}(r) \right] \geq 7/8 \right] \geq 3/4
\]

- Intuition:
  - With probability 1/4 over the choices of \( r_1, \ldots, r_k \), we implement a \textit{bad} oracle \( \tilde{H} \).
  - With probability 3/4 over the choices of \( r_1, \ldots, r_k \), we implement a \textit{good} oracle \( \tilde{H} \) that agrees with \( \chi_S \) with probability 7/8 (given the correct guesses \( b_1, \ldots, b_k \)). In the good oracle case, we recover \( S \), except with 1/n probability.
  - We recover \( S \) with probability \( 3/4 - 1/n \geq 1/2 \).
Suppose we enumerate all possible bits $b_1, \ldots, b_k$ (this is exponential in $k$ and, hence, is not efficient)

When each $b_i$ agrees with $\chi_S(r_i + r)$ then we can recover $S$

Note that for different $S, S' \in L_\varepsilon$, the guesses are correct for different values of $\{b_i : i \in [k]\}$. If $\{b_i : i \in [k]\}$ is consistent with $\chi_S(r_i + r)$ then we recover $S$. If $\{b_i : i \in [k]\}$ is consistent with $\chi_{S'}(r_i + r)$ then we recover $S'$.

Think: Can we generate $r_i$s and $b_i$s with less independence?