Computer Science 294 Lecture 9 Notes

Daniel Raban

February 14, 2023

1 DNFs and Random Restrictions

1.1 Concentration and computability of DNFs

Our previous topic was about how if a function is concentrated on not too many Fourier coefficients, then it is easy to learn. Now, we will see how to show that certain classes of functions are Fourier concentrated.

Definition 1.1. A **DNF** is an Or of And functions.

Example 1.1. A DNF looks like

$$\underbrace{(x_1 \wedge \overline{x}_2 \wedge x_7)}_{\text{term}} \vee (x_2 \wedge x_5) \vee \cdots .$$

Definition 1.2. The **width** of a DNF is the maximum number of literals in a term. The **size** is the number of terms.

From its truth table, any boolean function has a DNF of size $\leq 2^n$ and width $\leq n$ computing it.

Definition 1.3. A **CNF** is an And of Or functions.

Example 1.2. A CNF looks like

$$\underbrace{(x_{17} \vee \overline{x_8} \vee x_9)}_{\text{clause}} \wedge \cdots .$$

De Morgan's laws tells us that if $f = T_1 \vee T_2 \vee \cdots \vee T_n$, then

$$\neg f = (\neg T_1) \wedge \cdots \wedge (\neg T_m) = C_1 \wedge \cdots \wedge C_n$$

is a CNF.

Exercise 1.1. If f s computable by a size s and depth d decision tree, then f can be computed by a size s and width d DNF (or CNF).

[insert picture 1]

Proposition 1.1. *Suppose f is computable by a width w DNF. Then $\mathbb{I}(f) \leq 2w$.*

Here is a strengthening we won't prove:

Theorem 1.1 (Amano)**.** *Suppose f is computable by a width w DNF. Then $\mathbb{I}(f) \leq w$.*

Amano's theorem is tight because parity on w bits can b written as a width w DNF.

Proof. Recall that

$$\mathbb{I}(f) = \frac{\#\text{ of sensitive edges}}{2^{n-1}}.$$

Every input x that satisfies a width w DNF has at most w neighbors $y \sim x$ (y and x differ in one variable) that don't satisfy the DNF.

$$\begin{aligned} &\leq \frac{\#\{x : f(x) = \text{True}\} \cdot w}{2^{n-1}} \\ &\leq \frac{2^n w}{2^{n-1}} \\ &= 2w. \end{aligned}$$

□

Corollary 1.1. *Width w DNFs are ε-concentrated up to degree w/ε.*

Corollary 1.2. *Width w DNFs can be PAC learned (under the uniform distribution) from random labeled examples in time $n^{O(w)}$.*

Exercise 1.2. If f is computable by a size s DNF, then for all $\varepsilon > 0$, there is a g computable by width $\log(s/\varepsilon)$ DNF such that $\text{dist}(f, g) \leq \varepsilon$.

Corollary 1.3. *Size s DNFs are ε-concentrated up to degree $\log(s/\varepsilon)/\varepsilon$.*

Corollary 1.4. *Size s DNFs can be PAC learned (under the uniform distribution) from random labeled examples in time $n^{O(\log s)}$.*

Next time we will show the following theorem, which is an improvement.

Theorem 1.2 (Mansour)**.** *Width w DNFs are ε-concentrated on at most $w^{O(w \log(1/\varepsilon))}$ coefficients. All these coefficients are up to degree $O(w \log(1/\varepsilon))$.*

Conjecture 1.1 (Mansour's conjecture)**.** *Width w DNFs are 0.01-concentrated on $2^{O(w)}$ coefficients. That is, size s DNFs are 0.01-concentrated on $s^{O(1)}$ coefficients.*

With the Goldreich-Levin algorithm, this would imply that size s DNFs are learnable in polynomial time. To prove Mansour's theorem, we will introduce the technique of random restrictions.

1.2 Random restrictions

The idea is that it can be easier to analyze a function after assigning some of the bits. Then we translated results on the restricted function to results on the original function.

Example 1.3. The DNF $x_1 \wedge \cdots \wedge x_n$ might seem complicated because it has width n, but if we randomly assign some constant fraction of the bits, then at least one of them should be false, which greatly simplifies the DNF.

Definition 1.4 (Restriction)**.** Let $f : \{\pm 1\}^n \to \{\pm 1\}$, $J \subseteq [n]$ be the set of coordinates we're going to keep alive, and $z \in \{\pm 1\}^{\overline{J}}$ be an assignment for the rest of the coordinates. The **restriction** of f is $f_{J,z} : \{\pm 1\}^J \to \{\pm 1\}$ with

$$f_{J,z}(y) = f(\underbrace{y}_{J}, \underbrace{z}_{\overline{J}}).$$

Example 1.4. For $n = 5$ and $J = \{1, 3, 5\}$, we can denote the non-restricted variables as *. So we could have $z = (*, 1, *, -1, *)$.

Example 1.5. The **Multiple XOR** function is

$$\mathrm{MUX}(x_1, x_2, x_3) = \begin{cases} x_2 & \text{if } x_1 = 1 \\ x_3 & \text{if } x_1 = -1. \end{cases}$$

The restriction to x_2, x_3 with $x_1 = 1$ is

$$\mathrm{MUX}_{\{2,3\},1}(x_2, x_3) = x_2.$$

The restriction to x_1, x_2 with $x_3 = -1$ is

$$\mathrm{MUX}_{\{1,2\},-1} = \begin{cases} x_2 & \textbf{if } x_1 = 1 \\ x_2 & \text{if } x_1 = -1 \end{cases} = \min(x_1, x_2).$$

We can also think of $f_{J,z} : \{\pm 1\}^n \to \{\pm 1\}$ as $f(y) = f(y_J, z)$. So we don't need to notationally keep track of how many variables $f_{J,z}$ is a function of.

Proposition 1.2. *For* $S \subseteq [n]$*,*

$$\widehat{f_{J,z}}(S) = \begin{cases} 0 & S \not\subseteq J \\ \sum_{T \subseteq \overline{J}} \widehat{f}(S \cup T) \prod_{i \in T} z_i & S \subseteq J \end{cases}$$

Proof. Write $f_{J,z}(y) = f(y_J, z)$. Then

$$\widehat{f_{J,z}}(S^*) = \mathbb{E}_Y[f_{J,z}(Y) \chi_{S^*}(Y)]$$

Recall that $f_{J,z}(y) = f(y_J, z) = \sum_S \widehat{f}(S)\chi_S(y_J, z) = \sum_{S\subseteq J}\sum_{T\subseteq \overline{J}} \widehat{f}(S\cup T)\chi_S(Y_j)\chi_T(z)$ and plug this in.

$$\begin{aligned}
&= \mathbb{E}_Y\left[\sum_{S\subseteq J}\sum_{T\subseteq \overline{J}} \widehat{f}(S\cup T)\chi_S(Y_j)\chi_T(z)\chi_{S^*}(Y)\right] \\
&= \sum_{S\subseteq J}\sum_{T\subseteq \overline{J}} \widehat{f}(S\cup T)\chi_T(z)\underbrace{\mathbb{E}_Y\,\chi_S(Y_j)\chi_{S^*}(Y)]}_{\mathbb{1}_{\{S=S^*\}}} \\
&= \begin{cases} \sum_{T\subseteq \overline{J}} \widehat{f}(S^*\cup T)\chi_T(z) & S^*\subseteq J \\ 0 & \text{otherwise.}\end{cases}
\end{aligned} \qquad \square$$

The above is for fixed restrictions. Usually, we want to pick the coordinates that we keep at random and assign random bits to the other coordinates. First, consider the common scenario where J is fixed and z is uniformly random. Then for fixed $S\subseteq J$,

$$h(z) := \widehat{f_{J,z}}(S) = \sum_{T\subseteq \overline{J}} \widehat{f}(S\cup T)\chi_T(z).$$

We have

$$\mathbb{E}_Z[h(Z)] = \widehat{f}(S),$$
$$\mathbb{E}_Z[h(Z)^2] = \sum_{T\subseteq \overline{J}} \widehat{f}(S\cup T)^2 = \sum_{U:U\cap J=S} \widehat{f}(U)^2.$$

For $S\not\subseteq J$, these equal 0.

Now, let's consider p**-random restrictions**: Pick $J\subseteq_p [n]$, here every $i\in[n]$ is picked independently with probability p. Then pick $Z\in\{\pm1\}^{\overline{J}}$ uniformly at random. We denote this distribution as $(J,Z)\sim\mathcal{R}_p$. Then we can calculate

$$\begin{aligned}
\mathbb{E}_{(J,Z)\sim\mathcal{R}_p}[\widehat{f_{J,z}}(S)] &= \mathbb{E}_J[\mathbb{E}_Z[\widehat{f_{J,Z}}(S)]] \\
&= \mathbb{E}_J[\mathbb{1}_{S\subseteq J}\widehat{f}(S)] \\
&= \widehat{f}(S)\mathbb{P}(S\subseteq J) \\
&= \widehat{f}(S)p^{|S|},
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}_{(J,Z)\sim\mathcal{R}_p}[\widehat{f_{J,z}}(S)^2] &= \mathbb{E}_J[\mathbb{E}_Z[\widehat{f_{J,Z}}(S)^2]] \\
&= \mathbb{E}_J\left[\sum_U \widehat{f}(U)^2\mathbb{1}_{\{U\cap J=S\}}\right] \\
&= \sum_U \widehat{f}(U)^2\mathbb{P}(U\cap J=S) \\
&= \sum_{U\supseteq S} \widehat{f}(U)^2 p^{|S|}(1-p)^{|U\setminus S|}.
\end{aligned}$$

Lemma 1.1.

$$\mathbb{E}_{(J,Z)\sim\mathcal{R}_p}[\mathbb{I}(f_{J,Z})] = p \cdot \mathbb{I}(f).$$

We will prove this next time. This will allow us to analyze the total influence of a function by first restricting some of its coordinates at random and then analyzing the total influence of the simpler function.