Analysis of Boolean Functions **(CMU 18-859S, Spring 2007)**

Lecture 3: The BLR and Hstad Tests; local testing and decoding

Jan. 23, 2005

Lecturer: Ryan O'Donnell *Scribe: Samid Hoda*

1 Proof of the BLR Test

As promised we will finish the proof of the BLR test that we started in the last lecture. The BLR test (using the notation from the Great Notational Switch) has black-box access to a function $f : \{-1,1\}^n \to \{-1,1\}$ and executes the following steps:

- Picks $\boldsymbol{x}$ and $\boldsymbol{y}$ independently and uniformly at random from $\{-1,1\}^n$.
- Sets $\boldsymbol{z} = \boldsymbol{x} \circ \boldsymbol{y}$.
- Queries f on $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{z}$.
- "Accepts" iff $f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z}) = 1$.

The "vector product" $\boldsymbol{x} \circ \boldsymbol{y}$ used above is simply the coordinatewise vector product, i.e., $\boldsymbol{x} \circ \boldsymbol{y} = (x_1y_1, x_2y_2, \ldots, x_ny_n)$. Our goal is to establish the following two properties of the BLR test:

(i) If $f = \chi_T$ for some $T \subseteq [n]$, i.e., f is a parity function, then $\mathbf{Pr}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\text{BLR accepts}] = 1$.

(ii) If f is ϵ-far from being a parity function then $\mathbf{Pr}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\text{BLR accepts}] \le 1-\epsilon$.

The fact that (i) holds is clear. The main step in proving that the BLR test satisfies (ii) is the following lemma.

Lemma 1.1 *Given a function* $f : \{-1,1\}^n \to \{-1,1\}$

$$\mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\text{BLR}(f)\text{ accepts}] = \frac{1}{2} + \frac{1}{2}\sum_{S\subseteq[n]} \hat{f}(S)^3.$$

Proof: Since $f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z}) \in \{-1,1\}$ we can use the following indicator random variable for the event that the BLR test accepts

$$\mathbf{1}_{BLR} = \tfrac{1}{2} + \tfrac{1}{2} f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z}).$$

Therefore by linearity of expectation

$$\mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\text{BLR}(f)\text{ accepts}] = \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\mathbf{1}_{BLR}] = \tfrac{1}{2} + \tfrac{1}{2}\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})].$$

We now focus on $\mathbf{E}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})]$. Expressing $f(\boldsymbol{x})$, $f(\boldsymbol{y})$ and $f(\boldsymbol{z})$ in terms of their Fourier expansions we obtain the following:

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})] =$$

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}\left[\left(\sum_{S\subseteq[n]}\hat{f}(S)\chi_S(\boldsymbol{x})\right)\left(\sum_{T\subseteq[n]}\hat{f}(T)\chi_T(\boldsymbol{y})\right)\left(\sum_{U\subseteq[n]}\hat{f}(U)\chi_U(\boldsymbol{z})\right)\right].$$

Distributing out the product of sums (into a sum of products) yields

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})] = \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}\left[\sum_{S,T,U\subseteq[n]}\hat{f}(S)\hat{f}(T)\hat{f}(U)\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})\right].$$

By linearity of expectation this is equivalent to

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})] = \sum_{S,T,U\subseteq[n]}\left(\hat{f}(S)\hat{f}(T)\hat{f}(U)\cdot\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})]\right). \quad (1)$$

Recall that in the last lecture we established the following convention for the empty product: $\prod_{i\in\emptyset}x_i = 1$. In particular, $\chi_\emptyset$ is the function that is 1 on every input string. We will now focus our attention on simplifying the term $\mathbf{E}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})]$ in the equation above.

$$\begin{aligned}
\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})] &= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}\left[\prod_{i\in S}\boldsymbol{x}_i\cdot\prod_{i\in T}\boldsymbol{y}_i\cdot\prod_{i\in U}\boldsymbol{z}_i\right] \\
&= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}\left[\prod_{i\in S}\boldsymbol{x}_i\cdot\prod_{i\in T}\boldsymbol{y}_i\cdot\prod_{i\in U}\boldsymbol{x}_i\boldsymbol{y}_i\right] \qquad [\text{since } \boldsymbol{z} = \boldsymbol{x}\circ\boldsymbol{y}] \\
&= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}\left[\prod_{i\in S\triangle U}\boldsymbol{x}_i\cdot\prod_{i\in T\triangle U}\boldsymbol{y}_i\right] \qquad [\text{since } \boldsymbol{x}_i^2 = \boldsymbol{y}_i^2 = 1].
\end{aligned}$$

Now since $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent and since the expectation of the product of independent random variables is the product of their expectations we get

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})] = \mathop{\mathbf{E}}_{\boldsymbol{x}\in\{-1,1\}^n}\left[\prod_{i\in S\triangle U}\boldsymbol{x}_i\right]\cdot\mathop{\mathbf{E}}_{\boldsymbol{y}\in\{-1,1\}^n}\left[\prod_{i\in T\triangle U}\boldsymbol{y}_i\right].$$

Since $\boldsymbol{x}$ and $\boldsymbol{y}$ are random strings in $\{-1,1\}^n$ their coordinates are mutually independent and so

$$\begin{aligned}
\mathop{\mathbf{E}}_{\boldsymbol{x}\in\{-1,1\}^n}\left[\prod_{i\in S\triangle U}\boldsymbol{x}_i\right]\cdot\mathop{\mathbf{E}}_{\boldsymbol{y}\in\{-1,1\}^n}\left[\prod_{i\in T\triangle U}\boldsymbol{y}_i\right] &= \prod_{i\in S\triangle U}\mathop{\mathbf{E}}_{\boldsymbol{x}_i\in\{-1,1\}}[\boldsymbol{x}_i]\cdot\prod_{i\in T\triangle U}\mathop{\mathbf{E}}_{\boldsymbol{y}_i\in\{-1,1\}}[\boldsymbol{y}_i] \\
&= \begin{cases} 1 & \text{when } S\triangle U = \emptyset \text{ and } T\triangle U = \emptyset, \\ 0 & \text{otherwise.}\end{cases}
\end{aligned}$$

But $S \triangle U = \emptyset$ and $T \triangle U = \emptyset$ is equivalent to $S = T = U$. And so looking back at equation (1) we find that

$$\sum_{S,T,U \subseteq [n]} \hat{f}(S)\hat{f}(T)\hat{f}(U) \cdot \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y} \in \{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})] = \sum_{S \subseteq [n]} \hat{f}(S)^3.$$

This completes the proof. □

Remark 1.2 *For many tests one can analyze the probability of acceptance in a similar way; the proof of the above lemma is a good template to follow.*

Notation 1.3 *For brevity, if $x \in \{-1,1\}^n$, $S \subseteq [n]$ we will often write $x_S = \chi_S(x) = \prod_{i \in S} x_i$.*

Observation 1.4 *A few observations from the proof of the lemma:*

- $x_S \cdot x_T = x_{S \triangle T}$,
- $\mathbf{E}_{\boldsymbol{x} \in \{-1,1\}^n}[\boldsymbol{x}_S] = \begin{cases} 1 & \text{when } S = \emptyset, \\ 0 & \text{otherwise.} \end{cases}$
- $\mathbf{E}[f] = \mathbf{E}_{\boldsymbol{x} \in \{-1,1\}^n}[f(\boldsymbol{x}) \cdot 1] = \mathbf{E}_{\boldsymbol{x} \in \{-1,1\}^n}[f(\boldsymbol{x})\boldsymbol{x}_\emptyset] = \hat{f}(\emptyset)$.

With Lemma 1.1 under our belts, the proof of the following theorem requires just a few lines of reasoning.

Theorem 1.5 *If $f : \{-1,1\}^n \to \{-1,1\}$ is ϵ-far from being a parity function then*

$$\mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{y} \in \{-1,1\}^n}[\text{BLR}(f) \text{ accepts}] < 1 - \epsilon.$$

Proof: The proof is by contraposition. So suppose that

$$\mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{y} \in \{-1,1\}^n}[\text{BLR}(f) \text{ accepts}] \geq 1 - \epsilon.$$

Then from Lemma 1.1 we have

$$1 - \epsilon \leq \mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{y} \in \{-1,1\}^n}[\text{BLR}(f) \text{ accepts}] = \tfrac{1}{2} + \tfrac{1}{2} \sum_{S \subseteq [n]} \hat{f}(S)^3.$$

This in turn implies

$$1 - 2\epsilon \leq \sum_{S \subseteq [n]} \hat{f}(S)^3 = \sum_{S \subseteq [n]} \hat{f}(S)^2 \cdot \hat{f}(S) \leq \left(\max_{S \subseteq [n]} \hat{f}(S) \right) \cdot \sum_{S \subseteq [n]} \hat{f}(S)^2,$$

where the last step used the fact that all $\hat{f}(S)^2$ are nonnegative. Since f is boolean-valued, Parseval's Theorem tells us that $\sum_{S \subseteq [n]} \hat{f}(S)^2 = \mathbf{E}[f^2] = 1$, and we are left with the inequality $1 - 2\epsilon \leq \max_{S \subseteq [n]} \hat{f}(S)$. But this inequality implies that there exists a $T \subseteq [n]$ such that $\hat{f}(T) \geq 1 - 2\epsilon$. Recall that the Fourier coefficient $\hat{f}(T)$ just measures the correlation of f with the function Parity-on-T, i.e., $\hat{f}(T) = \langle f, \chi_T \rangle$. Hence f and χ_T agree on at least a $1 - \epsilon$ fraction of all strings; i.e., f is not ϵ-far from being a parity function. □

2 Local Testability and Decodability

What does Theorem 1.5 tell us in terms of testing whether a function is a parity function or not? It says that if our candidate function f is more than ϵ far from being a parity function then we have an ϵ chance of catching a bug. Function testing is a hot topic right now. Before discussing this in more detail we introduce the following definition.

Definition 2.1 *A property $\mathcal{P}$ of Boolean functions is called* locally testable *if there exists a randomized querying algorithm $\mathcal{T}$ making at most $O(1)$ queries such that:*

- *If $f \in \mathcal{P}$ then $\mathbf{Pr}[\mathcal{T}$ accepts$] = 1$.*
- *If f is ϵ-far from every $g \in \mathcal{P}$ then $\mathbf{Pr}[\mathcal{T}$ accepts$] \leq 1 - \Omega(\epsilon)$.*

Remark 2.2 *In this class we will only assume non-adaptive tests, i.e., tests that don't adjust their next query based on the results of previous queries.*

As an example, the BLR test shows that the property of being a parity function is locally testable with three queries. The following more general definition of property testing is due to Rubinfeld and Sudan.

Definition 2.3 *A property $\mathcal{P}$ of boolean functions is* testable with $q(\epsilon)$ queries *if there exists a randomized algorithm $\mathcal{T}$ (which gets ϵ as input) such that for all $\epsilon > 0$ it makes $q(\epsilon)$ queries and satisfies:*

- *If $f \in \mathcal{P}$ then $\mathbf{Pr}[\mathcal{T}$ accepts$] \geq \frac{2}{3}$.*
- *If f is ϵ-far from $\mathcal{P}$ then $\mathbf{Pr}[\mathcal{T}$ accepts$] \leq \frac{1}{3}$.*

By running the test multiple times independently and accepting if and only if at least half of the tests accept, one can boost $2/3$ vs. $1/3$ to $1-\delta$ vs. δ at the expense an extra multiplicative factor of $O(\log(1/\delta))$ in the number of queries. There are many more variations on the notion of testing ("one-sided error", adaptivity vs. nonadaptivity, letting $q(\epsilon)$ also depend on $n, \ldots$) but the above definition will suffice for us. Now using the Rubinfeld and Sudan definition above we can make the following observation about the BLR test:

Corollary 2.4 *The property of being a parity function is testable with $O(1/\epsilon)$ queries.*

Proof: Execute the following steps

- Run the BLR test $2/\epsilon$ times, independently.
- Overall accept if every test accepts.

The overall number of queries is $6/\epsilon$. Notice that if $f \in \mathcal{P}$ then the probability that the algorithm accepts is 1. On the other hand if f is ϵ-far from $\mathcal{P}$ then

$$\mathbf{Pr}[\text{Overall accept}] < (1-\epsilon)^{2/\epsilon} \approx \frac{1}{e^2} < \frac{1}{3}.$$

$\square$

The following idea will prove to be very useful if we have access to a function that is promised to be close to some parity function. Without telling *which* parity the function is close to, we can get at its values correctly, with high probability:

Proposition 2.5 *The set of parity functions $\mathcal{P}$ is* locally decodable with 2 queries, *meaning there exists a randomized 2-query algorithm $\mathcal{T}$ with access to a function $f : \{-1,1\}^n \to \{-1,1\}$ such that:*

- *If f is promised to be ϵ-close to a parity function, say χ_S, then for* every *string $x \in \{-1,1\}^n$, given x as input*
$$\mathop{\mathbf{Pr}}_{\mathcal{T}\text{'s randomness}}[\mathcal{T}(x) = x_S] \geq 1 - 2\epsilon.$$

Remark 2.6 *It's very important to note that the definition does not say that "for almost all x, $\mathcal{T}$ computes x_S correctly". To do that is trivial — $\mathcal{T}$ could just query on x. Instead, for* all *x, $\mathcal{T}$ gets the right answer with high probability over its internal randomness. This probability can be boosted by repeated independent trials, as long as $\epsilon < 1/2$.*

Proof: Given x, $\mathcal{T}$ picks $\boldsymbol{y} \in \{-1,1\}^n$ at random and returns $f(\boldsymbol{y})f(x \circ \boldsymbol{y})$. Since $\boldsymbol{y}$ is uniformly distributed, $\mathbf{Pr}_{\boldsymbol{y}\in\{-1,1\}^n}[f(\boldsymbol{y}) = \boldsymbol{y}_S] \geq 1-\epsilon$. Note that $x \circ \boldsymbol{y}$ is also uniformly distributed (although it is not independent of $\boldsymbol{y}$) and so

$$\mathop{\mathbf{Pr}}_{\boldsymbol{y}\in\{-1,1\}^n}[f(x \circ \boldsymbol{y}) = (x \circ \boldsymbol{y})_S] \geq 1 - \epsilon.$$

By the union bound with probability at least $1 - 2\epsilon$ both events occur; in this case $\mathcal{T}$ returns

$$\boldsymbol{y}_S(x \circ \boldsymbol{y})_S = \boldsymbol{y}_S x_S \boldsymbol{y}_S = x_S.$$

$\square$

We will now move on to develop algorithms for a related class of functions: the dictator functions.

3 Testing Dictator Functions

Definition 3.1 *The* dictator *functions on n bits are just the functions $\chi_{\{1\}}, \chi_{\{2\}}, \ldots, \chi_{\{n\}}$; i.e., the functions of the form $f(x) = x_i$. For notational simplicity we will write χ_i in place of $\chi_{\{i\}}$.*

Dictators are probably the most important class of functions for local testing, due to connections to probabilistically checkable proofs of proximity, and to hardness of approximation.

The following test looks very similar to the BLR test, but it flips each bit in the product with some small probability, given by a parameter δ.

Definition 3.2 *"Håstad's Test", parameterized by $\delta \in [0,1]$:*
Given query access to $f : \{-1,1\}^n \to \{-1,1\}$:

- *Pick $\boldsymbol{x}$, $\boldsymbol{y} \in \{-1,1\}^n$ uniformly and independently.*
- *Pick $\boldsymbol{w} \in \{-1,1\}^n$ with the δ-biased product distribution (i.e., $\mathbf{Pr}[w_i = -1] = \delta$, $\mathbf{Pr}[w_i = 1] = 1-\delta$ independently across i's).*
- *Set $\boldsymbol{z} = \boldsymbol{x} \circ \boldsymbol{y} \circ \boldsymbol{w}$.*
- *"Accept" iff $f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z}) = 1$.*

We now proceed with an analysis of Håstad's Test similar to that of the BLR test, i.e. we start with a result that expresses the probabilty that the test accepts in terms of the Fourier expansion of the function.

Theorem 3.3 *Given $f : \{-1,1\}^n \to \{-1,1\}$,*

$$\mathbf{Pr}[\textit{Håstad}_\delta(f) \textit{ accepts}] = \tfrac{1}{2} + \tfrac{1}{2}\sum_{S\subseteq[n]} (1-2\delta)^{|S|}\hat{f}(S)^3.$$

Proof: The analysis follows the BLR test almost exactly for the first few steps. Since $f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z}) \in \{-1,1\}$ we can use the following indicator random variable for the event that Håstad's Test accepts

$$\mathbf{1}_{\text{Håstad}} = \tfrac{1}{2} + \tfrac{1}{2}f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z}).$$

Therefore by linearity of expectation

$$\mathop{\mathbf{Pr}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\text{Håstad}(f) \text{ accepts}] = \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\mathbf{1}_{\text{Håstad}}] = \tfrac{1}{2} + \tfrac{1}{2}\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})].$$

We now focus on $\mathbf{E}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})]$. Expressing $f(\boldsymbol{x})$, $f(\boldsymbol{y})$ and $f(\boldsymbol{z})$ in terms of their Fourier expansions we obtain the following:

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})] =$$
$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}\left[\left(\sum_{S\subseteq[n]}\hat{f}(S)\chi_S(\boldsymbol{x})\right)\left(\sum_{T\subseteq[n]}\hat{f}(T)\chi_T(\boldsymbol{y})\right)\left(\sum_{U\subseteq[n]}\hat{f}(U)\chi_U(\boldsymbol{z})\right)\right].$$

Distributing out the product of sums (into a sum of products) yields

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})] = \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}\left[\sum_{S,T,U\subseteq[n]} \hat{f}(S)\hat{f}(T)\hat{f}(U)\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})\right]$$

By linearity of expectation this is equivalent to

$$\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})] = \sum_{S,T,U\subseteq[n]}\left(\hat{f}(S)\hat{f}(T)\hat{f}(U)\cdot \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})]\right). \tag{2}$$

At this point we wil focus on simplifying the term $\mathbf{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})]$ in the equation above. It is here that the analysis starts to differ from the BLR analysis.

$$\begin{aligned}
\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})] &= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\boldsymbol{x}_S\boldsymbol{y}_T(\boldsymbol{x}\cdot\boldsymbol{y}\cdot\boldsymbol{w})_U] \qquad [\text{since } \boldsymbol{z}=\boldsymbol{x}\circ\boldsymbol{y}\circ\boldsymbol{w}]\\
&= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\boldsymbol{x}_S\boldsymbol{y}_T\boldsymbol{x}_U\boldsymbol{y}_U\boldsymbol{w}_U]\\
&= \mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[(\boldsymbol{x}_{S\triangle U})(\boldsymbol{y}_{T\triangle U})\boldsymbol{w}_U]\\
&= \mathop{\mathbf{E}}_{\boldsymbol{x}\in\{-1,1\}^n}[\boldsymbol{x}_{S\triangle U}]\cdot\mathop{\mathbf{E}}_{\boldsymbol{y}\in\{-1,1\}^n}[\boldsymbol{y}_{T\triangle U}]\cdot\mathop{\mathbf{E}}_{\boldsymbol{w}\in\{-1,1\}^n}[\boldsymbol{w}_U],
\end{aligned}$$

where the last equation follows from the independence of $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{w}$. Since $\boldsymbol{x}$ is chosen uniformly at random from $\{-1,1\}^n$ it follows that $\mathbf{E}_{\boldsymbol{x}\in\{-1,1\}^n}[S\triangle U] = 1$ if $S\triangle U=\emptyset$ and zero otherwise; similarly for $\boldsymbol{y}_{T\triangle U}$. So it follows that

$$\begin{aligned}
\mathop{\mathbf{E}}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{w}\in\{-1,1\}^n}[\chi_S(\boldsymbol{x})\chi_T(\boldsymbol{y})\chi_U(\boldsymbol{z})] &= \mathop{\mathbf{E}}_{\boldsymbol{x}\in\{-1,1\}^n}[\boldsymbol{x}_{S\triangle U}]\cdot\mathop{\mathbf{E}}_{\boldsymbol{y}\in\{-1,1\}^n}[\boldsymbol{y}_{T\triangle U}]\cdot\mathop{\mathbf{E}}_{\boldsymbol{w}\in\{-1,1\}^n}[\boldsymbol{w}_U]\\
&= \begin{cases} 0 & \text{unless } S=T=U,\\ E[\boldsymbol{w}_S] & \text{otherwise.}\end{cases}
\end{aligned} \tag{3}$$

Since $\boldsymbol{w}$'s coordinates are mutually independent, we can conclude that

$$\mathop{\mathbf{E}}_{\boldsymbol{w}\in\{-1,1\}^n}[\boldsymbol{w}_S] = \prod_{i\in S}\mathbf{E}[\boldsymbol{w}_i] = \prod_{i\in S}(1-2\delta) = (1-2\delta)^{|S|}.$$

Substituting this into (3) and then into (2) proves the theorem. □

3.1 A First Attempt at a Dictator Test

With Theorem 3.3 we will take a first pass at designing a test for dictator functions.

Suppose we are given $\epsilon > 0$ as input and given query access to $f : \{-1,1\}^n \to \{-1,1\}$. Set $\epsilon = \min(\epsilon, .01)$, and then run Håstad's Test with $\delta = .75\epsilon$. Let's see if we can get some kind of dictator test out of this.

Suppose that f is a dictator, i.e., $f = \chi_i$ for some $i \in [n]$. Note that f's Fourier expansion has only one nonzero coefficient, $\hat{f}(\{i\}) = 1$. Thus using Theorem 3.3:

$$\mathbf{Pr}[f \text{ passes}] = \tfrac{1}{2} + \tfrac{1}{2}(1-2\delta)^1 \cdot 1^3 = 1 - \delta = 1 - .75\epsilon.$$

On the other hand, suppose

$$1 - \epsilon \leq \mathbf{Pr}[f \text{ passes}] = \tfrac{1}{2} + \tfrac{1}{2}\sum_{S \subseteq [n]} (1-2\delta)^{|S|} \hat{f}(S)^3. \quad \text{[By Theorem 3.3]}$$

This in turn implies that

$$1 - 2\epsilon \leq \left(\sum_{S \subseteq [n]} \hat{f}(S)^2 \right) \max_{S \subseteq [n]} \left\{ (1-2\delta)^{|S|} \hat{f}(S) \right\} = \max_{S \subseteq [n]} (1-2\delta)^{|S|} \hat{f}(S),$$

where we used Parseval's Theorem in the last step. Since $(1-2\delta)^{|S|} \leq 1$ it follows that there exists some subset S such that $\hat{f}(S) \geq 1 - 2\epsilon$. Next observe that

$$1 - 2\epsilon \leq \max_{S \subseteq [n]} (1-2\delta)^{|S|} \hat{f}(S) = \max_{S \subseteq [n]} (1-1.5\epsilon)^{|S|} \hat{f}(S).$$

Using the fact that $(1 - 1.5\epsilon)^{|S|} < 1 - 2\epsilon$ for all $|S| \geq 2$ (this also uses that ϵ is sufficiently small; specifically, $\epsilon < .01$) we know that the maximum must occur for some S such that $|S| \leq 1$. In other words, there exists a Fourier coefficient of magnitude at least $1 - 2\epsilon$ on a subset of cardinality at most one. But the Fourier coefficients are simply the correlation of f with the parity functions, and so we have

$$1 - 2\epsilon \leq \max_{|S| \leq 1} \hat{f}(S) = \max_{|S| \leq 1} \langle f, \chi_S \rangle.$$

Thus f is ϵ-close to either a dictator function *or to 1*. So we *almost* have a dictator test — if f belongs to the class of functions $\{\text{dictators}\} \cup \{1\}$ then $\mathbf{Pr}[\text{test accepts}] \geq 1 - .75\epsilon$. On the other hand if f is ϵ-far from the class of functions $\{\text{dictators}\} \cup \{1\}$ then $\mathbf{Pr}[\text{test accepts}] \leq 1 - \epsilon$, i.e., it accepts with a slightly smaller probability.

Corollary 3.4 ("Almost Dictator Test") *The class $\{\textit{dictators}\} \cup \{1\}$ is testable with $O(1/\epsilon^2)$ queries.*

Proof: Given $f : \{-1,1\}^n \to \{-1,1\}$ and $\epsilon > 0$, set $\epsilon = \min(\epsilon, .01)$ and then:

- Run Håstad's Test $O(1/\epsilon^2)$ times with $\delta = .75\epsilon$.
- Overall accept if empirical fraction of Håstad passes is at least $1 - .8\epsilon$.

The result now follows from an application of the Chernoff bound. □

3.2 Fixing the Dictator Test

There are at least a couple of different ways to fix the "Almost Dictator test" so that it also rejects the 1 function.

One suggestion made in class: Apply the "Almost Dictator Test" but additionally reject if the proportion of $f(\boldsymbol{x})$'s and $f(\boldsymbol{y})$'s that were 1 was at least $3/4$. Recall that the $\boldsymbol{x}$'s and $\boldsymbol{y}$'s are independent random strings. Thus the dictators (which are 1 on half the strings and -1 on the other half) will pass the extra test except with exponentially small probability in ϵ.

On the other hand, if f is ϵ-far from the set of dictators then$f$ is also ϵ-far from the set of functions $\{\text{dictators}\} \cup \{1\}$. Then the "Almost Dictator Test" will reject f with probability at least $2/3$ except when it is ϵ-close to 1. But in the latter case the additional test will reject the test function except with exponentially small probability in ϵ.

Another way to fix the "Almost Dictator Test" uses the local decodability of parity functions. For this fix, after running the "Almost Dictator Test" we do local decoding (2 more queries) on the string $(-1, -1, \ldots, -1)$ and reject if the answer is 1.

If we still haven't rejected after the "Almost Dictator Test" then f is ϵ-close to $\{1\} \cup \{\text{dictators}\}$ with probability at least $2/3$. Since all of these functions are parities, our local decoding will work except with probability 2ϵ. The correct value on $(-1, \ldots, -1)$ is -1 for dictators, so overall they pass with probability at least $2/3 - 2\epsilon$ (this is large enough when we assume that $\epsilon < .01$). But any function that is ϵ-close to 1 will pass this additional test with probability at most 3ϵ, so overall it will pass with probability at most $1/3 + 3\epsilon$, which is also fine.