CAS CS 599 B: Mathematical Methods for TCS

Lecturer: Mark Bun | Spring 2022

Lecture Notes 11:

Polynomial approximations, intro to extractors

Reading.

- Braverman, "Poly-logarithmic independence fools AC^0."
- Vadhan, Section 6.1

Last time, we proved Braverman's Theorem, which says that polylogarithmic independence fools circuits of constant depth and polynomial size. The way we proved it was by

1. Reducing the problem to one about constructing sandwiching polynomials, which in turn reduces to a question about constructing polynomial approximations with one-sided error.
2. Constructing these polynomials by combining "probabilistic" approximating polynomials with LMN-style ℓ_2 approximating polynomials.

Here is some intuition for how to construct these intermediate polynomials.

1 Constructing probabilistic polynomials

We'll start with the probabilistic polynomials:

Lemma 1. *For every size-s, depth-d circuit f and every distribution $\mathcal{D}$ over $\{0,1\}^n$, there exists a polynomial p with $\Pr_{x\sim\mathcal{D}}[p(x) \neq f(x)] \leq \varepsilon$ such that*

1. $\deg p \leq (\log s)^{O(d)} \cdot \log(1/\varepsilon)$
2. $\max_{x\in\{0,1\}^n}\{|p(x)|\} \leq \exp((\log s)^{O(d)} \cdot \log(1/\varepsilon))$
3. *There exists a size-*$\mathrm{poly}(s)$*, depth-*$O(d)$ *circuit* E *such that* $p(x) \neq f(x) \iff E(x) = 1$.

This statement is due to Harsha and Srinivasan and you can see their paper for the proof. Let's see why the statement is true when f is a single OR gate with fan-in n.

Lemma 2. *For every distribution $\mathcal{D}$ over $\{0,1\}^n$, there exists a polynomial p with $\Pr_{x\sim\mathcal{D}}[p(x) \neq \mathrm{OR}_n(x)] \leq \varepsilon$ such that*

1. $\deg p \leq O(\log n \cdot \log(1/\varepsilon))$
2. $\max_{x\in\{0,1\}^n}\{|p(x)|\} \leq \exp(O(\log^2 n \cdot \log(1/\varepsilon)))$
3. *There exists a size-*$\mathrm{poly}(n)$*, depth-*$O(1)$ *circuit* E *such that* $p(x) \neq \mathrm{OR}_n(x) \iff E(x) = 1$.

Proof. We will actually construct a distribution $\mathcal{P}$ over polynomials satisfying conditions 2 and 3 such that for every $x \in \{0,1\}^n$, we have $\Pr_{p \sim \mathcal{P}}[p(x) \neq \mathrm{OR}_n(x)] \leq \varepsilon$. This suffices because

$$\mathop{\mathbb{E}}_{p \sim \mathcal{P}} \left[\mathop{\mathbb{E}}_{x \sim D} \left[\mathbf{1}_{p(x) \neq \mathrm{OR}_n(x)} \right] \right] = \mathop{\mathbb{E}}_{x \sim D} \left[\mathop{\mathbb{E}}_{p \sim \mathcal{P}} \left[\mathbf{1}_{p(x) \neq \mathrm{OR}_n(x)} \right] \right] \leq \varepsilon$$

implies that there exists a p in the support of $\mathcal{P}$ such that $\Pr_{x \sim D}[p(x) \neq \mathrm{OR}_n(x)] \leq \varepsilon$.

We now describe how to sample a polynomial p from $\mathcal{P}$ using the following idea. Suppose $x \in \{0,1\}^n$ is an input which is promised to have either Hamming weight 0 (causing OR_n to evaluate to 0) or Hamming weight $|x| \in [2^j, 2^{j+1})$ (causing it to evaluate to 1). Then a certain random linear function is able to distinguish between these two cases as follows. Let $S_j \subseteq [n]$ where each variable x_i is included in S_j with probability 2^{-j-1}. Let $p_j(x) = \sum_{i \in S_j} x_i$. Then:

- If $|x| = 0$, we have $p_j(x) = 0$ with probability 1.
- If $2^j \leq |x| < 2^{j+1}$, then $p_j(x) = 1$ with probability at least $\frac{1}{6}$.

We can now handle an arbitrary input x by combining polynomials $p_1, \ldots, p_{\log n}$ as constructed above:

$$p(x) = 1 - \prod_{j=1}^{\log n} (1 - p_j(x)).$$

This polynomial is always correct when $|x| = 0$, but only succeeds with probability $1/6$ otherwise. Taking $O(\log(1/\varepsilon))$ repetitions of each inner polynomial p_j lets us boost the success probability on true inputs to $1 - \varepsilon$, for a final degree bound of $O(\log n \cdot \log(1/\varepsilon))$.

Condition 2 holds by noting that the maximum value any p_j can take is n.

We now need to check condition 3, that the error region for any p constructed in this way can be detected by a poly-size constant-depth circuit. On input $x \in \{0,1\}^n$, the polynomial p makes a mistake if and only if all the following are true:

1. $|x| \geq 1$. This can be checked using a single OR gate.
2. For all $O(\log n \cdot \log(1/\varepsilon))$ sets S_j constructed, S_j intersects the 1's of x either 0 or ≥ 2 times. This can be checked by taking an AND over all S_j's of an OR_2 over a NOR of the bits $(x_i)_{i \in S_j}$ and an OR of the pairwise ANDs $(x_i \wedge x_{i'})_{i \neq i' \in S_j}$.

Checking whether both conditions are satisfied can thus be done by a small circuit. □

A statement for general AC^0 circuits can be obtained by iteratively composing these probabilistic polynomials, taking the error parameter to be ε/s to permit union bounding over the gates in the circuit.

2 Uniform approximations

Recall the other flavor of polynomial approximations we've seen a few times:

Theorem 3 (Tal's strengthening of LMN). *Let f be computed by an $\{\wedge, \vee, \neg\}$ circuit of size s and depth d. Then there exists a polynomial $p : \{-1,1\}^n \to \mathbb{R}$ of degree $O(\log^{d-1}(s) \log(1/\varepsilon))$ such that $\mathbb{E}[(f-p)^2] \leq \varepsilon$.*

Today, we'll gain some intuition for why the following special case is true. Recall that a DNF is an OR of ANDs of literals (variables or their negations). The width of a DNF is the maximum number of literals appearing in any AND term.

Lemma 4. *If f is computed by a DNF of width w, then $\mathbf{W}^k[f] \leq \varepsilon$ for some $k = O(w \log(1/\varepsilon))$*

Using the fact that spectral concentration characterizes the minimum error of an ℓ_2 approximation (Lecture 4, Theorem 5), this tells us that low-width DNF are well-approximated by polynomials in the sense of Theorem 3.

The key technical ingredient we need is *random restrictions*. A restriction ρ is just a string $\rho \in \{-1, 1, \star\}^n$. The restriction $f|_\rho$ is the function one obtains by fixing $x_i = \rho_i$ whenever $\rho_i \in \{-1, 1\}$ and leaving x_i alone if $\rho_i = \star$. A δ-random restriction ρ is one where each coordinate is set to $\star$ with probability δ and fixed to either ± 1 with probability $(1-\delta)/2$ each.

Random restrictions play nicely with the Fourier expansion:

Proposition 5 (Proposition 4.17 in O'Donnell)**.** *Let ρ be a δ-random restriction. Then*

$$\mathbb{E}[\widehat{f|_\rho}(S)^2] = \sum_{U \subseteq [n]} \hat{f}(U)^2 \cdot \Pr[U \cap \mathrm{stars}(\rho) = S].$$

The other critical property of random restrictions is that they simplify low-width DNF into decision trees.

Theorem 6 (Håstad's Switching Lemma)**.** *Let f be computed by a DNF of width w, and let ρ be a δ-random restriction. Then for any $k \geq 1$,*

$$\Pr[\text{DT-depth}(f|_\rho) \geq k] \leq (5\delta w)^k.$$

Proving the Switching Lemma would be worthy of its own lecture, but you can find plenty of good expositions online. We'll use it by taking $\delta \approx 1/10w$. For these parameters, the Switching Lemma says that by restricting enough variables that we expect less than 1 to stay unrestricted in each term, the probability that the DNF fails to simplify to a shallow decision tree falls exponentially in k.

Proof of Lemma 4. For some $d = O(\log(1/\varepsilon))$ and $\delta = 1/(10w)$, let ρ be a δ-random restriction. By the Switching Lemma, we have that

$$\mathbb{E}\left[\sum_{|S|>d} \widehat{f|_\rho}(S)^2\right] \leq \Pr[\text{DT-depth}(f|_\rho) \geq d] \leq (5\delta w)^d \leq \frac{\varepsilon}{2}.$$

On the other hand, by Proposition 5,

$$\mathbb{E}\left[\sum_{|S|>d} \widehat{f|_\rho}(S)^2\right] = \sum_{U \subseteq [n]} \hat{f}(U)^2 \cdot \Pr[U \cap \mathrm{stars}(\rho) > d].$$

The random variable $|U \cap \mathrm{stars}(\rho)| \sim \mathrm{Bin}(|U|, \delta)$. When $|U| > d/\delta$, we have $\Pr[U \cap \mathrm{stars}(\rho) > d] \geq 1/2$, since the median of a binomial is its mean. Thus,

$$\mathbb{E}\left[\sum_{|S|>d} \widehat{f|_\rho}(S)^2\right] \geq \sum_{|U|>d/\delta} \hat{f}(U)^2 \cdot \frac{1}{2}.$$

Putting the inequalities together and setting $k = d/\delta$ gives the statement. □

3 Randomness Extractors

We're going to switch gears now to talking about a different kind of pseudorandom object: randomness extractors. A randomness extractor is a function that takes a sample from a "weak" source of randomness, e.g., biased and correlated bits, and outputs a string of nearly uniform bits. The original motivation was to implement randomized algorithms in a world where physical sources of randomness (atmospheric noise, temperature readings, the lower order bits of the system clock) do not immediately give us uniform random bits. Since then, randomness extractors have found numerous applications in algorithms, cryptography, and complexity. Ironically, despite begin in some sense the "opposite" of pseudorandomn generators, some of the most important constructions of PRGs are based on extractors and vice versa.

Von Neumann performed an early study of the randomness extraction problem. He considered random sources of the form $X_1, X_2, \ldots, X_n \in \{0,1\}$ that are i.i.d., but biased, i.e. $\Pr[X_i = 1] = \delta$ for some $\delta \in (0,1)$. He described the following extractor: Break the source of bits into pairs (X_{2i-1}, X_{2i}). If the pair comes up $(0,1)$, output 0. If it comes up $(1,0)$, output 1. If it comes up either $(0,0)$ or $(1,1)$, output nothing. Each output bit comes up 0 or 1 with probability $\delta(1-\delta)$ each; this extractor produces $n\delta(1-\delta)$ uniform bits in expectation. See Peres, "Iterating von Neumann's procedure for extracting random bits" for an asymptotically optimal extractor for these sources.

The sources von Neumann considered are still much more structured than we imagine physical sources of randomness to be. We'd like to be able to extract uniform bits from much more general classes of sources. A "source" is just a distribution over $\{0,1\}^n$.

Definition 7. A deterministic extractor for a class $\mathcal{C}$ of sources over $\{0,1\}^n$ is a function $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}^m$ if $TV(\mathrm{Ext}(X), \mathcal{U}_m) \leq \varepsilon$ for every $X \in \mathcal{C}$.

What is the most general class of sources from which we can hope to extract m uniform bits? Intuitively, if we want to extract m bits, the source must have "contained m bits of randomness" to start with, which suggests a lower bound on its entropy, or information content. In fact, we can observe that if $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}^m$ is a 0-extractor for X, then for every $x \in \mathrm{supp}(X)$, we must have $\Pr[X = x] \leq 2^{-m}$. Otherwise, if there were an x with $\Pr[X = x] > 2^{-m}$, we would have $\Pr[\mathrm{Ext}(X) = \mathrm{Ext}(x)] > 2^{-m}$ which is impossible under the uniform distribution on m bits. This motivates the following definition of entropy, or information content, of a distribution.

Definition 8. The min-entropy of a source X is

$$H_\infty(X) = \min_{x \in \mathrm{supp}(X)} \log \frac{1}{\Pr[X = x]}.$$

The observation above says that if we want to 0-(deterministically) extract m bits from a source X, then it must have min-entropy at least m.

Definition 9. An (n,k) source is a random variable X on $\{0,1\}^n$ such that $H_\infty(X) \geq k$. Equivalently, $\Pr[X = x] \leq 2^{-k}$ for all $x \in \{0,1\}^n$.

We'd like to be able to construct extractors for the class of all (n,k) sources. Unfortunately, this still turns out to be impossible using our definition of deterministic extractors: