CAS CS 599 B: Mathematical Methods for TCS

Lecturer: Mark Bun Spring 2022

Lecture Notes 4:

Spectral Concentration and Learning

Reading.

- O'Donnell, Analysis of Boolean Functions §3.1-3.2, 3.4

1 Spectral Concentration

Today we'll study another measure of the simplicity (or complexity) of Boolean functions based on whether they are approximated by low-degree polynomials.

Definition 1. The *degree* of a function $f : \{-1,1\}^n \to \mathbb{R}$ is the degree of its Fourier expansion as a multilinear polynomial, i.e., $\deg(f) = \max\{|S| \mid \hat{f}(S) \neq 0\}$.

Example 2.

1. The constant functions ± 1 have degree 0.
2. A dictator χ_i has degree 1.
3. A *k-junta* is a Boolean function $f(x_1, \dots, x_n)$ that depends only on some subset $I \subseteq [n]$ of variables with $|I| = k$. A k-junta has degree k.
4. In general, a low-degree function might still depend on all of its variables, e.g., $f(x) = x_1 + \cdots + x_n$. The general form of a degree-d function is
$$f(x) = \sum_{|S| \leq d} c_s \chi_S(x).$$
5. A *decision tree* T over variables $x_1, \dots, x_n$ is a binary tree where each internal node is labeled by a variable and each leaf and all internal edges are labeled by ± 1. It computes by traversing the root-to-leaf path consistent with the input and outputs the label of the leaf reached.

   If f is computed by a depth-d decision tree, then $\deg(f) \leq d$. To see this, for each leaf ℓ of the tree let b_ℓ be the label of that leaf. Let $\mathrm{cons}_\ell(x)$ be the indicator function for whether the tree ends up at leaf ℓ on input x. Each function cons_ℓ is a d-junta, since consistency with the path to ℓ can be checked by inspecting at most d variables. Thus, $f(x) = \sum_{\text{leaves } \ell} b_\ell \cdot \mathrm{cons}_\ell(x)$ is a linear combination of d-juntas, hence a degree-d polynomial.

Low-degree functions are nice because they have compressed representations. A general Boolean function f requires 2^n parameters to specify. But if you know f has degree d, you only need to specify

$$\binom{n}{\leq d} := \sum_{k=0}^{d} \binom{n}{k} \leq O(n^d)$$

n-bit numbers for its Fourier coefficients. This is useful in learning applications, since to learn an unknown low-degree polynomial, you just need to learn a relatively small number of coefficients.

Unfortunately, many simple functions are not literally low degree. For example,

$$\mathrm{AND}_n(x) = (1 - 2^{-n}) + \sum_{S \neq \emptyset} 2^{-n+1}(-1)^{|S|+1}\chi_S(x)$$

has degree n, but with respect to the uniform distribution, it is basically the constant all-false function $\mathbf{1}$. This is related to the fact that the sum of the squared Fourier coefficients of AND_n above level 0 is exponentially small.

You had an exercise where you explored the connection between approximability by degree-d polynomials and low Fourier weight above level k. To spell this out a bit more, we make a few definitions.

Definition 3. For functions $f, g : \{-1,1\}^n \to \mathbb{R}$, define their squared ℓ_2 distance by

$$\|f - g\|_2^2 = \mathop{\mathbb{E}}_{x \sim \{-1,1\}^n}\left[(f(x) - g(x))^2\right] = \langle f - g, f - g\rangle.$$

For $\varepsilon > 0$, a function g is said to ε-approximate f in squared ℓ_2 distance if $\|f - g\|_2^2 \leq \varepsilon$.

Definition 4. The Fourier weight of a function f at level d is

$$\mathbf{W}^{d}[f] = \sum_{|S|=d} \hat{f}(S)^2.$$

The Fourier weight of a function f above level d is

$$\mathbf{W}^{>d}[f] = \sum_{|S|>d} \hat{f}(S)^2.$$

The following theorem tightly connects low-degree approximation to concentration of the Fourier spectrum.

Theorem 5. *A real function $f : \{-1,1\}^n \to \mathbb{R}$ is ε-approximated in squared ℓ_2 distance by a degree-d polynomial p if and only if $\mathbf{W}^{>d}[f] \leq \varepsilon$.*

Proof. Let $p(x) = \sum_{|S| \leq d} \hat{p}(S)\chi_S(x)$ be an arbitrary degree-d polynomial. By Parseval,

$$\|f - p\|_2^2 = \sum_{S \subseteq [n]} (\hat{f}(S) - \hat{p}(S))^2 = \sum_{S \leq d} (\hat{f}(S) - \hat{p}(S))^2 + \sum_{S > d} \hat{f}(S)^2.$$

This is uniquely minimized by taking $\hat{p}(S) = \hat{f}(S)$ for all $|S| \leq d$, i.e., by setting $p(x) = f^{\leq d}(x)$ to be the part of f at or below level d. The error of this minimizer is exactly $\mathbf{W}^{>d}[f]$. □

This proof has a nice linear algebraic interpretation. Recall that the parity functions form an orthonormal basis for the vector space of all real-valued functions $f : \{-1,1\}^n \to \mathbb{R}$. The parities of width at most k span a subspace. The best ℓ_2 approximation p of degree k to a function f is its projection onto this subspace, which is obtained by throwing away the component in the orthogonal subspace. [Draw a picture.]

This connection motivates the following definition of spectral concentration.

Definition 6. A function f is ε-concentrated on degree up to d if

$$\mathbf{W}^{>d}[f] \leq \varepsilon.$$

2 Bounding Fourier Weight

One can of course estimate the Fourier weights of a given function f using its Fourier spectrum. This can be painful in cases where we don't have a grip on the specific Fourier coefficients. It turns out that we can exploit some simple relationships between spectral concentration, influence, and noise stability to estimate Fourier weights more combinatorially.

Proposition 7. *Any function $f : \{-1,1\}^n \to \mathbb{R}$ is ε-concentrated on degree up to $\mathbf{I}[f]/\varepsilon$.*

Proof. By Markov's inequality,

$$\mathbf{W}^{>d}[f] = \Pr_{S \sim \mathcal{S}_f}[|S| > d] \leq \frac{1}{d} \cdot \mathop{\mathbb{E}}_{S \sim \mathcal{S}_d}[|S|] = \frac{\mathbf{I}[f]}{d}.$$

To make this at most ε, it suffices to take $d = \mathbf{I}[f]/\varepsilon$. □

Example 8. For any monotone function f, we have $\mathbf{I}[f] = O(\sqrt{n})$. So monotone functions are ε-concentrated on degree up to $O(\sqrt{n}/\varepsilon)$. In particular, this is true for MAJ_n, but it turns out we can do better.

Next, we'll show that functions which are noise stable are also spectrally concentrated. It'll be more convenient to state the result in terms of noise sensitivity, which measures how much a function is likely to change under small random perturbations. (As opposed to noise stability, which measures how much a function is likely to stay the same on well-correlated inputs.) It's like how in some situations, it's more useful to work with distances, while in others with correlations, even though they capture the same concept.

Definition 9. Let $\delta \in [0,1]$. Consider the distribution on pairs (x,y) where x is uniformly random and for each i independently,

$$y_i = \begin{cases} x_i & \text{with probability } 1-\delta \\ -x_i & \text{with probability } \delta. \end{cases}$$

The *noise sensitivity* of a function $f : \{-1,1\}^n \to \{-1,1\}$ at δ is

$$\mathbf{NS}_\delta[f] = \Pr_{(x,y)}[f(x) \neq f(y)] = \frac{1}{2} - \frac{1}{2}\mathbf{Stab}_{1-2\delta}[f].$$

Proposition 10. *Any function f is ε-concentrated on degree up to $1/\delta$ for $\varepsilon = 3\mathbf{NS}_\delta[f]$.*

Proof. Using the Fourier formula for noise stability,

$$\begin{aligned}
2\mathbf{NS}_\delta[f] &= 1 - \mathbf{Stab}_{1-2\delta}[f] \\
&= \mathop{\mathbb{E}}_{S\sim\mathcal{S}_f}\left[1-(1-2\delta)^{|S|}\right] \\
&\geq (1-(1-2\delta)^d) \Pr_{S\sim\mathcal{S}_f}[1-(1-2\delta)^{|S|} > (1-(1-2\delta)^d)] \\
&= (1-(1-2\delta)^d) \Pr_{S\sim\mathcal{S}_f}[|S| > d].
\end{aligned}$$

The first inequality is Markov and the last equality follows from the fact that $(1-(1-2\delta))^x$ is an increasing function in x.

Taking $d = 1/\delta$ and using the fact that $(1-x)^{1/x} \leq 1/e$, we get

$$\Pr_{S\sim\mathcal{S}_f}[|S| > 1/\delta] \leq \frac{2\mathbf{NS}_\delta[f]}{1-(1-2\delta)^{1/\delta}} \leq \frac{2\mathbf{NS}_\delta[f]}{1-e^{-2}} \leq 3\mathbf{NS}_\delta[f].$$

□

Example 11. For every ρ, n,

$$\mathbf{Stab}_\rho[\mathrm{MAJ}_n] \geq \frac{2}{\pi}\arcsin\rho.$$

This implies that

$$\mathbf{NS}_\delta[\mathrm{MAJ}_n] \leq \frac{1}{\pi}\arccos(1-2\delta) = \frac{2}{\pi}\sqrt{\delta} + O(\delta^{3/2})$$

via the Taylor approximation to $\cos$. Hence $\mathbf{NS}_\delta[\mathrm{MAJ}_n] = O(\sqrt{\delta})$. So in fact, MAJ_n is ε-concentrated up to degree $O(1/\varepsilon^2)$.

3 Learning under the Uniform Distribution

A classic use of low-degree approximations is to the problem of binary classification. Here is the setting. You are given a *sample* $Z = ((x^1, f(x^1)), \ldots, (x^m, f(x^m)))$ where $x^1, \ldots, x^m \in \{-1,1\}^n$ are uniform and independent, and f is an unknown function. However, you are promised that $f \in \mathcal{C}$ for some *concept class* $\mathcal{C} \subseteq \{f : \{-1,1\}^n \to \{-1,1\}\}$. Think of $\mathcal{C}$ as being "low-depth decision trees" or "small Boolean circuits."

Using these random samples, your goal is to learn an approximation to the unknown function f. Specifically, your job is to identify a *hypothesis* $h : \{-1,1\}^n \to \{-1,1\}$ such that $\mathrm{dist}(f,h) \leq \varepsilon$. We say that $\mathcal{C}$ is (ε,δ)-learnable under the uniform distribution if there exists a learning algorithm such that for all $f \in \mathcal{C}$,

$$\Pr_Z[\mathrm{dist}(f,h) \leq \varepsilon] \geq 1-\delta.$$

Theorem 12. *Let $\mathcal{C}$ be any concept class such that every $f \in \mathcal{C}$ is ε-concentrated up to degree d. Then $\mathcal{C}$ is $(2\varepsilon,\delta)$-learnable under the uniform distribution in time* $\mathrm{poly}(n^d, 1/\varepsilon, \log(1/\delta))$.

Example 13. Here are some applications of spectral concentration bounds to learning:

1. Depth-d decision trees are learnable in time $n^{O(d)}$.

2. Monotone functions are learnable in time $n^{O(\sqrt{n})}$.

3. Any $\{\wedge, \vee, \neg\}$ circuit of size s and depth k is ε-concentrated up to degree $O(\log^{k-1}(s)\log(1/\varepsilon))$. (This is Tal's tightening of LMN's famous result re: learning AC^0 in quasipolynomial time.) Hence, size-s depth-k circuits are learnable in time $n^{O(\log^{k-1}(s)\log(1/\varepsilon))}$.

As alluded to before, the idea will be to learn all of the low-degree Fourier coefficients of f. This is enabled by the following subroutine that allows us to estimate a single Fourier coefficient using random samples labeled by f.

Lemma 14. *There exists a randomized algorithm Est that, given a set $S \subseteq [n]$ and random examples of the form $(x, f(x))$, outputs c_S such that $|c_S - \hat{f}(S)| \leq \varepsilon$ with probability at least $1 - \delta$ in time $poly(n, 1/\varepsilon, \log(1/\delta))$.*

Proof sketch. Recall that $\hat{f}(S) = \mathbb{E}_x[f(x)\chi_S(x)]$. A Chernoff bound shows that setting c_S to be the empirical average of $m = O(\log(1/\delta)/\varepsilon^2)$ random evaluations of $f(x^i)\chi_S(x^i)$ gives an ε-approximation with probability $1 - \delta$. I'll ask you to work out the details in an exercise. □

We are now ready to state Linial, Mansour, and Nisan's learning algorithm for spectrally concentrated classes.

LMN "Low Degree" Algorithm Given uniform random samples labeled by a function $f : \{-1,1\}^n \to \{-1,1\}$:

1. For every set $S \subseteq [n]$ of size at most d, use subroutine $\text{Est}(S)$ to compute an estimate c_S of $\hat{f}(S)$ to accuracy $\varepsilon' = \sqrt{\varepsilon/2n^d}$ and failure probability $\delta' = \delta/2n^d$.

2. Output $h(x) := \text{sgn}\left(\sum_{|S|\leq d} c_S\chi_S(x)\right)$

Proof of Theorem 12. By a union bound, with probability at least $1 - \delta$ we have $|c_s - \hat{f}(S)| \leq \varepsilon'$ for every S. Assuming this is the case, and using the fact that $h(x) \neq f(x) \implies (p(x) - f(x))^2 \geq 1$,

$$\begin{aligned}
\text{dist}(h, f) &\leq \mathop{\mathbb{E}}_{x\sim\{-1,1\}^n}\left[(p(x) - f(x))^2\right] \\
&= \sum_{|S|\leq d}(c_s - \hat{f}(S))^2 + \sum_{|S|>d}\hat{f}(S)^2 \\
&\leq \binom{n}{d}(\varepsilon')^2 + \varepsilon \leq 2\varepsilon.
\end{aligned}$$

□