Lecture 11: Application of restrictions

Rajat Mittal

IIT Kanpur

We have seen restrictions of Boolean functions when a few variables are set to a particular value. Given a Boolean function $f : \{-1,1\}^n \to \{-1,1\}$, we will generalize this to the case when a subset of parities, say

$$\Gamma \subseteq \{\chi_S : S \subseteq [n]\},$$

are set to assignment $b \in \{-1,1\}^{\Gamma}$. We will show two applications of restricted functions: relation between rank and sparsity and better bounds in learning theory.

1 Restricting a subset of parities

For a function $f : \{-1,1\}^n \to \{-1,1\}$, a set of parities Γ and an assignment $b \in \{-1,1\}^{\Gamma}$, we define the restriction

$$f_{\Gamma,b} := f|_{\{x\in\{-1,1\}^n : \chi_\gamma(x)=b_\gamma \text{ for all } \gamma\in\Gamma\}}.$$

Let us take an example, let f be the Majority function on 3 variables. What is the restricted function when $x_1x_2 = -1$, i.e., $\Gamma = \{x_1x_2\}$ and $b = -1$.

Exercise 1. What is the Fourier expression for Majority of 3 variables?

A small calculation shows that,

$$\text{MAJORITY}_3(x) = 1/2(x_1 + x_2 + x_3) - 1/2x_1x_2x_3.$$

If we substitute $x_1x_2 = -1$, we will get

$$1/2(x_1 + x_2) + 1/2x_3.$$

Exercise 2. Is this the correct answer? Look at the inputs where the restricted function is defined.

The function will give the correct answer but it is not in the simplest form. Notice that $x_1x_2 = -1$ means the value of x_1 and x_2 is different, that means $Maj_{\{x_1x_2\},-1}$ should just be the value of x_3. Though, the Fourier expression obtained is slightly more involved.

Looking at all parities as vectors in $\mathbb{F}_2^n$, the vectors corresponding to x_1, x_2 and x_1x_2 are linearly dependent. That means, setting $x_1x_2 = -1$ gives a relation between x_1 and x_2. In other words, $x_1 = -x_2$, substituting this in the previous expression will give,

$$\text{MAJORITY}_{\{x_1x_2\},-1} = x_3.$$

What happens to the Fourier expression of a general function when some parities are restricted? Let us define it formally.

Equivalence relation for a set of parities Let f be a Boolean function. Given a set of parities Γ, define the following equivalence relation among parities in supp(f).

$$\forall \gamma_1, \gamma_2 \in \text{supp}(f), \gamma_1 \equiv \gamma_2 \text{ iff } \gamma_1 + \gamma_2 \in \text{span}(\Gamma).$$

Let ℓ be the number of equivalence classes according to the equivalence relation for Γ. For $j \in [\ell]$, let k_j be the size of the j-th equivalence class. Since the equivalence classes form a partition of supp(f), we have

Claim. Following the notation of the paragraph above, $\sum_{j=1}^{\ell} k_j = \text{spar}(f)$.

Let $\beta_1, \ldots, \beta_\ell \in \text{supp}(f)$ be some representatives of the equivalence classes. For $j \in [\ell]$, let $\beta_j + \alpha_{j,1}, \ldots, \beta_j + \alpha_{j,k_j}$ be the elements of the j-th equivalence class. This notation gives a compact representation of f in terms of these equivalence classes. For all $x \in \{-1,1\}^n$,

$$f(x) = \sum_{j=1}^{\ell} P_j(x)\chi_{\beta_j}(x), \tag{1}$$

where

$$P_j(x) = \sum_{r=1}^{k_j} \hat{f}(\beta_j + \alpha_{j,r}) \cdot \chi_{\alpha_{j,r}}(x). \tag{2}$$

Note that P_j are non-zero multilinear polynomials and depend only on the parities in Γ. So, fixing parities in Γ collapses all the parities in an equivalence class to their representative, thereby making P_j's constant.

2 Rank vs Sparsity

As our first application, we will give a near optimal upper bound on rank in terms of the sparsity of the function f. A priori, we only know

$$\log(\text{spar}(f)) \leq \text{rank}(f) \leq \text{spar} f.$$

This is even true when the function is non-Boolean. We have seen that $\text{rank}(f)$ can even be $\sqrt{\text{spar}(f)}$.

Exercise 3. What function achieves this relation?

Can there exist functions where $\text{rank}(f)$ is higher than $\sqrt{\text{spar}(f)}$?

One of the important result (called Tsang's lemma in this text) from the paper of Tsang et al. is going to be really helpful.

Lemma 1 ([1]). *Let $f : \{-1,1\}^n \to \{-1,1\}$ a Boolean function. There exist a set of parties Γ and an assignment $b \in \{-1,1\}^\gamma$ such that $f_{\Gamma,b}$ is constant and*

$$|\Gamma| \leq 3\sqrt{\text{spar}(f)}.$$

The above lemma is stated slightly differently in [1]; they use Fourier 1-norm instead of $\text{spar}(f)$, but they can be interchanged using Parseval and Cauchy-Schwarz. We will *not* prove Tsang's lemma.

Exercise 4. Prove that the Fourier 1-norm of f is bounded by $\text{spar}(f)$.

Let $k = \text{spar}(f)$ and $r = \text{rank} f$, using Tsang's lemma we can easily show $r \leq (k/2) + \sqrt{k}$. The important observation is

Claim. If setting parities in Γ makes the function constant, every parity in $\text{supp}(f)$ must have been paired with at least one other parity in the equivalence class.

Using this observation, you will show in the assignment that $r \leq (k/2) + \sqrt{k}$. We are still far away from $\sqrt{k}$ (the bound for addressing function). The best known upper bound on the Fourier rank of a Boolean function in terms of the sparsity of the function was given by Sanyal [2]. He showed that for any Boolean function f,

$$r = O(\sqrt{k} \log k).$$

In rest of this section we will see a proof of this result. We will give an algorithm which takes a Boolean function as an input and outputs $O(\sqrt{k} \log k)$ parities such that *any* assignment of these parities makes the function constant. This gives an upper bound on Fourier rank of the function since the Fourier support of the function must be contained in the span of this set of parities.

Exercise 5. Convince yourself of the statement above.

As already noticed by Observation 2, Lemma 1 gives us to fix small number of parities such that the sparsity of every possible restriction (for any b) is at least $k/2$. In other words, there are less than $k/2$ active parities now (the representatives of each equivalence class), and support of each restriction is a subset of these active parities. The main idea is to apply Tsang's lemma repeatedly.

Suppose we apply Tsang's lemma for the first time, the set of parities get bundled in at least a group of two and the set of active parities become less than $k/2$. Ideally, we want to apply Tsang's lemma on $k/2$ parities and reduce them further to $k/4$ active parities. Such an algorithm will need $\sqrt{k}+\sqrt{k/2}+\sqrt{k/4}+\cdots =$ $O(\sqrt{k \log k}$ parities and the set of active parities will become empty. That means, all restrictions become constant after fixing these $O(\sqrt{k \log k})$ parities.

Unfortunately, there is no Boolean function corresponding to these $k/2$ active parities. Every restriction will have some subset of these $k/2$ parities active, if the subset is small then applying Tsang's lemma will not reduce the set of active parities much. The resolution is to *greedily* pick the restriction with largest sparsity at each step and apply Tsang's lemma. A bit of Fourier analysis will show that there always exist a restriction whose sparsity is comparable to the set of active parities.

Exercise 6. Write the algorithm formally.

2.1 Analysing the algorithm:

It remains to show that the number of parities fixed in our algorithm is small. Given a Boolean function f and a set of parities Γ over the set of the variables of f, remember the equivalence relation over $\text{supp}(f)$,

$$\forall \gamma_1, \gamma_2 \in \text{supp}(f), \gamma_1 \equiv \gamma_2 \text{ iff } \gamma_1 + \gamma_2 \in \text{span}(\Gamma).$$

Let ℓ be the number of partitions of the Fourier support of f with respect to the equivalence relation corresponding to Γ. We first show that there exist a restriction with high sparsity for any Γ (remember that Tsang's lemma is applied such a restriction at each step of the algorithm).

Lemma 2. *Given a function Boolean function f and a subset of parities Γ, there exist an assignment $b \in \{-1,1\}^{\Gamma}$ such that*

$$\text{supp}(f_{\Gamma,b}) \geq \frac{\ell^2}{k}.$$

Proof. Using the notation defined in previous section,

$$f_{\Gamma,b} = \sum_{j=1}^{\ell} P_j(b)\chi_{\beta_j}(x).$$

The sparsity of $f_{\Gamma,b}$ is the number of non-zero $P_j(b)$'s. $P_j(x)$ is itself a polynomial with sparsity k_j (the size of j-th equivalence class).

By Uncertainity principle on Boolean hypercube (proved in last assignment),

$$\Pr_b(P_j(b) \neq 0) \geq 1/k_j.$$

Hence, the expected value of $\text{supp}(f_{\Gamma,b})$ is

$$\sum_{j=1}^{\ell} \Pr_b(P_j(b) \neq 0) \geq \sum_{j=1}^{\ell} 1/k_j.$$

Applying Jensen's inequality on $1/x$ (it is a convex function),

$$\mathbf{E}[\text{supp}(f_{\Gamma,b})] \geq \ell \frac{1}{\frac{\sum_j k_j}{\ell}} \geq \frac{\ell^2}{k}.$$

Exercise 7. Read about Jensen's inequality.

This finishes the proof of the lemma. □

To bound the total number of parties fixed in our algorithm, we would like to bound the number of parities included in the i-th iteration. Let us denote Γ after the i-th iteration by $\Gamma^{(i)}$ ($\Gamma^{(0)} = \emptyset$). Let ℓ_i be the number of partitions of the Fourier support of f with respect to the equivalence relation corresponding to $\Gamma^{(i)}$ (the set of active parities). Say the algorithm fixes q_i parities in the i-th iteration. Let $f^{(i)}$ be the selected function after the i-th iteration ($f^{(0)} = f$).

By Tsang' lemma, q_i is of the order of the square root of the sparsity of $f^{(i-1)}$. The reduction in the size of active parities, $\ell_{i-1} - \ell_i$ is at least $\text{spar}(f^{(i-1)})/2$. This implies,

$$\frac{q_i}{\ell_{i-1} - \ell_i} = O(1/\text{spar}(f^{(i-1)})).$$

Using Lemma 2,

$$q_i = (\ell_{i-1} - \ell_i) O(\frac{\sqrt{k}}{\ell_{i-1}}).$$

The total number of parities fixed by our algorithm is the sum of q_i's.

$$\sum_i q_i = O(\sqrt{k}) \sum_i \frac{\ell_{i-1} - \ell_i}{\ell_{i-1}} \tag{3}$$

You will show in the assignment that $\sum_i \frac{\ell_{i-1} - \ell_i}{\ell_{i-1}} \leq \sum_{i=1}^{k} 1/i$.

Exercise 8. Show that the number of parities fixed are $O(\sqrt{k \log k})$.

3 Learning theory (again)

The contents of this section are taken from the book Analysis of Boolean Functions from Ryan O'Donnell [3].

We have already seen that it is easy to *learn* a function if we know that it is ϵ-concentrated on a small family $\mathcal{F}$ of subsets of parities.

Theorem 1. *Given access to random queries to $f : \{-1,1\}^n \to \{-1,1\}$, if we know $\mathcal{F}$ such that f is $\frac{\epsilon}{2}$-concentrated on $\mathcal{F}$, then f can be* learned *in time poly$(|\mathcal{F}|, n, \frac{1}{\epsilon})$.*

The theorem was proved before in class. Suppose, we know that f is ϵ-concentrated but don't know the $\mathcal{F}$ on which it is concentrated. Goldreich-Levin algorithm, explained below, allows us to find the collection $\mathcal{F}$ efficiently. Though, in this case, we will need query access to the function f (and not just random access).

Given query access (you can ask $f(x)$ for any x), Golreich-Levin algorithm gives a list of parities such that the list contains all big Fourier coefficients.

Theorem 2 (Goldreich-Levin). *Given a Boolean function $f : \{-1,1\}^n \to \{-1,1\}$, a threshold τ and query access to f, we can output a list L of parities satisfying,*

- *if $\left|\hat{f}(S)\right| \geq \tau$ then $S \in L$,*
- *and if $S \in L$ then $\left|\hat{f}(S)\right| \geq \tau/2$,*

in time poly$(n, 1/\tau)$.

Suppose f is ϵ-concentrated on some family $\mathcal{F}$. We can run Goldreich-Levin with $\tau = \sqrt{\frac{\epsilon}{|\mathcal{F}|}}$. This will output a list such that f is 2ϵ-concentrated on this list.

Exercise 9. Prove the above statement. Show that the size of the list L is $\text{poly}(1/\epsilon, |\mathcal{F}|)$ using Parseval's identity.

Using Theorem 1, we can efficiently ($\text{poly}(n, 1/\epsilon, |\mathcal{F}|)$ time) learn f.

Exercise 10. We have shown that f can be learned if the *concept class* is the class of all functions ϵ-concentrated on some small family $\mathcal{F}$. Observe that this can be generalized (without any extra work) to the class of all functions whose Fourier spectra is concentrated on small number of parities (say M).

Our final task is to prove Goldreich-Levin.

Proof of Goldreich-Levin. We already know how to estimate a particular Fourier coefficient using random access (it is expectation of $\chi_S(x)f(x)$). Unfortunately, there are too many Fourier coefficients to estimate. The main idea is to divide these coefficients into *buckets* and estimate their weight. If $k < n$, let $S \subseteq [k]$ and define

$$W^{S,k} = \sum_{T \subseteq \{k+1, k+2, \cdots, n\}} \hat{f}(S \cup T)^2.$$

Lemma 3 will show that $W^{S,k}$ can be estimated efficiently with error ϵ and probability $1 - \delta$ (in time $\text{poly}(n, 1/\epsilon, \log(1/\delta))$).

The idea is to start with the full bucket $W^{\emptyset,0}$ and divide it into two parts: $W^{\emptyset,1}$ and $W^{\{1\},1}$. The division will continue further and $W^{S,k}$ will be divided into $W^{S,k+1}$ and $W^{S\cup\{k+1\},k+1}$. If we find a bucket with weight less than $\tau^2/2$, we will discard that bucket. Finally, we will be left with lots of buckets having just one element; those are the elements of L.

If we estimate the bucket weights with error less than $\tau^2/4$, then a Fourier coefficient with weight $\geq \tau^2$ will never be discarded and any coefficient with weight less than $\tau^2/4$ will not be in the list. This shows that the list given by the algorithm satisfies the required property.

Exercise 11. Convince yourself that the list satisfies the requirements of the theorem.

We only need to show that the number of estimates are small. Since every bucket has weight at least $\tau^2/4$, there are at most $4/\tau^2$ buckets at any point (Parseval). Each bucket gets split at most n times, so the number of estimates are at most $8n/\tau^2$.

The final algorithm is to keep estimating (splitting and discarding) buckets using Lemma 3 with error $\tau^2/4$ and with error probability $\tau^2/24n$.

Exercise 12. Show that the algorithm gives the correct list with constant error in time $\text{poly}(n, 1/\tau)$.

□

The above proof required the following lemma.

Lemma 3. *$W^{S,k}$ can be estimated with error ϵ and probability $1-\delta$ in time poly$(n, 1/\epsilon, \log(1/\delta))$.*

Proof. To estimate $W^{S,k}$, we require the expression in terms of restrictions proved in the class before.

$$W^{S,k} = \mathbf{E}_{z \in \{-1,1\}^{n-k}}[\widehat{f_{[k]|z}}(S)^2]$$

Remember that $f_{[k]|z}$ is the function where f is restricted to first k variables and last $n-k$ variables are set according to z. Since we know

$$\widehat{f_{[k]|z}}(S)^2 = \mathbf{E}_{y,y' \in \{-1,1\}^k}[\chi_S(y)f(y,z)\chi_S(y')f(y',z)],$$

the expression becomes

$$W^{S,k} = \mathbf{E}_{z \in \{-1,1\}^{n-k}, y \in \{-1,1\}^k, y' \in \{-1,1\}^k}[\chi_S(y)f(y,z)\chi_S(y')f(y',z)].$$

This can be estimated using Chernoff bound in the required time (same as when we estimated Fourier weights). □

4 Assignment

Exercise 13. Using Tsang's lemma show that $r \leq (k/2) + \sqrt{k}$.

Exercise 14. Look at Equation 3, prove $\sum_i \frac{\ell_{i-1} - \ell_i}{\ell_{i-1}} \leq \sum_{i=1}^{k} 1/i$.

References

1. Hing Yin Tsang, Chung Hoi Wong, Ning Xie and Shengyu Zhang, "Fourier Sparsity, Spectral Norm, and the Log-Rank Conjecture," 54th Annual IEEE Symposium on Foundations of Computer Science (FOCS), 2013.
2. Swagato Sanyal, "Fourier Sparsity and Dimension," Theory of Computing, 2019, Vol. 15, Number 11.
3. Ryan O'Donnell, "Analysis of Boolean Functions," Cambridge University Press, 2014.