18.218 — Analysis of Boolean Functions

Class by Dor Minzer

Notes by Sanjana Das

Spring 2024

Notes for the MIT class **18.218** (Analysis of Boolean Functions), taught by Dor Minzer. All errors are my responsibility.

Contents

§1 February 6, 2024

§1.1 Introduction

The topic of this course is analysis of Boolean functions. In this course, we'll mainly talk about Boolean functions on the hypercube — functions $f\colon \{0,1\}^n \to \{0,1\}$. (We refer to $\{0,1\}^n$ as the *hypercube*.) We can think of the hypercube as a probability space; for the majority of the course we'll consider the uniform distribution. But many applications require domains that are different. So we'll mainly work with the cube to be concrete, but there are extensions to other domains. For example, instead of working with $\{0,1\}^n$, you can work with more general product spaces — i.e., a space of the form $\Omega_1 \times \cdots \times \Omega_n$ with the product measure. (The Boolean cube is a specific example of a product space, where we take each Ω_i to be $\{0,1\}$ with the uniform measure.) There's also *product-line spaces* (as a few buzzwords — the Johnson graph, the symmetric group, the Grassman graph, high-dimensional expanders). Much of the theory we'll see works for these domains. (Much of this is fairly recent, and we may or may not see it in this course.)

In these more complicated domains, you start getting connections to other areas of math — for example, for S_n you start needing algebra and representation theory.

§1.2 Course overview

If there's something specific we would like to see, we should tell Dor. Otherwise, here's a tentative outline of what we'll see.

We'll start with the very fundamentals of this area — fundamental ideas in discrete Fourier analysis. This will be a few lectures (we'll start today by giving the definition of the discrete Fourier transform). This includes definitions, applications to property testing, learning, and so on. We'll then see some heavier tools, like hypercontractivity (which is a scary-looking word, but is really a basic result that many further results will use) and its applications. This will be our basic toolbox.

Then we'll see applications in hardness of approximation. Dor is originally from the areas of PCPs and hardness of approximation. Often, the questions of interest in analysis come up naturally when you try to prove some result from this area — e.g. you try to do a NP-hardness reduction, and you suddenly need a tool from analysis. We'll see some of these applications.

Then we'll see applications in extremal combinatorics. Here's an example of what a typical problem in extremal combinatorics looks like.

Example 1.1

Suppose we have a collection of subsets $\mathcal{F} \subseteq \binom{[n]}{n/2}$ which is *intersecting*, meaning that for all $A, B \in \mathcal{F}$ we have $A \cap B \neq \emptyset$. How large can $\mathcal{F}$ be?

This is a classical problem, called the Erdős–Ko–Rado problem, and you don't need analysis to solve it; but for more complicated questions things become more complicated, and there's a lot you can do using analysis.

Remark 1.2. As a construction for the above problem, we can take $\mathcal{F}$ to consist of all subsets containing a particular element — for example, $\mathcal{F} = \{A \subseteq [n] \mid |A| = \frac{n}{2}, 1 \in A\}$. It turns out that this is tight (i.e., the best possible construction). For reasons we'll see later on, this family is called a *dictatorship* — because to see whether a set is inside $\mathcal{F}$ or not, we only need to look at one element.

Finally, and most vaguely, we'll see advanced topics. This area has seen quite interesting developments in the last few years, and we may see some of this. Some buzzwords include the sensitivity conjecture, analysis over other domains (as mentioned previously), *global hypercontractivity*, and so on. We'll see some of this; if we have specific requests we should ask Dor.

§1.3 Functions on the Boolean cube

We're looking at the Boolean cube $\{0,1\}^n$, but in some sense the right way to look at this is as an abelian group — with addition mod 2. Modular arithmetic is not very convenient to write, so the first thing we'll do is change notations so that instead of talking about mods, we'll talk about real numbers (and in general complex numbers).

So instead of using $\{0,1\}$ and addition mod 2, we'll look at $\{1,-1\}$ with multiplication (via the map $b \mapsto (-1)^b$). We can check that addition mod 2 on the left corresponds to multiplication on the right. This will be much more convenient for us notationally.

Definition 1.3. A *Boolean function* is a map $f\colon \{-1,1\}^n \to \{-1,1\}$.

More generally, we can consider all maps $f\colon \{-1,1\}^n \to \mathbb{R}$. We only really care about Boolean functions, but we need mathematical tools that work better when we work with $\mathbb{R}$ — we'll use linear algebra, and we can't do much with $\{-1,1\}$ using linear algebra. But once we move to $\mathbb{R}$, we can say that the collection of all these functions is a *vector space* over the reals.

Vector spaces are nice — you can choose bases. But sometimes, when you have more structure on your vector space — in particular, an *inner product* — it becomes even nicer.

Definition 1.4. For two functions $f, g\colon \{-1,1\}^n \to \mathbb{R}$, we define $\langle f, g\rangle = \mathbb{E}_{x\sim\{-1,1\}^n}[f(x)\cdot g(x)]$.

(Whenever we write expectations, it's with respect to the uniform measure on $\{-1,1\}^n$.)

It's easy to check that $\langle -,-\rangle$ is indeed an inner product. Notationally, it's convenient to think of this as $L_2(\{-1,1\}^n)$ (where the association is that we think of the cube with the uniform measure and this inner product).

§1.4 The Fourier basis

Next, we'd like to choose a basis for this vector space. First, here's a naive example of a basis:

Example 1.5

For each $x \in \{-1,1\}^n$, we can consider the indicator function $1_x\colon \{-1,1\}^n \to \mathbb{R}$, defined as

$$1_x(y) = \begin{cases} 1 & \text{if } y = x \\ 0 & \text{otherwise.} \end{cases}$$

The functions 1_x form a basis, and this basis is even orthogonal. But it doesn't buy us much mileage.

There's many bases — we can take any collection of 2^n linearly independent functions, apply the Gram–Schmidt process, and get a basis. But most won't be very useful.

Here's a basis that's much more useful. It might seem like magic, but it also comes from group theory.

Definition 1.6. For each subset of coordinates $S \subseteq [n]$, we define a function $\chi_S\colon \{-1,1\}^n \to \{-1,1\}$ as

$$\chi_S(x) = \prod_{i\in S} x_i.$$

In other words, χ_S is defined on x by taking the product of all the bits of x with indices in S.

> **Remark 1.7.** If we know group theory, these are the characters of $\mathbb{F}_2^n$ — so these functions arise naturally in group theory.

We have 2^n functions here, so at least we have the right number. But unlike the first example, it's not clear that this is a basis. We're now going to prove that.

> **Lemma 1.8**
> The collection $\{\chi_S \mid S \subseteq [n]\}$ an orthonormal basis of $L_2(\{-1,1\}^n)$.

Proof. First suppose that we have two subsets $S, T \subseteq [n]$. For each we have a character; what happens when we multiply these characters? We want to calculate $\chi_S(x) \cdot \chi_T(x)$; by opening up the definition, we have

$$\chi_S(x) \cdot \chi_T(x) = \prod_{i \in S} x_i \cdot \prod_{i \in T} x_i.$$

We can rewrite this as

$$\chi_S(x) \cdot \chi_T(x) = \prod_{i \in S \cap T} x_i^2 \cdot \prod_{i \in S \Delta T} x_i$$

(where $S \,\Delta\, T$ denotes the *symmetric difference* of S and T — the set of elements that appear in exactly one of S and T). Now, the first product is simply 1 (as x_i^2 is always 1); this means we simply get

$$\chi_S(x) \cdot \chi_T(x) = \prod_{i \in S \Delta T} x_i.$$

So we took two characters, multiplied them, and got something that looks like a character; and in fact, this really *is* a character — we have

$$\chi_S(x) \cdot \chi_T(x) = \chi_{S \Delta T}(x).$$

So to summarize, if we multiply two characters, then we get another character. (This is a very nice property that e.g. doesn't hold for our first example of a basis, and it's one reason why this basis is nice.)

We now want to prove that a basis is orthogonal. This means we want to take an *expectation* of $\chi_S(x) \cdot \chi_T(x)$. We know that this product is some other character, so we now want to study the expectation of a character.

Suppose we fix some $S \subseteq [n]$; then what is $\mathbb{E}_x[\chi_S(x)]$? If $S = \emptyset$ then $\chi_S(x)$ is just an empty product, which is 1; this means

$$\mathbb{E}_x[\chi_S(x)] = 1.$$

Meanwhile, if S is *not* empty, then we have

$$\mathbb{E}_x[\chi_S(x)] = \mathbb{E}[\prod_{i \in S} x_i] = \prod_{i \in S} \mathbb{E}[x_i] = 0$$

(since the bits of x are independent, and $\mathbb{E}[x_i] = 0$ for each i).

Now with these two observations, we're basically done — we have

$$\langle \chi_S, \chi_T \rangle = \mathbb{E}[\chi_S(x)\chi_T(x)] = \mathbb{E}[\chi_{S\Delta T}(x)] = \begin{cases} 1 & \text{if } S = T \\ 0 & \text{otherwise} \end{cases}$$

(as $S \,\Delta\, T$ is empty if and only if $S = T$). This proves that our collection is orthonormal.

Finally, to see that it is a basis, we can count dimension — every orthonormal set is linearly independent, and we have 2^n functions in a vector space of dimension 2^n, so they do form a basis. □

§1.5 Fourier coefficients

If we have a basis, then we can take any vector in our vector space and express it according to our basis. So let's do that — since $\{\chi_S\}$ is a basis, for every $f\colon \{-1,1\}^n \to \mathbb{R}$, we can express f as a linear combination of these basis elements, i.e,.

$$f(x) = \sum_{S\subseteq[n]} \widehat{f}(S)\chi_S(x)$$

for some coefficients $\widehat{f}(S)$.

Definition 1.9. We call the coefficients $\widehat{f}(S)$ the *Fourier coefficients* of f.

This is true for *any* basis, but because we have an *orthonormal* basis, there's a very nice formula for these coefficients — we have

$$\widehat{f}(S) = \langle f, \chi_S\rangle$$

for each S. So we have a very nice formula for the Fourier coefficients.

We'll now state several basic facts.

Lemma 1.10

Let $f, g\colon \{-1,1\}^n \to \mathbb{R}$ be two functions.

(1) We have $\langle f, g\rangle = \sum_S \widehat{f}(s)\cdot\widehat{g}(s)$. (This is called the Plancherel equality.)

(2) We have $\lVert f\rVert_2^2 = \sum_S \widehat{f}(S)^2$. (This is called Parseval's equality.)

This means if we want to compute an inner product, we can do it just by looking at Fourier coefficients. Note that (2) is a special case of (1), because $\lVert f\rVert_2^2 = \langle f, f\rangle$. (The fact that there's two different names is for historical reasons.)

Proof. We'll simply plug in the Fourier expansions of f and g and see where this takes us — we have

$$\langle f, g\rangle = \mathbb{E}_x[f(x)g(x)] = \mathbb{E}_x\left[\left(\sum_S \widehat{f}(S)\chi_S(x)\right)\left(\sum_T \widehat{g}(T)\chi_T(x)\right)\right].$$

Now we can simply multiply out these sums to get that

$$\langle f, g\rangle = \mathbb{E}_x\left[\sum_{S,T} \widehat{f}(S)\widehat{g}(T)\chi_S(x)\chi_T(x)\right].$$

Now we can note that $\chi_S(x)\chi_T(x) = \chi_{S\Delta T}(x)$. We have an expectation of a sum, and we can use linearity of expectation to push the expectation inside — so we get

$$\langle f, g\rangle = \sum_{S,T} \widehat{f}(S)\widehat{g}(T)\mathbb{E}_x[\chi_{S\Delta T}(x)].$$

And we already know this expectation is 1 if $S = T$ and 0 otherwise, so we get

$$\langle f, g\rangle = \sum_S \widehat{f}(S)\widehat{g}(S). \qquad \square$$

Often, when we have a function f on the hypercube, we'll think of $f(x)$ as a random variable — where we sample x uniformly from the hypercube. Then we can ask about quantities such as $\mathbb{E}[f(x)]$ and $\mathrm{Var}[f(x)]$; we'll now give Fourier analytic formulas for both of these.

Claim 1.11 — We have $\mathbb{E}[f(x)] = \widehat{f}(\emptyset)$.

Proof. We have $\mathbb{E}f(x) = \mathbb{E}_x[f(x)\chi_\emptyset(x)] = \langle f, \chi_\emptyset\rangle = \widehat{f}(\emptyset)$. □

Claim 1.12 — We have $\mathrm{Var}[f(x)] = \sum_{S\neq\emptyset} \widehat{f}(S)^2$.

Proof. First, using the definition of variance, we have

$$\mathrm{Var}[f(x)] = \mathbb{E}_x\left[(f(x) - \mathbb{E}f)^2\right].$$

There's two things we could do next. One is to open up the square and hope for the best; then we'll get $\mathbb{E}[f(x)^2] - \mathbb{E}[f(x)]^2$, and we've already studied $\mathbb{E}[f(x)^2]$. Alternatively, we can directly plug in the Fourier expansion of f inside here; we have

$$f(x) - \mathbb{E}f = \sum_{S\subseteq[n]} \widehat{f}(S)\chi_S(x) - \widehat{f}(\emptyset).$$

Since $\chi_\emptyset(x)$ is the constant function 1, the term $S = \emptyset$ cancels out; this means we get

$$\mathrm{Var}[f(x)] = \mathbb{E}_x\left(\sum_{S\neq\emptyset} \widehat{f}(S)\chi_S(x)\right)^2.$$

We can now think of this sum $\sum_S \widehat{f}(S)\chi_S(x)$ as a new function g (whose Fourier coefficients we already know), and we want to compute its 2-norm; and so by Parseval, we know that this is

$$\sum_{S\neq\emptyset} \widehat{f}(S)^2.$$

□

§1.6 Property testing

So far, we haven't done anything fancy; but this already has some very nice and nontrivial applications (which we'll discuss next lecture).

§1.6.1 Blum–Luby–Rubinfeld linearity testing

In property testing, we often have some property we want to test. For example, suppose we have a function, and we want to look at some values of the function and know whether it's monotone or far from monotone. (Here the property is being monotone.)

In this example, the property we're considering is *linearity*.

Definition 1.13. A function $f\colon \{-1,1\}^n \to \{-1,1\}$ is *linear* if for all $x, y \in \{-1,1\}^n$, we have $f(x\cdot y) = f(x)\cdot f(y)$.

For $x, y \in \{-1,1\}^n$, we use $x\cdot y$ to denote coordinatewise multiplication — i.e., $x\cdot y$ is the vector z where $z_i = x_i\cdot y_i$ for each i. (The fact that we have multiplication is just an artifact of the fact that we're using $\{-1,1\}$ notation; in $\mathbb{F}_2$ notation these operations both become sums mod 2.)

Question 1.14. What are the linear functions on $\{-1,1\}^n$?

Exercise 1.15. Prove that if f is linear, then there exists some $S \subseteq [n]$ such that $f = \chi_S$.

We can observe that any character satisfies linearity; and it turns out that they're the *only* functions that are linear.

So we've now defined linearity.

Question 1.16. Suppose we have some unknown function f, and we're allowed to call a few of its values; we want to determine whether f is linear or not. How do we do this?

In other words, we're given query access to f, and we want to test whether f is linear or not.

Right now, this question is not phrased very well; when we solve it we'll phrase it better. Distinguishing between something linear and something not linear is in general hopeless — if we take a linear function and corrupt it in one location, then it's no longer linear but we have no hope of detecting that. So we need to relax this a bit.

Question 1.17. What can we say about f if it satisfies $f(xy) = f(x)f(y)$ for 'many' $x, y \in \{-1, 1\}^n$?

What we mean by 'many' is that if we sample $x, y \in \{-1, 1\}$, this happens with probability a bit over $\frac{1}{2}$ (e.g., 0.51).

It turns out that using what we've just seen, we can prove some very nice structure about f in this case.

§1.6.2 Learning sparse functions

The second application we'll see (which requires a bit more tools, but nothing fancy) is the problem of learning sparse functions.

First, what does this mean? Suppose that we have some unknown function $f\colon \{-1, 1\}^n \to \{-1, 1\}$ which is *sparse* in the Fourier domain — i.e., the number of nonzero Fourier coefficients is at most some given value M.

Question 1.18. Can we learn f by querying only polynomially many of its values?

We won't exactly quantify what we mean by 'learn'; but informally, we make polynomially many queries to f, get its values, do some polynomial-time computations, and give a function g that's very close to f — maybe it's not exactly the same, but it should agree with f on almost all inputs. This is what we refer to as *learning.*

Remark 1.19. We can't detect whether a function is sparse. To see this, given any function f, we can add to it some function that's 0 almost all the time — e.g., $f + 1_x$. It turns out that all the Fourier coefficients of 1_x are nonzero, but they're tiny; so we have no chance of detecting this.

We'll see this application later on; but combining what we know right now, we know the characters are linear functions; so we're asking about functions which are sparse when you express them as linear combinations of linear functions.

Question 1.20. What happens when we try to learn functions f that are sparse in other forms?

As a concrete example (reverting back to $\mathbb{F}_2$ notation), we can consider polynomials $P\colon \mathbb{F}_2^n \to \mathbb{F}_2$. One example is $P(x) = x_1 + \cdots + x_n$, which is linear; another example is $P(x) = x_1x_2 + x_3x_4 + \cdots$, which is *quadratic.*

The functions f that we were previously discussing are functions $f\colon \mathbb{F}_2^n \to \mathbb{R}$ that can be expressed as $\sum_S a_s(-1)^{P_S}$ where the P_S are linear. But what if instead of linear functions, we allowed quadratic functions — for example, what if we considered

$$f(x) = a_1(-1)^{\sum x_{2i}x_{2i+1}} + a_2 \cdot (\text{some other quadratic}) + \cdots.$$

So we have a sparse sum of quadratic functions; can we learn this?

> **Remark 1.21.** Last time Dor gave the course, this was an open problem; and some students solved it afterwards. The solution is very nice; it builds on Gowers uniformity norms and so on.

§1.6.3 Junta testing

(We've already used the word *dictatorship* in this course; the reason for these political-sounding words is that this is related to social choice theory.)

What is a dictator? A dictator is when we have n voters, but only the opinion of one matters. More generally, a *junta* is where we have n voters, but only the opinions of e.g. 10 of them matters.

> **Definition 1.22.** A function $f\colon \{-1,1\}^n \to \mathbb{R}$ is called a *t-junta* if there exists $T \subseteq [n]$ of size $|T| = t$ and a function $g\colon \{-1,1\}^T \to \mathbb{R}$ such that $f(x) = g(x_T)$.

(We usually think of t as $O(1)$.)

In other words, we do a voting and everyone has their opinions; then we throw away everyone's opinions except those in T, and just look at the coordinates in T. (You can think about this in terms of elections, which is a negative way to view this. But you can also think about this positively — sometimes in life we'll have functions that are very complicated (e.g., the effects of your genes on some property). But sometimes this only depends on a few things, which can be simpler.)

> **Question 1.23.** Suppose we're given query access to a function $f\colon \{-1,1\}^n \to \{-1,1\}$.
>
> - Can we test whether f is a junta?
> - Suppose we do know that f is a junta. Can we learn it efficiently?

§1.6.4 Kalai's proof of Arrow's impossibility theorem

The fourth application is to social choice theory. Arrow's impossibility theorem is a very well-known result in economics, for which Arrow won a Nobel prize; it turns out it can be proven easily using Fourier analysis. Its statement is very interesting and surprising.

Suppose that we have elections between three candidates Alice, Bob, and Charlie. We have n voters; each voter gives a preference among each pair of the three candidates. We say a voter's preferences are *consistent* if the rankings are transitive (e.g., $A > B$, $B > C$, $A > C$) and *inconsistent* otherwise (e.g., $A > B$, $B > C$, $C > A$).

Each voter gives their preferences; so we get preferences $x_1, \ldots, x_n$. We'll assume that the people who vote are consistent.

Now we have n votes, and we need to choose a winner of the election — this means we need some aggregation function. So we apply a function f on our votes to get a ranking of A, B, and C. (Each voter has preferences; we take all these preferences, aggregate them using some f, and get a ranking of the candidates.)

Arrow's theorem tells us that if this aggregator function satisfies two logical-looking properties, then there are not too many options for f.

Theorem 1.24 (Arrow's impossibility theorem)

Assume that f satisfies the following properties (assuming that the votes x_1, ..., x_n are consistent):

(1) The output $f(x_1, \ldots, x_n)$ is also consistent.

(2) If everyone prefers Alice over Bob, then Alice is ranked higher than Bob in $f(x_1, \ldots, x_n)$.

Then there exists some $i \in [n]$ such that $f(x_1, \ldots, x_n) = x_i$.

These are both properties that we'd certainly like an aggregator function to have. But this theorem says that the only aggregator functions with these properties are dictatorships — where only one voter's preferences matter.

We'll see in a few lectures that using analysis we can prove this easily — in fact, we can prove a robust version (where we relax these conditions — for example, to only require the output to be consistent 99% of the time).

These are four examples of problems in distinct areas of math (or social choice theory) that we'll see; we'll actually see more, but this is just a beginning.

§2 February 8, 2023

§2.1 Review

Last time, we slightly misstated Arrow's impossibility theorem; here's the correct version (we'll discuss it later on in the course).

As before, we have n voters and 3 candidates — A, B, and C. Each one of the n voters has a preference between each pair of candidates — a preference between A and B, B and C, and A and C — such that their preferences are consistent.

We care about voting rules where to determine which one of A and B is preferred, the voting scheme should only need to look at the preferences between A and B among all voters — so we only aggregate the preferences between A and B to determine our final preference.

And the property we want is that whenever the voters are consistent, this scheme should give us something consistent.

Theorem 2.1 (Arrow's impossibility theorem)

Suppose that f as above outputs a consistent preference whenever each voter submits a consistent preference. Then if f is non-trivial (i.e., not fixed), then it is a dictatorship.

(Here we're applying the same function f to all three columns; people have also studied versions with robustness, or with different functions for the different columns.)

§2.2 Testing linearity

Last time, we ended by presenting a few problems we said we could already solve using the material we know; now we're going to actually study one of them, the linearity testing problem.

§2.2.1 Property testing

First, here's a specialized introduction to property testing (specialized to linearity testing).

Definition 2.2. A function $f\colon \{-1,1\}^n \to \{-1,1\}$ is called *linear* if $f(xy) = f(x)f(y)$ for all $x, y \in \{-1,1\}^n$ (where xy denotes the coordinate-wise product $(xy)_i = x_i y_i$).

Our goal is the following:

Question 2.3. Design a (randomized) query-efficient algorithm that, given oracle access to a function f, distinguishes between the following cases:

(1) f is linear.

(2) f is not linear.

First, let's go over some of the words here that may be unfamiliar.

- First, there's *oracle access*; this means we can choose whatever input we want and submit it, and we get back $f(x)$. So in simple terms, we can get the output of f on whatever input we want.
- What do we mean by *query-efficient*? We mean we want the algorithm to query f on as few inputs as possible (ideally, constantly many).

Now that we've seen these two words, there should be an objection — this is impossible. What happens if we take a proper linear function f, take one input, and change the value there? Then f is not linear, but how on earth can we distinguish between the original function and the corrupted one by only looking at e.g. 10 inputs?

So what we've learned is that we need the two things we're distinguishing between to be 'far' from each other — because if there's something in the 'no' case which is just a slight perturbation of the 'yes' case, then our task is impossible.

So we need to relax the question; now let's do this.

Question 2.4. Design an query-efficient algorithm that, given oracle access to a function f:

(1) Accepts if f is linear.

(2) Rejects if f is 'far' from all linear functions.

Now we've introduced another word 'far' which we need to define — so we need to define a distance measure on functions. Many distance measures turn out to be equivalent; here's the most convenient one.

Definition 2.5. For any two functions f and g, we define the distance between them as

$$\Delta(f, g) = \frac{1}{2^n} \#\{x \in \{-1,1\}^n \mid f(x) \neq g(x)\}.$$

In other words, the distance between f and g is the fraction of inputs on which they are different.

§2.2.2 The BLR linearity tester

Now we finally have a well-posed problem — we need to design a query-efficient algorithm that accepts f with probability 1 if f is linear; and if f is far from linear, then it should reject with some noticeable probability.

The most natural thing to do is check if our definition holds on two *random* inputs. It's a common theme in probability testing that the algorithms are very natural; and what's more complicated is the *analysis*. (In this case the analysis will be beautiful and not that hard, but in many cases it will be hard.)

Algorithm 2.6 (BLR) — Our linearity tester works as follows:

(1) Sample two inputs x and y uniformly at random.

(2) Compute $f(x)$, $f(y)$, and $f(xy)$. Accept if $f(xy) = f(x)f(y)$ and reject otherwise.

One thing is obvious about this test: if f is linear, the test accepts with probability 1 (by the definition of linear functions).

Historically, this tester was built in the 1980s (by Bloom–Luby–Rubenfield), who proved the following.

Theorem 2.7 (BLR)

If f passes the test with probability at least $1 - \varepsilon$, then f is $O(\varepsilon)$-close to some linear function.

Their proof was purely combinatorial (and didn't use Fourier analysis); it was pretty nice (kind of like Sudoku). But the reason from Dor's point of view that this became very interesting is that later on, an extension of this result was used in the proof of the PCP theorem. Developments in the analysis of Boolean functions were often motivated by other areas, and PCPs were one of them. This statement itself was enough to get *some* PCP theorems, but to get more fancy ones, we need to study this tester in the regime where the acceptance isn't close to 1, but maybe just slightly above the trivial bound.

First, what would be the trivial threshold? If we sample f purely at random, then each output is $+1$ or -1 uniformly at random. And then our test accepts with probability $\frac{1}{2}$ — if we fix x and y, then $f(xy)$ is a random variable independent of $f(x)$ and $f(y)$ (ignoring the cases where x or y is 0), so the test passes with probability $\frac{1}{2}$. This means a random function passes with probability very close to $\frac{1}{2}$ (and a random function is certainly not close to linear).

Question 2.8. If f beats random *slightly*, then can we say anything about it?

Theorem 2.9

If f passes the BLR test with probability at least $\frac{1}{2} + \delta$, then there exists a linear function χ_S such that

$$\mathbb{P}_x[f(x) = \chi_S(x)] \geq \frac{1}{2} + \delta.$$

Remark 2.10. If δ is close to $\frac{1}{2}$, then we're getting something close to 1; this recovers the BLR theorem.

§2.2.3 Convolution

We can actually prove this without introducing any new tools, but let's use this opportunity to define an operation which will be useful for us later.

Definition 2.11. Given $f, g \colon \{-1, 1\}^n \to \mathbb{R}$, we define their *convolution* $f * g \colon \{-1, 1\}^n \to \mathbb{R}$ as

$$(f * g)(x) = \mathbb{E}_{y \in \{-1,1\}^n}[f(y)g(y \cdot x)].$$

Remark 2.12. Technically, we need to write y^{-1} instead of y, but here since we're working over Booleans we can just use y instead. In general, what happens is we take all pairs which multiply to x.

The main property of convolutions, which is equivalent to the definition, is the following. Suppose we have f and g, and we want to know the Fourier coefficients of $f * g$. It turns out that these are just the products of the Fourier coefficients of f and g.

Lemma 2.13

For all $S \subseteq [n]$, we have $\widehat{f * g}(S) = \widehat{f}(S)\widehat{g}(S)$.

Proof. We'll just unpack the definitions — first, we have

$$\widehat{f * g}(S) = \mathbb{E}_x[(f * g)(x)\chi_S(x)] = \mathbb{E}_{x,y}[f(y)g(y \cdot x)\chi_S(x)].$$

Now we'll change variables, replacing yx with z (so our expectation is now over y and z). Then we need to get rid of x in $\chi_S(x)$; and we have $x = yz$, so $\chi_S(x) = \chi_S(yz) = \chi_S(y)\chi_S(z)$ (since characters are linear functions). So we get

$$\widehat{f * g}(S) = \mathbb{E}_{y,z}[f(y)g(z)\chi_S(y)\chi_S(z)] = \mathbb{E}_{y,z}[f(y)\chi_S(y) \cdot g(z)\chi_S(z)].$$

And since y and z are independent, we can split this expectation as a product to get

$$\widehat{f * g}(S) = \mathbb{E}_y[f(y)\chi_S(y)] \cdot \mathbb{E}_z[g(z)\chi_S(z)] = \widehat{f}(S)\widehat{g}(S).$$ □

§2.2.4 Proof of our theorem

With this claim, we can now prove our theorem. We'll actually prove something slightly different:

Theorem 2.14

Suppose that $f\colon \{-1,1\}^n \to \{-1,1\}$ passes the BLR test with with probability at least $\frac{1}{2} + \delta$. Then there exists $S \subseteq [n]$ such that $\widehat{f}(S) \geq 2\delta$.

In other words, if f passes the BLR test with probability better than random, then it has a large Fourier coefficient.

Proof. We know that $f(x \cdot y) = f(x)f(y)$ with probability at least $\frac{1}{2} + \delta$, but we don't know what to do with this probability — we only have things in terms of expectations. So the first thing to do is rephrase this property in a more analytic way — this is in some sense the most significant step of the proof.

To do this, we can look at $f(xy) \cdot f(x) \cdot f(y)$ — this is always either 1 or -1, and we accept if and only if it's 1. This means we have

$$\mathbb{E}_{x,y}[f(x)f(y)f(x \cdot y)] = \mathbb{P}[\text{test accepts}] - \mathbb{P}[\text{test rejects}] = 2\mathbb{P}[\text{test accepts}] - 1.$$

We know the test accepts with probability at least $\frac{1}{2} + \delta$, so we have

$$\mathbb{E}_{x,y}[f(x)f(y)f(xy)] \geq 2\left(\frac{1}{2} + \delta\right) - 1 = 2\delta.$$

So we've now translated the information that the test accepts with probability $\frac{1}{2} + \delta$ to information about the expectation of a product of three values — we now know that

$$2\delta \leq \mathbb{E}_{x,y}[f(x)f(y)f(x \cdot y)].$$

In principle, if we just plug in the Fourier expansion of f three times and follow our nose, we'll get the exact same thing; but we'll present this with convolutions (as this is more elegant).

First, we'll pull out the expectation over x to get

$$2\delta \leq \mathbb{E}_x \left[f(x)\mathbb{E}_y[f(y)f(x \cdot y)]\right].$$

Now we see convolution entering the picture — the inner expectation is precisely $(f * f)(x)$, so we get that this is equal to

$$\mathbb{E}_x[f(x)(f * f)(x)] = \langle f, f * f\rangle.$$

Eventually we want to get to Fourier coefficients; and last time we saw Parseval and Plancherel, which let us go from inner products to inner products in the Fourier domain — this gives us

$$2\delta \leq \sum_{S\subseteq[n]} \widehat{f}(S)\widehat{f * f}(S).$$

And now we're going to use the claim about $\widehat{f * f}$; this gives us

$$2\delta \leq \sum_{S\subseteq[n]} \widehat{f}(S)^3.$$

Now we've gotten a sum of third powers, and we're going to pull out one of these powers outside the sum — this gives us

$$2\delta \leq \max_S \widehat{f}(S) \cdot \sum_S \widehat{f}(S)^2.$$

And we have a sum of squares of Fourier coefficients, which we can use Parseval on — Parseval tells us that $\sum_S \widehat{f}(S)^2 = \|f\|_2^2 = 1$ (since f is Boolean, so its 2-norm is just 1). And so we get $2\delta \leq \max_S \widehat{f}(S) \cdot 1$, and we're done. □

So we've now proven our theorem, which gives us a nice structural statement about f given that it passes the BLR test with probability at least $\frac{1}{2} + \delta$. We can now use this to prove the original theorem (that f is close to linear).

Proof. By the second theorem, there exists $S \subseteq [n]$ such that $\widehat{f}(S) \geq 2\delta$. Now we're going to unpack the Fourier coefficients to get the desired statement — we have

$$\widehat{f}(S) = \mathbb{E}_x[f(x)\chi_S(x)].$$

And we can use the same logic as at the beginning of the first proof — $f(x)\chi_S(x)$ is $+1$ when f and χ_S agree and -1 otherwise, so we can write this as

$$\widehat{f}(S) = \mathbb{P}[f(x) = \chi_S(x)] - \mathbb{P}[f(x) \neq \chi_S(x)] = 2\mathbb{P}[f(x) = \chi_S(x)] - 1.$$

And now we're done — we get

$$2\mathbb{P}[f(x) = \chi_S(x)] - 1 \geq 2\delta,$$

and rearranging gives the desired conclusion. □

§2.2.5 Some remarks

This is very elegant, and this theorem is very significant in many different areas. If we come from extremal combinatorics, we've probably seen arithmetic progressions — e.g., if a set doesn't have an arithmetic progression of size 3, then it can't be that large (Roth's theorem). The proof again involves third powers of Fourier coefficients; it's in the same ballpark, and this kind of reasoning is very important.

We'll now mention a related result in computational complexity.

Suppose that we have a collection of variables $\{z_1, \ldots, z_n\}$, which need to be assigned values in $\mathbb{F}_2$ (either 0 or 1). And we also have equations with these variables — which look like $z_{i_1} + z_{i_2} + z_{i_3} \equiv 1 \pmod 2$, $z_{j_1} + z_{j_2} + z_{j_3} \equiv 0 \pmod 2$, and so on. (So we have variables and linear equations, with each linear equation containing three variables.)

Question 2.15. Can we find an assignment of values to the z_i satisfying many of the equations?

If we're promised there's a solution satisfying *all* the equations, then we can find it in polynomial time (using linear algebra, e.g., diagonalizing matrices). But what happens when the system is *not* fully satisfiable?

Question 2.16. Suppose that the system is $(1-\varepsilon)$-satisfiable — i.e., there is an assignment of variables satisfying at least $1-\varepsilon$ of the equations. What's the best we can do?

Theorem 2.17

It is NP-hard to find an assignment satisfying $\frac{1}{2}+\delta$ of the equations.

We can always get $\frac{1}{2}$ of the equations (by setting our variables randomly), and it turns out we can't do better.

And we can use our earlier theorem to prove this; we won't see the proof though (it's much harder).

§2.3 Random restrictions

We'll now study some new tools that will help us. The next tool is very simple but very powerful; as with everything we'll see, the point is knowing how to use it.

Definition 2.18. Let $f\colon \{-1,1\}^n \to \mathbb{R}$ be a function, and let $J \subseteq [n]$ and $y = \{-1,1\}^J$. The *restricted function* $f_{J\to y}\colon \{-1,1\}^{\overline{J}} \to \mathbb{R}$ is the function defined by

$$f_{J\to y}(z) = f(x_J = y, x_{\overline{J}} = z).$$

So we choose a subset J of coordinates, and we also choose a bit-fixing y on this subset. We then define f as a function only on the coordinates in the *complement* of J.

In words, this is a very simple operation — we have a subset of the variables J, and we have some y that we want these variables to be. So we look at f, and we fix the coordinates in J to be the values we chose; while the rest of the coordinates are our input.

Example 2.19

Suppose that we have a function $f(x_1, \ldots, x_{10})$. Then we can define the restriction

$$g(x_4, \ldots, x_{10}) = f(1, -1, 1, x_4, \ldots, x_{10}).$$

In this example, g is $f_{J \to y}$ for $J = \{1, 2, 3\}$ and $y = (1, -1, 1)$.

(There's nothing complicated going on, but the notation can be a bit frightening at first.)

We've defined restrictions; now we'll define *random* restrictions, which are essentially just restrictions where the values of the fixed coordinates are chosen randomly.

Definition 2.20. Given f and $J \subseteq [n]$, a *random restriction* of f on J is a function $f_{J \to y}$ where y is chosen uniformly at random.

There's another definition of random restrictions — what if we choose J at random? We'll use this definition later, but for now we'll just look at this one.

Question 2.21. Suppose we have some function $f \colon \{-1, 1\}^n \to \mathbb{R}$, set $J \subseteq [n]$, and $y \in \{-1, 1\}^n$. What do the Fourier coefficients of $f_{J \to y}$ look like?

Lemma 2.22

Given $f \colon \{-1, 1\}^n \to \mathbb{R}$, $J \subseteq [n]$, and $y \in \{-1, 1\}^n$, for every $S \subseteq \overline{J}$ we have

$$\widehat{f_{J \to y}}(S) = \sum_{T \subseteq J} \widehat{f}(S \cup T) \chi_T(y).$$

Proof. Let's look at the value of our restricted function $f_{J \to y}$ at a point z; by definition this is

$$f_{J \to y}(z) = f(x_J = y, x_{\overline{J}} = z).$$

Now we're going to Fourier-expand f and hope for the best — we can write this as

$$f_{J \to y}(z) = \sum_{R \subseteq [n]} \widehat{f}(R) \chi_R(y, z)$$

(we write $\chi_R(y, z)$ as shorthand for $\chi_R(x_J = y, x_{\overline{J}} = z)$). We want to split our variables between J and $\overline{J}$, so instead of summing over all R, we'll sum over $R_1 \subseteq J$ and $R_2 \subseteq \overline{J}$ (so $R_1 \cup R_2$ ranges over all subsets); then we can write our character as $\chi_{R_1}(y)\chi_{R_2}(z)$, so we get

$$f_{J \to y}(z) = \sum_{\substack{R_1 \subseteq J \\ R_2 \subseteq \overline{J}}} \widehat{f}(R_1 \cup R_2) \cdot \chi_{R_1}(y) \chi_{R_2}(z).$$

Now we're thinking of y as fixed and z as our variable, so $\chi_{R_1}(y)$ is fixed and $\chi_{R_2}(z)$ are the characters we need to write $f_{J \to y}$ as a summation over (to get its Fourier expansion). So we can simply write this as

$$\sum_{R_2 \subseteq \overline{J}} \left(\sum_{R_1 \subseteq J} \widehat{f}(R_1 \cup R_2) \chi_{R_1}(y) \right) \chi_{R_2}(z).$$

We've now managed to write $f_{J \to y}(z)$ as a linear combination of the caracters on $\overline{J}$, so by *definition*, the coefficients here are the Fourier expansion of the function we started with (by uniqueness of the Fourier expansion). So we get

$$\widehat{f_{J \to y}}(R_2) = \sum_{R_1 \subseteq J} \widehat{f}(R_1 \cup R_2) \chi_{R_1}(y).$$

□

Now that we have this formula, we're going to play a bit with adding randomness to it and seeing what happens.

Question 2.23. Suppose that we look at a Fourier coefficient $\widehat{f_{J\to y}}(S)$. What is its *expectation* over a random choice of y?

If we look at our sum, $\chi_T(y)$ has expectation 0 for any nonempty T. So the only contribution is when T is empty, and we get

$$\mathbb{E}_y \widehat{f_{J\to y}}(S) = \widehat{f}(S).$$

In other words, the expected Fourier coefficient of the restriction is the original Fourier coefficient.

Question 2.24. What happens to the *square* of a Fourier coefficient — i.e., what is $\mathbb{E}_y \widehat{f_{J\to y}}(S)^2$?

Lemma 2.25

We have $\mathbb{E}_y \widehat{f_{J\to y}}(S)^2 = \sum_{T\subseteq\overline{J}} \widehat{f}(S\cup T)^2$.

Proof. Often when you work with restrictions, the game is about mentality — what's the variable, and what's fixed? We can think of $\widehat{f_{J\to y}}(S)$ as a function of y — it's a function mapping y to something, namely $\sum_{T\subseteq J} \widehat{f}(S\cup T)\chi_T(y)$. So the expectation of its square is the 2-norm of this function, which we can find using Parseval (since the above formula is its Fourier expansion).

In other words, we define $g(y) = f_{J\to y}(S)$, and use Parseval. □

So now we know restrictions, and we have some properties. Now we can finally introduce the actual restrictions used in applications, which are the ones where the set J is *also* chosen randomly.

Definition 2.26. For a function $f\colon\{-1,1\}^n\to\mathbb{R}$ and a parameter $p\in[0,1]$, a *p-random restriction* of f is a function $f_{\overline{J}\to z}$ where each $i\in[n]$ is included in J with probability p independently, and $z\in\{-1,1\}^{\overline{J}}$ is chosen uniformly at random.

This is the definition of a random restriction, and if you know how to use it well, you can prove many results using it. But the key is understanding what it's doing and how to use it. Today we'll try to give some intuition as to what this is doing, and then as far as time allows, we'll write some technical-looking claims that capture this intuition (but the intuition behind them is what's actually important).

Question 2.27. What are random restrictions doing?

To answer this, we'll consider some examples.

Example 2.28 (Monomials)

Consider $f(x) = \prod_{i\in S} x_i$ for some $S\subseteq[n]$ of size d (i.e., f is a character of size d), and suppose we apply a p-random restriction on f. What will the result look like?

After a p-random restriction, if we look at the variables that remain alive (which are the variables of J), roughly a p-fraction of the things in S will remain alive. So after the restriction, f will look like a monomial of degree roughly pd. And the other variables in S are going to get fixed to some value, so we'll have some sign.

So to summarize, monomials typically reduce their degree, and this reduction exactly corresponds to the rate of restricion p.

Example 2.29 (Logical AND)

Consider $f(x) = \prod_{i \in S} \mathbf{1}_{x_i = 1}$ (we can think of this as the logical AND of x_i for $i \in S$). What'll happen to f after a random restriction?

Again, roughly $p\,|S|$ variables will remain alive, and the rest will be set to 1 or -1. And if even *one* of them is set to -1, then this product will vanish. So typically a random restriction of f will be the constant 0.

Example 2.30 (CNF formulas)

Consider $f(x) = \bigwedge_{j \in S} \left(\bigwedge_{i \in I_j} \mathbf{1}_{x_i = 1}\right)$ (i.e., the function evaluates to 1 if and only if in each I_j, at least one of the variables is equal to 1). What happens to f after a random restriction?

One way to think about this is that random restrictions 'simplify' functions. So in this case, the intuition is that some of the clauses become trivial, so our number of clauses is going to shrink.

On Canvas, there are some technical claims that capture these intuitions.

§3 February 13, 2024

The first problem set is posted on Canvas; there are some questions which we probably can't solve yet, but will be able to at the end of this week (but there are some problems we already know how to solve).

§3.1 PAC Learning

Today we'll see some applications of random restrictions. We'll start with the PAC learning model, which we're going to define (we're not yet going to prove anything); this is sort of the ideal model you can hope to learn things in.

First, what is *learning*? In learning, we have some class of Boolean functions $\mathcal{C} \subseteq \{f \colon \{-1,1\}^n \to \{-1,1\}\}$, and there is some unknown function $f \in \mathcal{C}$ that we want to learn. How are we going to do this?

In the PAC learning model, what's given to us is a sequence of input-output pairs — so we're given a sequence $\{(x_i, f(x_i))\}_{i=1}^q$ generated uniformly at random. Our goal is to come up with a function $g \colon \{-1,1\}^n \to \{-1,1\}$ which is 'close' to f — i.e., the distance between f and g, defined as

$$\Delta(f,g) = \frac{\#\{x \mid f(x) \neq g(x)\}}{2^n},$$

is small.

There are various sub-definitions. For example, you can require $g \in \mathcal{C}$ as well; this is called *proper learning*. Or you can allow $g \notin \mathcal{C}$; this is called *improper learning*.

This is in some sense the most realistic model of learning. Often when we have functions we like (e.g., the input is someone's DNA and the output is whether they'd have some disease), we don't get to choose the input — we just get some random sample generated from a distribution we usually don't even know, and we only get to observe the input and output. And based on that, we have to make some conclusion about the output. That's why this model is viewed as a realistic model of learning.

Today we're not going to use this PAC model, because sometimes it's a bit too weak; there's many learning problems where if you allow more control over what queries you're allowed to make, we can be much more efficient than if you don't allow control.

For example, we talked about the *junta* functions — these are an example of a class of functions. There's large gaps between what we know how to do if the algorithm can specify what queries it wants to make vs. if it just gets random samples. (In fact, it's still an open question to know what exactly is the query complexity in the PAC model.)

Dor would love to tell us about PAC learning models, but that's not the topic of today; today we'll relax this model and study a very nice class of functions. We're going to slowly build our toolbox and then eventually see a learning algorithm in action.

§3.2 Estimating Fourier coefficients

We'll start with a basic fact.

Theorem 3.1 (Chernoff)
Suppose that $y_1, \ldots, y_n$ are independent random variables such that $|y_i| \leq 1$, and let $\varepsilon > 0$. Then

$$\mathbb{P}\left[\left|\sum y_i - \sum \mathbb{E}[y_i]\right| \geq \varepsilon n\right] \leq 2e^{-\frac{\varepsilon^2}{2+\varepsilon}n}.$$

In other words, the probability that $\sum y_i$ is decently far from its expectation is exponentially small. (The explicit constant in the exponent depending on ε doesn't really matter; it just matters that this is exponential.)

Remark 3.2. Later on in the course, we'll have the tools to prove this in the case where $y_i \in \{0, 1\}$. (We won't prove it in the general case.)

Now we'll see an application of this fact.

Lemma 3.3
Let $\varepsilon, \delta > 0$. Then there exists $q = O(\log(\delta^{-1})/\varepsilon^2)$ and a (PAC) learning algorithm that, given samples from a function $f\colon \{-1, 1\}^n \to \{-1, 1\}$ and a subset $S \subseteq [n]$, with probability at least $1 - \delta$ outputs a number $a_S \in \mathbb{R}$ which satisfies $|a_S - \widehat{f}(S)| \leq \varepsilon$.

So in simple words, we're saying that if we're given sufficiently many random samples of f, then we can approximate any Fourier coefficient of f that we'd like (our learning algorithm should output a number a_S close to the Fourier coefficient of S).

Proof. The Fourier coefficient is defined as an average $\widehat{f}(S) = \mathbb{E}_x[f(x)\chi_S(x)]$, where we sample *every* possible input from the domain. We don't have every possible input, but we do have q inputs; so we can try averaging the ones that we do have.

Let's suppose that we're given $\{(x_i, f(x_i))\}_{i=1}^q$. Then we can calculate the same average restricted to only $\{x_1, \ldots, x_q\}$ — for each i, we'll calculate $y_i = f(x_i)\chi_S(x_i)$ (we can do this because we know x_i and $f(x_i)$).

Now instead of averaging over *all* inputs to calculate $\widehat{f}(S)$, we'll average over the ones that we have — we'll define $a_s = \frac{1}{q}\sum_{i=1}^q y_i$ and output a_s.

This is our algorithm; we now just need to argue that it's correct. This is just by the Chernoff bound — the Chernoff bound tells us that the sum we got from our random variables should be close to its expectation. And so Chernoff tells us that

$$\mathbb{P}\left[\left|\frac{1}{q}\sum y_i - \mathbb{E}\frac{1}{q}\sum y_i\right| \geq \varepsilon\right] \leq 2e^{-\varepsilon^2 q/3}$$

(we're being a bit sloppy with the exponent). And this expectation is *exactly* the Fourier coefficient of S (because $\mathbb{E}[y_i] = \widehat{f}(S)$ for each i). This finishes the proof (with q chosen so that $2e^{-\varepsilon^2 q/3} \leq \delta$). □

We didn't do anything too clever here — we just did the most natural averaging. But this already tells us some very nice things.

Here's an example:

> **Question 3.4.** Suppose we have a function f where we know that $\sum_{|S| \leq k} \widehat{f}(S)^2 \geq 1 - o(1)$. Can we find an algorithm that comes up with a function g that's 'close' to f without making too many queries?

One idea is that we can estimate $\widehat{f}(S)$ for all $|S| \leq k$. Here we need to be a bit precise as to what accuracy we want. We have $\binom{n}{k} \leq n^k$ such Fourier coefficients, so we'll need to estimate each up to quite good accuracy — we'll need accuracy roughly n^{-k}.

Suppose that our estimates are a_S. Then we can define the function in the most natural way, replacing the Fourier coefficients by our estimates — we define $g(x) = \sum_{|S| \leq k} a_S \chi_S(x)$.

This does work; but the number of queries isn't that small — we need roughly $q = \mathsf{poly}(n^k, 2/\varepsilon, \log(1\delta))$ queries. This isn't bad, but if k is e.g. 100 then it's not great.

(If we have accuracy ξ, then we can show that $\lVert f - g \rVert_2^2 \leq n^k \cdot \xi^2 + o(1)$ — where the $o(1)$ corresponds to the Fourier coefficients from $|S| > k$. This is why we need quite good accuracy.)

But this is only using the power of uniformly given queries. We're now going to shift gears and allow ourselves stronger queries.

§3.3 Learning with membership queries

We'll now see a stronger model of queries.

> **Definition 3.5.** *Membership queries* refer to the ability to request the value of a function f on any particularly chosen input $x \in \{-1, 1\}^n$.

This comes in several flavors — it can be adaptive (we ask $f(x_1)$, get it, and then based on this we ask $f(x_2)$, and so on), or non-adaptive (where we choose $x_1, \ldots, x_{100}$ at the start, ask all of them and get responses, and based on that we try to learn the function).

> **Remark 3.6.** Whenever we do any sublinear-type stuff, there's many different models we can ask for; we won't discuss all of them, but we will try to mention some of the distinctions.

So this is the extra power we have — we can choose whatever input x we want, and get $f(x)$. Other than this, the game is the same — there's some unknown function f from a known class $\mathcal{C}$, and our goal is to learn f using as few queries as possible.

The class that we'll be after today is the class of *sparse* functions (functions which are sparse in the Fourier domain) — i.e.,

$$\mathcal{C} = \{f \colon \{-1,1\}^n \to \{-1,1\} \mid \widehat{f}(S) \neq 0 \text{ for at most } t \text{ sets } S \subseteq [n]\}.$$

We'll think of t as constant; the catch is that we of course don't know which Fourier coefficients are 0 or nonzero. So based on what we've developed so far, there seems to be no hope to learn a function from this class, because we have no idea what Fourier coefficients we're after.

§3.4 Some techniques

We'll now develop some techniques to help us handle this class of functions.

We'll first state a claim that looks completely unrelated, but is actually the engine of what will allow us to do the learning algorithm.

Lemma 3.7

For every $\varepsilon, \delta > 0$, there exists $q = O(\log(\delta^{-1})/\varepsilon^2)$ and an algorithm with q (membership) queries that, given a function $f\colon \{-1,1\}^n \to \{-1,1\}$ and a pair of subsets $T \subseteq J \subseteq [n]$, outputs a number $b_{T,J}$ such that with probability at least $1-\delta$, we have

$$\left| b_{T,J} - \sum_{S \cap J = T} \widehat{f}(S)^2 \right| \leq \varepsilon.$$

In other words, our goal is to approximate the sum of squares of all Fourier coefficients $\widehat{f}(S)$ over sets S such that when projected onto J, they look like T.

This probably looks completely unrelated to the problem of learning t-sparse functions, but we'll first prove it and then see why it's useful.

Proof. The idea is that this expression comes from random restrictions — last time, we looked at what happens to Fourier coefficients when we take a random restriction, and the expected *square* of the Fourier coefficient was exactly the expression we see here.

More explicitly, last lecture we saw that if we take $z \in \{-1,1\}^{\overline{J}}$ and consider $\mathbb{E}_z f_{\overline{J}\to z}(T)^2$, this is exactly the sum $\sum_{S\cap J=T} \widehat{f}(S)^2$.

So now we're in good shape — we've expressed the thing we wish to approximate as an expectation. And since this is an expectation, it behaves well with respect to Chernoff bounds; and we've already seen how to approximate Fourier coefficients. So it morally makes sense that this should work; but we'll now prove this.

First, opening up our restriction, we have

$$\sum_{S\cap J=T} \widehat{f}(S)^2 = \sum_{z} f_{\overline{J}\to z}(T)^2 = \mathbb{E}_z \left|\mathbb{E}_x f_{\overline{J}\to z}(z)\chi_T(x)\right|^2 = \mathbb{E}_{z,x,x'} f(z,x)f(z,x')\chi_T(x)\chi_T(x').$$

(The reason we have x and x' is by opening the square — the square corresponds to sampling two x's.) And now that we have an expectation, we can immediately get an algorithm.

Algorithm 3.8 — Sample q pairs (z_i, x_i) and (z_i, x_i') for $i = 1, \dots, q$, and for each one of them, compute

$$y_i = f(z_i, x_i) f(z_i, x_i') \chi_T(x_i) \chi_T(x_i').$$

Output $b_{T,j} = \frac{1}{q}\sum y_i$.

To see that this works, we can use Chernoff again; the analysis is the exact same thing as before, so we won't do it again. □

Here's a tricky question — why did we need membership queries here? It's because we have (z_i, x_i) and (z_i, x_i') — so we need inputs which are correlated, meaning the J-parts are the same. If we just had uniform inputs, this would be very unlikely to happen — it'd happen only with probability $2^{-|J|}$. So to get inputs this correlated, we need to design them ourselves; and for this we need membership queries.

So at this point, we know how to estimate Fourier coefficients, and certain sums of squares of Fourier coefficients. But now we're ready to organize a learning algorithm for sparse functions.

§3.5 Learning algorithm for sparse functions

Definition 3.9. We say $f\colon \{-1,1\}^n \to \{-1,1\}$ is (t,ε)-*sparse* if there exists a function $g\colon \{-1,1\}^n \to \mathbb{R}$ which is t-sparse (in the Fourier domain) and satisfies $\|f-g\|_2^2 \le \varepsilon$.

Our goal is to study (t,ε)-sparse functions, and not just t-sparse functions.

Theorem 3.10

For every $t \in \mathbb{N}$ and $\varepsilon, \delta > 0$, there exists $q = \mathsf{poly}(n, t, \varepsilon^{-1}, \delta^{-1})$ and a q-time algorithm such that given oracle access to a function $f\colon \{-1,1\}^n \to \{-1,1\}$ that is (t,ε)-sparse, outputs a function $h\colon \{-1,1\}^n \to \mathbb{R}$ which is t-sparse and such that $\|f-h\|_2^2 \le 4\varepsilon + \delta$, with probability at least $1-\delta$.

Remark 3.11. You could think of this as saying the query complexity of the algorithm is at most q, but we can make an even stronger statement — that the *runtime* is at most q.

Morally speaking, this says that if we have a function f which is sparse, then we can learn it quite well using polynomially many queries.

§3.5.1 Some preliminary observations

First, there's a few preliminary observations.

Let $\xi = \delta/8t$. First, we'll say that if we want to come up with h, then it suffices to estimate the 'heavy' Fourier coefficients of f. To formalize this, let $\mathcal{S} = \{S \subseteq [n] \mid \left|\widehat{f}(S)\right| \ge \xi\}$ be the set of Fourier coefficients of f which are reasonably large (at least ξ in absolute value).

We know that $\sum_S \widehat{f}(S)^2 = 1$ (where the sum is over all S). We want to argue that if we look at the Fourier mass contributed from coefficients *outside* $\mathcal{S}$, this is very small.

Claim 3.12 — We have $\sum_{S\notin\mathcal{S}} \widehat{f}(S)^2 \le \varepsilon + \delta/8$.

What this says is that we don't really have to care about the Fourier coefficients outside $\mathcal{S}$ — they don't contribute much at all in total. So then most of our discussion will be focused on the Fourier coefficients inside $\mathcal{S}$; and we'll see there's not too many of them.

First, here's a proof of the claim.

Proof. It's helpful to first think about the case where f itself is t-sparse. Then when we look at this sum, there's at most t terms that are nonzero. And because we're outside $\mathcal{S}$, each of these nonzero terms is at most ξ^2. So in total, we get an upper bound of $t\xi^2 \le \delta/8$.

But f isn't t-sparse; rather, it's (t,ε)-sparse. So we need to do something slightly more complicated — let g be the t-sparse function closest to f, so that we know $\|f-g\|_2^2 \le \varepsilon$ (by our assumption on f).

Now we have a sum $\sum_{S\notin\mathcal{S}} \widehat{f}(S)^2$ that we want to estimate. And we can rewrite this as

$$\sum_{S\notin\mathcal{S}} |\widehat{f}(S) - \widehat{g}(S)|^2 \mathbf{1}_{\widehat{g}(S)=0} + \sum_{S\notin\mathcal{S}} |\widehat{f}(S)|^2 \mathbf{1}_{\widehat{g}(S)\neq 0}$$

(we introduce a $\widehat{g}(S)$ term whenever it's 0; whenever it's not 0 we can't introduce it).

For the first term, by Parseval we have

$$\sum |\widehat{f}(S) - \widehat{g}(S)| \le \|f-g\|_2^2 .$$

And for the second term, since g is t-sparse, there's only t terms in the sum (since $\widehat{g}(S) \neq 0$ for at most t sets S). So there's at most t summands and each is at most ξ^2. This gives us a bound of $\|f-g\|^2 + t\xi^2$, which if we do some computations is at most $\varepsilon + \delta/8$. □

So we've proven the desired inequality, which we can think of as stating that only heavy Fourier coefficients matter — because the non-heavy Fourier coefficients have very small total contribution.

We'll need one more preliminary observation — we claim that the set $\mathcal{S}$ cannot be very large.

> **Claim 3.13** — We have $|\mathcal{S}| \leq 1/\xi^2$.

Proof. We have $\sum_S \widehat{f}(S)^2 = 1$ by Parseval (summing over all Fourier coefficients), and each $S \in \mathcal{S}$ contributes at least ξ^2 to the sum. □

These two observations mean that we only have to care about the heavy coefficients, and there aren't too many of them. So this makes the problem seem more tractable.

But we're still missing the key idea — we know there's only a small number of things we need to estimate, but we don't know which are the things we want. So the next step is to figure out how to *locate* the heavy Fourier coefficients.

§3.5.2 Locating the heavy coefficients

We'll now move on to trying to locate the heavy Fourier coefficients.

To motivate the solution, suppose we want to answer the following question:

> **Question 3.14.** Is there any heavy Fourier coefficient containing the element 1? (More precisely, can we figure this out with a few queries?)

This is related to the lemma we proved earlier — we can estimate the sum of squares of Fourier coefficients containing 1 using that lemma. In the lemma, we're summing over all Fourier coefficients S such that when we project them onto J, they look like T. So we can take $J = T = \{1\}$ — then we're looking at all S containing 1.

So what we've learned is that we can identify the sum of squares of Fourier coefficients containing 1. And similarly, we can estimate the sum of squares of Fourier coefficients that *don't* contain 1, by taking $J = \{1\}$ and $T = \emptyset$.

So we can imagine drawing a tree: first, we try to estimate the sums of squares of Fourier coefficients with $1 \notin S$, and those with $1 \in S$. This gives us two estimates $b_{\emptyset,\{1\}}$ and $b_{\{1\},\{1\}}$.

Suppose that when we do these estimates, we get that $b_{\emptyset,\{1\}} = 0$. This tells us we shouldn't go down that branch of the tree — there's no heavy coefficients there, so there's no reason to go down that branch. Meanwhile, if there *is* mass there, then we *should* go down that path.

This captures the idea of what's going on, but now we'll write the algorithm down explicitly.

> **Algorithm 3.15** — We'll have an iterative algorithm, with iterations $k = 1, \ldots, n$.
>
> At each point, the algorithm maintains a list of 'live' nodes $L_k = \{A \subseteq [k] \mid b_{A,[k]} \geq \xi^2/2\}$.
>
> In the iteration step, for each $A \in L_k$, we estimate $\sum_{S \cap [k+1] = A} \widehat{f}(S)^2$ and $\sum_{S \cap [k+1] = A \cup \{k+1\}} \widehat{f}(S)^2$ (we call these estimates $b_{A,[k+1]}$ and $b_{A \cup \{k+1\},[k+1]}$). If the first estimate is at least $\xi^2/2$, then we insert A into L_{k+1}. If the second estimate is at least $\xi^2/2$, then we insert $A \cup \{k+1\}$ into L_{k+1}.

In other words, at each point in the algorithm, we're developing our tree one more layer. We'll have some live nodes. When we perform the next iteration, we're only going to develop the live nodes (the rest of nodes are kind of dead, and we don't care about them).

To do this, for each $A \in L_k$, we estimate what happens if $k+1$ is not in our coefficient, and what happens if it is — so our node A_k splits into two.

What's happening in the tree is that we keep live nodes. At each step we try to explore different coordinates — the first layer corresponds to exploring 1, the second to 2, and so on. At each point, some nodes will be dead and some live; we only try to grow the tree from live nodes.

Remark 3.16. As a technical note, for all of this to work, we need to estimate with precision something like $\xi^2/10$ and probability of error at most $\delta\xi^2/n$. So we have to pick the parameters accordingly; and if we do, then we get the right bounds. But we're not going to do this.

Importantly, why isn't it the case that we're just going to branch over all the leaves?

Claim 3.17 — With probability $1 - o(1)$, we have $|L_k| = O(1/\xi^2)$ for all k.

Proof. To prove this precisely, we have to do induction on k; we're going to handwave instead. The point is that if we look at a certain level in the algorithm and look at the corresponding Fourier sums at all the nodes, then the total sum of all of them is at most 1 (since it's just the total Fourier mass of f). So if we have enough that are at least $\xi^2/2$, then we're heavy. (We have to be a bit more precise, because we have some error probability.) □

In the end, we have a list L_n. At this point we're just talking about Fourier coefficients, so L_n is such that with high probability $\mathcal{S} \subseteq L_n$, and $|L_n| = O(1/\xi^2)$.

§3.5.3 The conclusion

Now we do the obvious thing — for each character in L_n we just estimate it, and we're done. Explicitly, we estimate each $\widehat{f}(S)$ for $S \in L_n$ by some a_S, with accuracy $\varepsilon\xi^2$. We then take $h(x) = \sum a_S \chi_S(x)$ (taking a_S to be the Fourier coefficients of our new function).

Then when we look at $\|f - h\|_2^2$, we get

$$\|f - h\|_2^2 = \sum_S |\widehat{f}(S) - \widehat{h}(S)|^2.$$

We can split this into terms where $\widehat{h}(S)$ is zero and nonzero — we get

$$\sum_{S \notin \mathcal{S}} |\widehat{f}(S)|^2 + \sum_{S \in \mathcal{S}} |\widehat{f}(S) - a_S|^2.$$

The first term is at most $\varepsilon + \delta/8$ (by what we started the proof with). And for the second, there's at most $|\mathcal{S}|$ terms, and we did an approximation up to accuracy $\varepsilon\xi^2$. Using that $|S| = O(1/\xi^2)$, we get that this is $O(\varepsilon + \delta)$.

Remark 3.18. The main idea of the proof is using the tree strategy to search for all relevant sums, and observing that there aren't too many.

§4 February 15, 2024

Today we're going to define the notion of *influences*, which is very important. We're going to motivate it in a somewhat weird way, by voting; but most of the applications we'll see have nothing to do with voting.

§4.1 Voting systems

We've talked a bit about Arrow's theorem, but we didn't prove it. But in general, when you have a Boolean function $f\colon \{0,1\}^n \to \{0,1\}$, you can think of it as a way to aggregate votes (i.e., as a *voting system*). Explicitly, suppose we are trying to make a decision, and each one of n voters is either for the decision (in which case they vote $x_i = 1$) or against it (in which case then vote $x_i = 0$).

When we look at all these votes, we get a vector of votes $x = (x_1, \ldots, x_n)$. But eventually we want to reach a decision — do we accept the decision or not? So we can think of f as aggregating all these votes — we aggregate via f to get $f(x)$.

Any boolean function can be thought of in this way. From day-to-day life, there's several examples.

When we talk about elections (not in the U.S.), we often think about *majorities*.

Example 4.1

The *majority* function is defined as $f(x_1, \ldots, x_n) = 1$ if and only if at least half of the x_i are 1 (i.e., $f(x_1, \ldots, x_n) = \mathbf{1}_{\sum x_i \geq n/2}$).

Another voting rule, from longer ago, is a dictatorship.

Example 4.2

A *dictatorship* function is $f(x_1, \ldots, x_n) = x_1$.

Another way to vote (it's debatable whether it's common today or not) is when instead of just having one person whose vote matters, there may be 10 or 20.

Example 4.3

A *junta* is a function $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_{20})$.

The word *junta* refers to the fact that only very few of the voters matter (and the rest don't).

You can think about many other voting schemes as well.

These voting schemes seem clearly different — in a majority each voter has a little bit of effect on the outcome, but they don't fully determine it. Meanwhile, in a dictatorship only one voter has an effect, but they do fully determine it.

This is where the notion of influence comes in.

§4.2 Influence

Definition 4.4. For a function $f\colon \{-1,1\}^n \to \{0,1\}$, the *influence* of a coordinate $i \in [n]$ is defined by

$$I_i[f] = \mathbb{P}_{x \in \{-1,1\}^n}[f(x) \neq f(x \cdot e_i)]$$

(where e_i is the vector with 1's in all coordinates except a -1 in coordinate i).

In other words, we sample all the votes uniformly at random and ask what's the probability that if the ith voter flips their vote, the outcome changes?

For now, we've defined influence for functions with range $\{0, 1\}$. You can generalize the definition to functions taking any value; we'll do this later on, because we'll want Fourier analytic formulas. But for now we'll stick to this case.

Example 4.5

In a dictatorship f, we have $I_1[f] = 1$ and $I_i[f] = 0$ for all $i \neq 1$.

Example 4.6

In a majority f, if $n = 2m + 1$ is odd, then we have $I_i[f] = 2^{-2m}\binom{2m}{m}$ (since we ignore the variable i, and we need the counts of ± 1 among the other variables to be equal). By Stirling's formula, this is $\Theta(n^{-1/2})$.

Example 4.7

In a junta, for $i > 20$ we have $I_i[f] = 0$, while the influence of 1, . . . , 20 depends on g.

Here's a motivating question, which we'll answer over the next few lectures. We'd like to think about voting, and we'd like our voting to be 'fair'; we'd really like to find a voting system where all influences are as small as possible.

Question 4.8. What is a function f where all influences are as small as possible?

The trivial answer is that f could be constant; but this is not a very good voting system (we're just taking all the votes, putting them into the garbage can, and outputting 0). So we'll require that f is balanced — i.e., that $\mathbb{P}[f = 0] = 1/2$.

Question 4.9. Given that $\mathbb{P}[f = 0] = 1/2$, what is $\min_f \max_i I_i[f]$?

This is a nontrivial question, but we'll think of it as motivation. Today we'll prove a very modest result — that this is always at least $1/n$. (So you cannot make all influences very close to 0.) (We'll see later that this is not the actual answer.)

Here's another central notion.

Definition 4.10. The *total influence* of f is defined as $I[f] = \sum_{i=1}^{n} I_i[f]$.

This is a very central notion attached to a Boolean function; today Dor will convince us that this is true, in the sense that there's three equivalent interpretations of what this captures and each is useful.

§4.2.1 A more general definition

The topic of today is understanding influence and total influence. But to do this, we'll first have to generalize the definition coming out of voting schemes to something more analytical; this is what we'll do now.

Definition 4.11. Suppose we have a function $f\colon \{-1,1\}^n \to \mathbb{R}$ and $i \in [n]$. Then we define the *discrete derivative* of f in direction i, denoted $\partial_i f\colon \{-1,1\}^{n-1} \to \mathbb{R}$, as

$$\partial_i f(y) = \frac{1}{2}(f(x_i = 1, x_{-i} = y) - f(x_i = -1, x_{-i} = y)).$$

So given y, we look at what happens when we plug in $x_i = 1$ and plug in the rest of the coordinates according to y, and what happens when we switch x_i to -1; and we subtract them. The reason we call this a derivative is that in calculus, the derivative measures the effect of changing one variable; and we're doing the same here.

Definition 4.12. The L^2*-influence of coordinate* i on f is defined as

$$I_i[f] = \lVert \partial_i f \rVert_2^2 .$$

We define the *total influence* as $I[f] = \sum_{i=1}^n I_i[f]$.

This definition is the same as the one for $\{0,1\}$-valued functions up to a constant factor (they're off by a factor of 4; if we used ±-valued functions instead then we'd get that they're the exact same). So they're not equivalent exactly, but they morally are.

Remark 4.13. The reason we call this the L^2-influence is that sometimes people also use other norms to measure the discrete derivative. But this is the most common one (because it has nice formulas); and when we say influence in this course, we're referring to the L^2-influence unless otherwise specified.

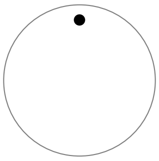
We've promised that influence and total influence is a central notion; now we'll support this claim by looking at a few interpretations of it.

§4.3 A combinatorial view of influence

Consider a graph G on vertices $\{-1,1\}^n$, with edges between two vertices that differ by exactly one coordinate — i.e., $V = \{-1,1\}^n$ and $E = \{(x, x \cdot e_i) \mid x \in \{-1,1\}^n, i \in [n]\}$.

Now suppose that we have some Boolean function $f\colon \{-1,1\}^n \to \{0,1\}$. In the graph language, this corresponds to taking a subset of vertices $F = \{x \mid f(x) = 1\} \subseteq \{-1,1\}^n$.

Then when we have a subset of vertices, we can ask ourself, for each vertex in the set, how many edges land outside the set?



Definition 4.14. We define $S_F(x) = \#\{i \mid x \cdot e_i \notin F\}$ for each $x \in F$, and $S_F(x) = \#\{i \mid x \cdot e_i \notin F\}$ for each $x \notin F$.

In other words, we can think of having a bipartition of our graph defined by F, and we're trying to capture counts of edges that go between the two parts.

We can sample an element and then look at the *average* number of edges that cross the cut. And this turns out to be exactly the total influence.

Claim 4.15 — We have $\mathbb{E}_x S_f(x) = I[f]$.

(Here we need the original definition of influence, not the new one; but with the new one, we just get a factor of $\frac{1}{4}$.)

Proof. We need to define some random variables. We can think of x as being chosen uniformly, and we define $Z_i[x] = \mathbf{1}_{f(x) \neq f(x \cdot e_i)}$ as the indicator variable of whether the corresponding edge crosses the cut. Then we can write

$$S_F(x) = \sum_{i=1}^{n} Z_i[x]$$

(by definition). Now we have an expectation and a sum, so we can exchange them — we have

$$\mathbb{E} S_F(x) = \mathbb{E} \sum_i Z_i[x] = \sum_i \mathbb{E}_x Z_i[x].$$

And $\mathbb{E}_x Z_i[x]$ is just $I_i[f]$, so we're done. □

§4.4 Sharp thresholds

So this is the combinatorial view on what influences are; now we're going to move to the view of sharp thresholds. We're going to switch a lot between $\{0,1\}$ notation and $\{-1,1\}$ notation (since which one is more natural depends on the application), but it doesn't really matter. Here we'll consider functions $f\colon \{0,1\}^n \to \{0,1\}$.

Definition 4.16. We say a function $f\colon \{0,1\}^n \to \{0,1\}$ is *monotone* if given two inputs $x, y \in \{0,1\}^n$, if $x \le y$ (i.e., $x_i \le y_i$ for each i), then we have $f(x) \le f(y)$.

(The reason we switched to $\{0,1\}$-notation is that if we identify 0 and 1 with 1 and -1, then the inequalities flip, and things become a bit confusing.)

The dictatorship and majority functions are both monotone. In random graph theory, there's many properties you'd like to study which are monotone functions.

Example 4.17 (Random graphs)

Suppose that $n = \binom{N}{2}$. We can then associate a vector $x \in \{0,1\}^n$ with a graph G_x on vertex set $[n]$, where the entries of x tell us whether each edge exists or not. (For every potential edge $\{u,v\}$ in G_x, we have a coordinate in $[n]$, and the value of x at this coordinate indicates whether the edge exists in G_x or not — 1 means there's an edge, and 0 means there is no edge.)

Here are some examples of monotone functions:

- $f(x) = 1$ if and only if G_x is connected. (This is monotone because if we have some connected graph and we add edges, then it remains connected.)
- $f(x) = 1$ if and only if G_x contains a perfect matching.

(There's many more interesting examples.)

There's one slight issue, which is a bit incompatible with what we discussed so far. It's quite easy to prove that if you sample a graph *uniformly* at random (i.e., each edge exists with probability $\frac{1}{2}$), then it's connected with *extremely* high probability (exponentially close to 1). So both functions are morally constant.

So we have to adapt our view a bit — we have to change the measure on $\{0,1\}^n$ (we can't just take each coordinate to be 0 or 1 with probability $\frac{1}{2}$).

Definition 4.18. The *p-biased measure* on $\{0,1\}^n$, denoted $\mu_p^{\otimes n}$, is defined such that if we want to sample $x \sim \mu_p^{\otimes n}$, then each x_i is 1 with probability p and 0 otherwise, independently for each i.

So far (and in the majority of the course), we only have dealt with $p = \frac{1}{2}$. But for these examples we need to think about other values of p; the examples mentioned here are interesting for $p = \Theta(1/n)$ or $p = \Theta(\log n/n)$.

Now how do we generalize influence to this setting?

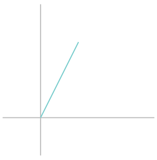
Definition 4.19. The influence of a function f with respect to $\mu_p^{\otimes n}$, defined $I_i[f; \mu_p^{\otimes n}]$, is defined as $\mathbb{E}_{x\sim\mu_p^{\otimes n}} |\partial_i f(x)|^2$, and the total influence is defined as $I[f; \mu_p^{\otimes n}] = \sum_i I_i[f; \mu_p^{\otimes n}]$.

Previously we defined influence using a 2-norm, and 2-norms are defined using a measure; in this definition we're just changing the measure from the uniform one to the p-biased one.

Let's inspect the function of connectivity. If you sample a uniformly random graph (where each edge appears with probability 1/2), it's not hard to show it's connected with probability 1 minus something exponentially small. Meanwhile, it's very easy to prove that if $p = 0$, then the graph is *never* connected.

So we can imagine considering $\mu_p(f) = \mathbb{P}_{x\sim\mu_p^{\otimes n}}[f(x) = 1]$ as a function of $p \in [0,1]$. If we look at this connectivity function, we've said that at $p = \frac{1}{2}$ the function is essentially almost always 1, whereas if $p = 0$ then of course it's 0.

But what goes on in between these two points?



In general, it's possible to show that whenever f is monotone, so is this graph — as we increase p, the probability that f is going to fire increases.

Fact 4.20 — If f is monotone, then the map $p \mapsto \mu_p(f)$ is monotone.

Question 4.21. How sharp is this increase?

In other words, can we say anything about the rate at which this function increases?

Theorem 4.22 (Russo–Margulis)
We have $\frac{d\mu_p(f)}{dp} = I[f; \mu_p^{\otimes n}]$ for any monotone $f\colon \{0,1\}^n \to \{0,1\}$ (and for all p).

So this is the second interpretation of total influence — it captures the slope of this curve.

Remark 4.23. Is it usually easy to calculate this total influence? We will touch on this question later. For now, we'll comment that if you want to prove a fuction has a sharp threshold, you just want to eliminate the possibility that this is small. For this, you prove theorems that if your function has small total influence, then stuff happens. These are very important theorems, but they're not completely understood.

There's a famous paper that uses these sorts of stuff (how we eliminate the possibility some specific function has a small influence) to prove that k-SAT has a sharp threshold. (Just as Boolean strings can encode graphs, they can also encode k-SAT formulas.) To this day no one knows how to compute the influences for k-SAT, but we do know that they can't be very small.

Usually, the way this is proved is by expressing $\mu_p(f)$ as a function of p's and f's and taking a derivative, and stuff happens. Dor will give us a proof that's probably 'worse' but may make more intuitive sense.

Proof. We're going to take $\varepsilon > 0$ to be very small and sample $x \sim \mu_p^{\otimes n}$. And we're also going to sample $y \sim \mu_{p+\varepsilon}^{\otimes n}$ in a coupled way — such that $x \leq y$.

How do we do this? We can first sample $x \sim \mu_p^{\otimes n}$. Now we need to sample y. We'll sample each coordinate i independently. If $x_i = 1$, then our hand is forced — we need $y \geq x$, so we need to take $y_i = 1$. Meanwhile, if $x_i = 0$, then we take y_i to be 1 with probability $\frac{\varepsilon}{1-p}$ and 0 otherwise.

As a sanity check, we have

$$\mathbb{P}[y_1 = 1] = p \cdot 1 + (1-p) \cdot \frac{\varepsilon}{1-p} = p + \varepsilon,$$

as desired. So we've managed to get x and y which are coupled, and which are marginally distributed in the ways we want.

Now we can look at $\mu_{p+\varepsilon}(f) - \mu_p(f)$. And we're going to use this coupling — by definition this is $\mathbb{E}_y f(y) - \mathbb{E}_x f(x)$ (where we draw x and y according to the distributions we're generating them from). And we can write this as $\mathbb{E}_{x,y}[f(x) - f(y)]$. We said x and y are really tiny, so there's a chance that these are the same; in that case $f(x) = f(Y)$, so we can write this as

$$\mathbb{E}_{x,y}[(f(y) - f(x))\mathbf{1}_{x \neq y}].$$

Now if $x \neq y$, we claim that most likely they differ in exactly one coordinate — the probability they differ on coordinate 1 is roughly ε, while the probability they differ in both 1 and 2 is ε^2, which is tiny. So we can write this as

$$\sum_{i=1}^{n} \mathbb{E}_{x,y}[(f(y) - f(x))\mathbf{1}_{y,x \text{ differ one one coordinate}} + O_n(\varepsilon^2)]$$

(we don't really care about what the $O_n(\varepsilon^2)$ term is, because essentially we'll take $\varepsilon \to 0$).

And this is exactly $(x - O_n(\varepsilon^2))I_i[f; \mu_p^{\otimes n}]$.

Then we get that ε times this sum is $\varepsilon I[f] + O_n(\varepsilon^2)$. And now we're done — consider

$$\frac{\mu_p(f)}{dp} = \lim_{\varepsilon \to 0} \frac{\mu_{p+\varepsilon}(f) - \mu_p(f)}{\varepsilon}.$$

So we end up with

$$\lim_{\varepsilon \to \infty} I[f, \mu_i^{\otimes n}] + O_n(\varepsilon) = I[f; \mu_p^{\otimes n}].$$ □

This is maybe a slightly worse proof because there's coupling and ε^2 and stuff, but in some sense this proof explains why the theorem is true — when we look at a derivative we're essentially just increasing one coordinate, while the influence exactly captures increasing one variable.

§4.5 Some Fourier analytic formulas

Finally we'll look at influence from an analytic point of view, and prove some formulas.

First, there's a simple formula for the Fourier expansion of the discrete derivative.

Claim 4.24 — We have $\partial_i f(x) = \sum_{S \ni i} \widehat{f}(S)\chi_{S\setminus\{i\}}(x)$.

Proof. By definition we have $\partial_i f(y) = \frac{1}{2}(f(1,y) - f(-1,y))$. (By $(1,y)$ we mean that coordinate i is 1, and the rest of the coordinates are y.) Now we can just plug in the Fourier expansion of f to get that this is

$$\frac{1}{2}\sum_S \widehat{f}(S)\chi_S(1,y) - \sum_S \widehat{f}(S)\chi_S(-1,y). = \frac{1}{2}\sum_S \widehat{f}(S)(\chi_S(1,y) - \chi_S(-1,y)).$$

Now we're almost done. Suppose frist that S doesn't contain i. Then this diference is 0, because the value at i doesn't affect the value of the character — so if $i \notin S$ then $\chi_S(1,y) - \chi_S(-1,y) = 0$. meanwhile, if $i \in S$ then we get $\chi_S(-1,y) = -\chi_S(1,y)$, so $\chi_S(1,y) - \chi_S(-1,y) = 2\chi_{S\setminus\{i\}}(y)$. (The reason we put a $\frac{1}{2}$ in the definition is to cancel this.) So we end up with $\partial_i f = \sum_{S\ni i} \widehat{f}(S)\chi_{S\setminus\{i\}}(y)$, and we're done. □

Now we have a formula for the derivative; we'll use it to get one for the influences.

Claim 4.25 — We have $I_i[f] = \sum_{S\ni i} \widehat{f}(S)^2$.

Proof. The proof is Parseval. (The influence is the 2-norm, and Parseval tells us that the 2-norm is the sum fo squares of the Fourier coefficients.) □

Claim 4.26 — We have $I[f] = \sum_{S\subseteq[n]} |S|\,\widehat{f}(S)^2$.

Proof. We defined $I[f] = \sum_i I_i[f] = \sum_i \sum_{S\ni i} \widehat{f}(S)^2$. We can rewrite this as $\sum_i \sum_S \widehat{f}(S)^2 \mathbf{1}_{i\in S}$, and if we interchange the summation then we get precisely the above sum — we get $\sum_S \widehat{f}(S)^2 \sum_i \mathbf{1}_{i\in S} = \sum |S|\,\widehat{f}(S)^2$. □

This last formula gives us a third interpretation of the influence — if we looked at the quantity $\sum |S|\,\widehat{f}(S)^2$, the most reasonable name for it might be the 'average degree' of our function. (This is because χ_S has degree $|S|$, and the squares of Fourier coefficients define a sort of distribution over sets S, since they sum to 1.)

So this is our third interpretation — the total influence is the same as the average degree.

Remark 4.27. In our first interpretation, we saw that it's also the average degree in a graph, which is called the average *sensitivity*. There was a huge open problem about showing a better relationship between sensitivity and degree, which we may talk about later.

Corollary 4.28 (Poincare's inequality)
We have $I[f] \geq \mathrm{Var}[f]$ for all f.

Proof. We've seen that $I[f] = \sum |S| \widehat{f}(S)^2$. And earlier, we proved that $\mathrm{Var}[f] = \sum_{S \neq \emptyset} \widehat{f}(S)^2$. Since $\widehat{f}(S)^2$ is always nonnegative and any nonempty S has $|S| \geq 1$, this is clear (all the corresponding terms in $I[f]$ are at least those in $\mathrm{Var}[f]$). □

This is very simple, but it already tells us an answer to the question we started with.

Lemma 4.29

Suppose we have some balanced Boolean function — i.e., a function $f \colon \{-1,1\}^n \to \{-1,1\}$ with $\mathbb{E} f = 0$. Then there exists i with $I_i[f] \geq 1/n$.

Proof. We must have $\mathrm{Var}[f] = 1$, so $I[f] \geq 1$. But $I[f]$ is a sum of n individual influences $I_i[f]$, so one of these influences must be at least $1/n$. □

We started the lecture talking about a voting scheme, and we proved there's a limit to how little the influence of a voter can be.

But if you think about trying to actually construct f with all influences $1/n$, you'll probably fail; because it turns out there is no such function. This is a very important result in the area, which is a vast strengthening of this result. (Depending on what your expectation is, it will either look very disappointing or very impressive.)

Theorem 4.30 (Kahn–Kalai–Linial)

For every $f \colon \{-1,1\}^n \to \{-1,1\}$, there exists $i \in [n]$ such that

$$I_i[f] \geq c \cdot \frac{\log n}{n} \cdot \mathrm{Var}[f]$$

(where c is some absolute constant).

(We can think of f as balanced, in which case $\mathrm{Var}[f]$ is just 1.)

At this point in time, it's hard to say why the $\log n$ matters so much — usually when you insert a log somewhere, no one cares. But here it's very important. (All the connections we saw earlier, e.g. regarding sharp thresholds, use this.)

Remark 4.31. This bound is tight.

We can't prove this result right now — it requires a very important tool called the *hypercontractivity inequality*, which we'll discuss next week.

Remark 4.32. We don't have class on Tuesday, because Tuesday is a Monday schedule.

§5 February 22, 2024

Today we'll discuss hypercontractivity — which is a really useful tool in this area — and see a few basic applications.

§5.1 Hypercontractivity

Definition 5.1. Given a function $f\colon\{-1,1\}^n \to \mathbb{R}$, we define

$$\|f\|_p = \left(\mathbb{E}_x \left|f(x)\right|^p\right)^{1/p}.$$

(This is a standard definition of norm.)

What hypercontractivity is about is trying to compare the p-norms of a given function.

Fact 5.2 — For any function f, the p-norm is monotone in p — if $1 \le p \le q$, then $\|f\|_p \le \|f\|_q$.

(This is not hard to see, using e.g., Jensen's inequality.)

Remark 5.3. One way to remember the direction of the inequality is that $\|f\|_\infty$ is the maximum of $|f|$, while $\|f\|_1$ is the average of $|f|$. So $\|f\|_\infty \ge \|f\|_1$.

What hypercontractivity is about is trying to get inequalities in the *reverse* direction.

We can't *always* get an inequality in the reverse direction — that doesn't make any sense. But what we'll see is that inequalities hold in the reverse direction for a certain class of functions — which will be a really important class.

§5.1.1 Degree 1 functions

To get some intuition, we'll look at the simplest class of functions — degree-1 functions, i.e,. functions f of the form $f(x) = a + \sum_{i=1}^n a_i x_i$. For convenience we'll assume $a = 0$, so that f is just of the form $\sum a_i x_i$. And we'll normalize so that $\|f\|_2 = 1$, i.e., $\sum a_i^2 = 1$.

We can think of $f(x)$ as a sum of independent random variables (each of the coordinates of x are independent of each other). We know that $\mathbb{E} a_i x_i = 0$ for each i, and $\mathbb{E}(\sum a_i x_i)^2 = 1$.

In probability courses, we often want to prove things about limiting distributions (e.g., central limit theorems). Here, suppose that we have a sum of independent random variables, and each one of the individual variances a_i is small. Then the central limit theorem says that $f(x) \sim \mathcal{N}(0,1)$ — i.e., f behaves like a normal random variable. (This isn't exact — it depends on how big the a_i are — but it's an approximation.)

So for degree-1 functions, we literally know the *distribution* of $f(x)$, which means of course we can compute the norms — in this case, we end up with

$$\|f\|_{2q}^{2q} = \mathbb{E}_x \left|f(x)\right|^{2q} = \mathbb{E}\left|\mathcal{N}(0,1)\right|^{2q} + O(\varepsilon) = (2q-1)!! + O(\varepsilon)$$

(if we assume $|a_i| \le \varepsilon$ for all i; you can compute moments of the Gaussian by using integration by parts).

If we think of q as constant, we see that $\|f\|_q$ is still a constant. In particular,

$$\|f\|_{2q} \le O_q(\|f\|_2).$$

We stated this for the case where all the a_i are small, but we can actually get a similar statement in general.

Theorem 5.4 (Hypercontractivity for degree 1 functions)
If $f\colon\{-1,1\}^n \to \mathbb{R}$ has degree at most 1, then $\|f\|_4 \le \sqrt{3}\,\|f\|_2$. More generally for any $q \ge 2$, we have $\|f\|_q \le \sqrt{q-1}\,\|f\|_2$.

Remark 5.5. When we talk about hypercontractivity, sometimes it's most useful to think about 4-norms, rather than general q-norms; so we state that case separately.

Remark 5.6. Are there similar statements comparing general p and q instead of just 2 and q? The answer is yes; we might get to such statements later.

(We're not going to prove this, because we're going to prove something stronger.)

§5.1.2 The hypercontractivity inequality

All this intuition was for degree 1 functions; what happens if we have degree 2 functions? Now we no longer have a sum of *independent* random variables — we might have things like x_1x_2, x_1x_3, and x_2x_3 (which are not independent). So this intuition collapses.

But it turns out that you can still compute moments and get something similar.

Theorem 5.7 (Hypercontractivity for degree d functions)
For a function $f\colon\{-1,1\}^n \to \mathbb{R}$ of degree at most d, we have

$$\|f\|_q \le (q-1)^{d/2}\,\|f\|_2$$

for all $q \ge 2$.

This is the promised theorem; right now it looks a bit odd, and it's not clear why it's useful or how to use it. We'll clarify that later in the lecture; but before doing that, we'll give the proof (which is actually very simple; Dor comments that it is a 'zero knowledge proof' in the sense that it is correct but does not tell us anything).

Proof. We'll only do the proof for $q = 4$; the same proof generalizes immediately to any even q, and if we want to deal with other (or non-integer) q's, we have to do some sort of expansion.

We'll use induction on n and d. We'll skip the case $n = 1$ (which is the base case).

Now suppose we have a function $f(x_1, \ldots, x_n)$. We want to use induction, so what we want to do is express this in terms of functions of a smaller number of variables. So we'll take out the part of f that depends on x_n — we write

$$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_{n-1}) + x_n h(x_1, \ldots, x_{n-1}).$$

If we look at the Fourier expansion of f, it's clear we can do this — we write down f as a sum $\sum \widehat{f}(S)\chi_S(x)$, and separate the characters that don't contain n and those that do; and from the characters that do contain n, we just pull out x_n. Explicitly, we take $g = \sum_{S\not\ni n} \widehat{f}(S)\chi_S(x)$, and $h(x) = \sum_{S\ni n} \widehat{f}(S)\chi_{S\setminus\{n\}}(x)$ (which we named last time as $\delta_n f(x)$).

We have $\deg g \le d$ and $\deg h \le d-1$ (for h, we had characters of degree at most d, and then we removed x_n, which decreases their degree by 1).

Now we want to upper-bound $\mathbb{E}_x\,|f(x)|^4$; we just plug this into our identity and hope for the best. We have $f = g + x_n h$, so we get

$$\mathbb{E}_x\,|f(x)|^4 = \mathbb{E}_x\,|g(x) + x_n h(x)|^4 = \mathbb{E}_x\left[g(x)^4 + 4g(x)^2 x_n h(x) + 6g(x)^2 x_n^2 h(x)^2 + 4g(x)x_n^3 h(x)^3 + x_n^4 h(x)^4\right].$$

We can make this look a bit nicer by noting that some of these terms are 0 — specifically, we have $\mathbb{E}[g(x)^3 x_n h(x)] = 0$. To see this, note that g and h are only functions of the first $n-1$ variables, so we can separate this as $\mathbb{E}[g(x)^3h(x)]\mathbb{E}[x_n] = 0$ (since $\mathbb{E}[x_n] = 0$). Similarly, we can also get rid of the term with x_n^3, and we can replace x_n^2 and x_n^4 with 1. So now we only have three terms — specifically, we get

$$\mathbb{E}\left|f(x)\right|^4 = \mathbb{E}[g(x)^4] + 6\mathbb{E}[g(x)^2h(x)^2] + \mathbb{E}[h(x)^4].$$

Now there's no more terms we can get rid of, so there's only one thing we can do. First there's some 4-norms we can use induction on; but there's also this weird-looking term $g(x)^2h(x)^2$ that we can't induct on. So here we're going to use Cauchy–Schwarz to bound it by something that *is* inductible (it's not clear why this works, but it does) — we have

$$\mathbb{E}[g(x)^2h(x)^2] \leq \mathbb{E}[g(x)^4]\mathbb{E}[h(x)^4]^{1/2}.$$

And now we can express this as a bunch of 4-norms — we get

$$\mathbb{E}\left|f(x)\right|^4 \leq \left\|g\right\|_4^4 + 6\left\|g\right\|_4^2\left\|h\right\|_4^2 + \left\|h\right\|_4^4.$$

And now everything is inductible, so let's induct — using the fact that g and h are functions of $n-1$ variables, and that $\deg g \leq d$ and $\deg h \leq d-1$, we get

$$\mathbb{E}\left|f(x)\right|^4 \leq \sqrt{3}^{4d}\left\|g\right\|_2^4 + \sqrt{3}^{2d}\sqrt{3}^{2(d-1)}\cdot 6\left\|g\right\|_2^2\left\|h\right\|_2^2 + \sqrt{3}^{4(d-1)}\left\|h\right\|_4^4.$$

Now we're almost done — the whole point of the factor of $\sqrt{3}^{d-1}$ is to make the factor of 6 into a 2. So we get that this is at most

$$9^d(\left\|g\right\|_2^4 + 2\left\|g\right\|_2^2\left\|h\right\|_2^2 + \left\|h\right\|_4^4) = 9^d(\left\|g\right\|_2^2 + \left\|h\right\|_2^2)^2.$$

And finally, we have $\left\|g\right\|_2^2 + \left\|h\right\|_2^2 = \left\|f\right\|_2^2$; so we get that $\left\|f\right\|_4^4 = \mathbb{E}_x\left|f(x)\right|^4 \leq 9^d\left\|f\right\|_2^4$, which is exactly what we wanted. □

This result is pretty puzzling. First, what is this telling us mathematically. And from a linguistic point of view, why would you call this hypercontractivity — what is contracting?

§5.2 Noise operator formulation

Let's answer the linguistic question first. We'll now give a completely equivalent formulation, but in terms of operators and the way that they act on functions.

> **Definition 5.8.** For $x \in \{-1,1\}^n$ and $\rho \in [0,1]$, we define the distribution $T_\rho x$ as follows — to sample $y \sim T_\rho x$, for each coordinate $i \in [n]$ independently, we take
>
> $$y_i = \begin{cases} x_i & \text{with probability } \rho \\ \textsc{Unif}\{-1,1\} & \text{with probability } 1-\rho. \end{cases}$$

This defines some sort of noisy process on the Boolean cube. If you like probability you can view it as a Markov chain. And if you like linear algebra, you can view it as a linear operator — we can consider the linear operator $T_\rho\colon L_2(\{-1,1\}^n) \to L_2(\{-1,1\}^n)$ defined by $T_\rho f(x) = \mathbb{E}_{y\sim T_\rho x} f(y)$ (this is an 'averaging' operator — given f, we define $T_\rho f$ as the average of f over y sampled according to this Markov chain).

Now we can give an equivalent formulation of hypercontractivity for which the name makes sense.

First note that T_ρ is an averaging operator, and averaging makes things more smooth — for example, $\max\left|f\right|$ should become smaller when we apply noise. And if we play around with Jensen's inequality, this is true for *every* norm — we have $\left\|T_\rho f\right\|_p \leq \left\|f\right\|_p$ for all $p \geq 1$. So whenever we apply noise, our norms decrease. This is called *contraction*, which makes sense — it says T_ρ contracts functions.

And what *hypercontractivity* says is that it *really* contracts them.

Theorem 5.9 (Hypercontractivity)
For every function $f\colon\{-1,1\}^n \to \mathbb{R}$ and all $1 \le p \le q$ and $0 \le \rho \le \sqrt{(p-1)/(q-1)}$, we have

$$\|T_\rho f\|_q \le \|f\|_p\,.$$

So we've really managed to flip the direction of our inequality — if we look without the noise, then we always have $\|f\|_q \ge \|f\|_p$. But if we apply enough noise, then we can make the q-norm smaller.

Remark 5.10. When $\rho = 1$, our operator is just the identity (since $y_i = x_i$). When $\rho = 0$, there is no correlation — we're just picking a random y, so $T_\rho f$ is just a constant function whose value is $\mathbb{E}f$. So you can see these two extreme endpoints — $\rho = 1$ does nothing, and $\rho = 0$ is constant.

This is the source of the name hypercontractivity, and it's instructive to try to prove this for $q = 4$ and $p = 2$. Then the proof is the same as what we saw, if you stare at it; we need the following claim (about how T_ρ acts on characters).

Claim 5.11 — Suppose that $f\colon\{-1,1\}^n \to \mathbb{R}$ and $\rho \in [0,1]$. Then

$$T_\rho f(x) = \sum \rho^{|S|}\widehat{f}(S)\chi_S(x).$$

In other words, the Fourier expansion of $T_\rho f$ is the same as that of f, except that large characters get damped exponentially.

Proof. We have $T_\rho f = T_\rho \sum_S \widehat{f}(S)\chi_S$. And since T_ρ is linear, we can exchange the summation and operator to rewrite this as $T_\rho f = \sum_S \widehat{f}(S) T_\rho \chi_S(x)$. So now our goal is just to understand how T_ρ acts on characters. To do so, we have

$$T_\rho\chi_S(x) = \mathbb{E}_{y\sim T_\rho x}\chi_S(y) = \mathbb{E}_{y\sim T_\rho x}\prod_{i\in S} y_i.$$

Now since the y_i's are independent, we can write this as

$$\prod_{i\in S}\mathbb{E}_{y\sim T_\rho x} y_i.$$

And now if we look at our process, if $y_i = x_i$ (which happens with probability ρ) we simply get ρx_i. And otherwise (if the second case happens), then y_i is uniform so has expectation 0. So when the smoke clears, we get $\mathbb{E}[y_i] = \rho x_i$, and so

$$T_\rho\chi_S(x) = \prod_{i\in S}\rho x_i = \rho^{|S|}\prod_{i\in S} x_i = \rho^{|S|}\chi_S(x),$$

which is exactly what we wanted. □

Now we know why the theorem is called hypercontractivity, but we don't yet know why it's useful. In the rest of this lecture we'll see a few basic applications; next lecture we'll see the real applications that make it important (this actually relates to things we saw in the previous lecture).

§5.2.1 Application — small set expansion

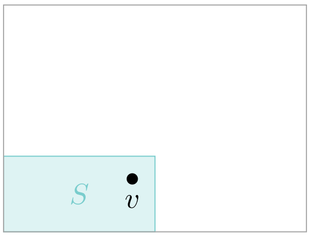
Edge expansion is a central notion in combinatorics and computer science, and this inequality has a very nice interpretation in terms of expansion of sets in the noisy hypercube.

Definition 5.12. If $G = (V, E, w)$ is a weighted regular graph and $S \subseteq V$ is a set of vertices, then the *edge expansion* of S, denoted by $\Phi(S)$, is defined as

$$\Phi(S) = \mathbb{P}_{u \in S, (u,v) \in E}[v \notin S].$$

By a weighted regular graph, we mean that we have weights on all the edges, and for each vertex the sum of weights at each vertex should be equal (e.g., all sums are 1).

So we have a graph and a set S, which we think of as small. And we sample a random vertex of S and take a step in the graph (we choose a random edge according to the edge weights); and we ask, what's the probability that we escape from S?



Often when we talk about edge expansion, we're interested in expansion graphs. That's somewhat related to what we'll talk about here; here we're interested in small set expansion, which intuitively asks the expansion of any small set to be very large.

Definition 5.13. We call G an (ε, δ)-small set expander (SSE) if for all $S \subseteq V$ of size $|S| \leq \delta |V|$, we have $\Phi(S) \geq 1 - \varepsilon$.

When people say a graph is a SSE, they really mean that there is a parameter tending to ∞ for the number of vertices, and they talk about a family of graphs; and then they mean that for every ε, we can pick sufficiently small δ for which this condition holds.

Definition 5.14. We say a family of graphs $\{G_n\}$ is a SSE if for every $\varepsilon > 0$, there exists $\delta > 0$ such that for all large n, we have that G_n is an (ε, δ)-SSE.

We've defined a Markov chain (or operator) T_ρ, which we can also view as a weighted graph; we call this the *noisy hypercube* with ρ. Here the weight of the edge xy is the probability that when we're at x, we go to y — so $V = \{-1, 1\}$, with

$$w(x, y) = \mathbb{P}_{y' \sim T_\rho x}[y' = y].$$

Theorem 5.15

The noisy hypercube with $\rho = 1/\sqrt{3}$ is a SSE.

Unlike the previous proof we saw today (which was disappointing or uninformative), this proof is really nice — and shows a nice trick in the area.

Proof. We have to prove that the noisy cube is a small-set expander; so let's consider a set $A \subseteq \{-1, 1\}^n$ with $\mu(A) \leq \delta$. Now we want to study the expansion of A. But we know inequalities about functions, so we need to move to function-land — so let's take $f = 1_A$ to be the indicator of A.

Now we want to relate the expansion of A to stuff that we can actually bound, so let's do that. It turns out that it's analytically easier to work with the *non-expansion* of A, namely

$$1 - \Phi(A) = \mathbb{P}_{a \in A, (a,v)}[v \in A].$$

(Here we're sampling $a \in A$ and a random edge, and looking at the probability that we *remain* inside A.)

We have a probability here, and we prefer to work with expectations instead of probabilities; so we can turn this into an expectaiton by writing this as

$$1 - \Phi(A) = \mathbb{E}_{a,v}[1_{v \in A}].$$

And then instead of sampling $a \in A$, we're going to sample a *uniformly* — we can write this as

$$\frac{1}{\mu(A)}\mathbb{E}_{a,v\sim T_\rho a}[1_{a\in A}1_{v\in A}] = \frac{1}{\mu(A)}\mathbb{E}_{a,v\sim T_\rho a}[f(a)f(v)]$$

(the factor of $1/\mu(A)$ corresponds to the probability that we're in A).

And now this expression is

$$1 - \Phi(A) = \frac{1}{\mu(A)}\mathbb{E}_a f(a) T_\rho f(a)$$

(note that $\mathbb{E}_{v\sim T_\rho a} f(v)$ is precisely $T_\rho f(a)$, by the definition of the noise operator). And this is exactly the inner product — so we get

$$1 - \Phi(A) = \frac{1}{\mu(A)}\langle f, T_\rho f\rangle.$$

Now we get to Dor's favorite trick in the proof. So far we haven't really done anything — we've just moved into analytic form. But the next move is a very nice trick popular whenever you apply hypercontractivity — we'll apply Hölder's inequality to get that

$$1 - \Phi(A) \le \frac{1}{\mu(A)}\left\lVert f\right\rVert_{4/3}\left\lVert T_\rho f\right\rVert_4.$$

(We do this to bring norms higher than 2 into the picture.)

The only reason we chose $\rho = \sqrt{3}$ is because $(2-1)/(4-1) = 1/3$, so this allows us to apply hypercontractivity; then we can bound $\left\lVert T_\rho f\right\rVert_4 \le \left\lVert f\right\rVert_2$, and we get that

$$1 - \Phi(A) \le \frac{1}{\mu(A)}\left\lVert f\right\rVert_{4/3}\left\lVert f\right\rVert_2.$$

And we can compute both of these norms — we have $\left\lVert f\right\rVert_2^2 = \mathbb{E}[f(x)^2] = \mathbb{E}[f(x)] = \mu(A)$ (since f is Boolean, so it doesn't matter whether we square it or not), so $\left\lVert f\right\rVert_2 = \mu(A)^{1/2}$. And similarly $\left\lVert f\right\rVert_{4/3} = \mu(A)^{3/4}$. So finally, we get that $1 - \Phi(A) \le \mu(A)^{1/4}$, and we're done (if A is small enough, then this is smaller than ε — this works because $3/4 + 1/2 > 1$). □

This is one application of hypercontractivity — often whenever you can prove these types of inequalities, you can also prove mixing-type things on your graph.

§5.3 Application — tail bounds

Theorem 5.16 (Concentration bound)

Suppose that $f\colon \{-1,1\}^n \to \mathbb{R}$ has degree at most d. Then for all $t \ge 1$, we have

$$\mathbb{P}_x[|f(x)| \ge t\left\lVert f\right\rVert_2] \le e^{-t^{2/d}/2}.$$

Intuitively, we typically expect $|f(x)|$ to be around $\left\lVert f\right\rVert_2$ (since $\mathbb{E}\,|f(x)|^2 = \left\lVert f\right\rVert_2^2$). So you can view this as saying that the probability f deviates a lot is exponentially decaying.

Remark 5.17. Note that for $d = 1$, we get the Chernoff bound; for larger d, this gives a generalization that's sometimes useful.

Proof. We're going to pick some $q \geq 1$ (which we'll decide on later) and raise this inequality to the qth power — so we're interested in

$$\mathbb{P}_x[|f(x)|^q \geq t^q \|f\|_2^q].$$

And now we just apply Markov and hope for the best — by Markov, this is at most

$$\frac{\mathbb{E}\,|f(x)|^q}{t^q \|f\|_2^q} = \frac{\|f\|_q^q}{t^q \|f\|_2^q}.$$

And now we can upper-bound $\|f\|_q^q$ using hypercontractivity — we get that this is at most

$$\frac{(q-1)^{qd/2}}{t^q} \cdot \frac{\|f\|_2^q}{\|f\|_2^q} = \frac{(q-1)^{qd/2}}{t^q}.$$

So now we have some function of q, which we can just optimize; taking $q = t^{2/d}/2$ gives the desired result. □

§5.4 Application — Anticoncentration

Qualitatively, this result is very informative — it tells us that if we look at a low-degree function, its values are not going to be very large. So it's 'almost bounded,' which is very useful intuition (and we're going to use this intuition quite a lot).

But it turns out the same tool also tells us that the values f takes cannot always be very close to 0.

Theorem 5.18 (Anti-concentration)
For all $\theta \in (0, 1)$ and f with $\deg f \leq d$, we have

$$\mathbb{P}_x[|f(x)| \geq \theta \|f\|] \geq \frac{(1-\theta^2)^2}{9^d}.$$

Proof. Let's look at $\|f\|_2^2 = \mathbb{E}\,|f(x)|^2$; and let's call the event $|f(x)| \geq \theta \|f\|$ as E. We'll split our expectation according to whether E occurs or not — we get that

$$\|f\|_2^2 = \mathbb{E}\,|f(x)|^2 \cdot 1_E + \mathbb{E}\,|f(x)|^2 \cdot 1_{\overline{E}}.$$

And now we can just upper-bound these — when $\overline{E}$ occurs, this means $|f(x)|$ is not too large, so we can bound

$$\mathbb{E}\,|f(x)|^2\, 1_{\overline{E}} \leq \theta^2 \|f\|_2^2\,.$$

For the first term, we don't know what to do, so we'll use Cauchy–Schwarz; this gives us

$$\mathbb{E}\,|f(x)|^2\, 1_E \leq \sqrt{\mathbb{E}\,|f(x)|^4 \cdot \mathbb{P}[E]}.$$

And now we have a 4-norm, which we can use hypercontractivity to bound; when we do this, we get

$$\mathbb{E}\,|f(x)|^2\, 1_E \leq 3^d \|f\|_2^2 \sqrt{\mathbb{P}[E]}.$$

Finally, we can finish the proof by shuffling things around — we get that

$$3^d \|f\|_2^2 \sqrt{\mathbb{P}[E]} \geq (1-\theta^2) \|f\|_2^2\,,$$

and cancelling out $\|f\|_2^2$ and squaring gives the desired result. □

> **Remark 5.19** (Summary)**.** The notes have one more application, which is cute — so we'll likely discuss it later. But today the point was to see a few basic applications of hypercontractivity.
>
> First we saw the hypercontractive inequality, whose proof was quite opaque. But then we saw it has some nice applications that do have intuition. We saw something about mixing in the noisy hypercube. And then we saw two bounds, kind of opposite each other — one gives an upper bound on the probability a function is very large, and one gives a lower bound on the probability that a function is not too small.

§6 February 27, 2024

Today we'll see some nice applications of hypercontractivity. We'll start with a nice technical trick; we'll then discuss two important theorems — the FKN theorem and KKN theorem.

§6.1 The 1-norm trick

We'll start with a cute technical trick. Recall that when we looked at hypercontractivity, it allowed us to say that if $1 < p \leq q$ and $\deg f \leq d$, then $\|f\|_q \leq C_{p,q}^d \|f\|_p$ for some constant $C_{p,q}$.

> **Question 6.1.** Can we get an inequality when $p = 1$?

It turns out that we can.

> **Lemma 6.2**
>
> Suppose that $f \colon \{-1,1\}^n \to \mathbb{R}$ is of degree at most d. Then
>
> $$\|f\|_2 \leq 3^d \|f\|_1 .$$

So we *can* take $p = 1$, though the constant changes a bit.

Proof. We have $\|f\|_2^2 = \mathbb{E}_x |f(x)|^2$. Now we're literally going to set up a Hölder inequality that gives us a 1-norm and something else — we have

$$\|f\|_2^2 = \mathbb{E}_x |f(x)|^2 = \mathbb{E}_x |f(x)|^{4/3} |f(x)|^{2/3} \leq (\mathbb{E} |f(x)|^4)^{1/3} (\mathbb{E} |f(x)|)^{2/3}.$$

(We choose the exponents so that Hölder gives us what we want.) Now we're essentially done — we get

$$\|f\|_2^2 \leq \|f\|_4^{4/3} \|f\|_1^{2/3} .$$

Now we can use standard hypercontractivity to reduce the first term to the 2-norm — we get

$$\|f\|_2^2 \leq (\sqrt{3}^d \|f\|_2)^{4/3} \|f\|_1^{2/3} .$$

Now moving the terms involving $\|f\|_2$ to one side, we end up with $\|f\|_2^{2/3} \leq \sqrt{3}^{4/3} \|f\|_1^{2/3}$, or equivalently $\|f\|_2 \leq 3^d \|f\|_1$. □

What we're really trying to say here is that sometimes it's useful to reduce to 1-norms — since these are the easiest to understand. This sort of completes the picture of what hypercontractivity looks like. Dor doesn't know of too many applications of this, but here's one.

Remark 6.3. What's the intuition for why this is useful? Well, why's it ever useful to go from a q-norm to a p-norm in general?

When Dor was studying these things, it wasn't at first clear to him. But then you see it time and time again, and you actually develop some sort of feeling for what tools work well with others. If you just stare at this inequality and try to prove stuff using it, you won't get very far. But somehow it's very powerful when you combine it with other stuff. (For example, using Hölder together with hypercontractivity is very standard and powerful, and we'll see it again today.)

Remark 6.4. Is it obvious that we should use 4/3 and 2/3? You could really use any power less than 1, though you'd get some other norm instead of the 4-norm (Dor chose these numbers because he likes using the 4-norm, but you could use any exponents and get some result).

Remark 6.5. Is the exponential in d necessary in some sense? Dor thinks that it is, but isn't sure.

Remark 6.6. How tight is this inequality? A few years ago Dor saw some papers where people tried to optimize these constants; the way they did this was using the noise form of hypercontractivity where the noise rate is a *complex* number (it turns out that you can make sense of this and it's useful). So this (i.e., the value of 3) probably isn't tight.

§6.2 The FKN theorem

Theorem 6.7

Let $f\colon\{-1,1\}^n \to \{-1,1\}$ be a function such that $\|f - f^{=1}\|_2^2 \le \varepsilon$. Then there exists $i \in [n]$ and a sign $b \in \{\pm 1\}$ such that $\|f - bx_i\|_2 = O(\varepsilon)$.

Here $f^{=1}$ denotes the degree-1 part of f — i.e., $f^{=1}(x) = \sum_{|S|=1} \widehat{f}(x)\chi_S(x) = \sum_{i=1}^n \widehat{f}(\{i\})x_i$. So the given condition essentially means that f is 'approximately' degree 1.

On the problem set, we already saw the case where $\varepsilon = 0$ — we proved that if f is a Boolean function which is also a degree-1 function, then it must be a function of the form bx_i (i.e., a dictatorship). This is a robust version of that statement — if we're *close* to a degree-1 function, then we have to be close to the answer in the exact case (i.e., a function bx_i).

Proof. Let $\ell(x) = \sum_{i=1}^n \widehat{f}(\{i\})x_i$ denote the degree-1 part of f. The intuition of the proof is that we know ℓ is close to f, and f is ± 1-valued, so f^2 is constant. So then ℓ^2 should also be close to constant.

But then the question is, what norm are we measuring the distance from a constant function in? And that's where hypercontractivity comes in.

First we'll consider $\ell(x)^2$. For simplicity, let $\widehat{f}(\{i\}) = a_i$, so that $\ell(x)^2 = \sum_i a_i^2 + 2\sum_{i<j} a_i a_j x_i x_j$. We're trying to show ℓ^2 is close to constant; here $\sum_i a_i^2$ is supposed to be the constant part, and we want to say that the remainder is very small. (We don't yet know this — this is what we want to prove.)

From this, we can compute $\mathrm{Var}(\ell^2)$ — the above equation is the Fourier expansion for $\ell(x)^2$, and by Parseval the variance is the sum of squares of the non-constant Fourier coefficients. So we get

$$\mathrm{Var}(\ell^2) = 4\sum_{i<j} a_i^2 a_j^2 = 2\left(\left(\sum_{i=1}^n a_i^2\right)^2 - \sum_{i=1}^n a_i^4\right).$$

Let's look at this to see that we're at least making some progress. First, $\sum_{i=1}^n a_i^2 = \|\ell\|_2^2$, which should be close to $\|f\|_2^2 = 1$. We want to show that $\mathrm{Var}(\ell^2)$ is very small; this will imply that $\sum_{i=1}^n a_i^4$ is also very close to 1. But if $\sum a_i^2$ and $\sum a_i^4$ are both very close to 1, the only way this is possible is if one of the a_i is itself very close to 1.

So now our goal is to upper-bound $\mathrm{Var}(\ell^2)$. But if we just expand, we get some fourth powers of ℓ and some awful stuff which we don't really want to handle, since we only really like dealing with 2-norms. First, by definition we have

$$\mathrm{Var}(\ell^2) = \|\ell^2 - \mathbb{E}\ell^2\|_2^2.$$

Let's call $\ell^2 - \mathbb{E}\ell^2 = h$, so that $\mathrm{Var}(\ell^2) = \|h\|_2^2$. We don't want to expand this out — that'd give us 4-norms, which we don't want to deal with. So we'll instead switch from the 2-norm of h to the 1-norm of h. To do this, we need to ensure we're dealing with a low-degree function — and $h = \ell^2 - \mathbb{E}\ell^2$ has degree 2, so by the 1-norm trick we have

$$\mathrm{Var}(\ell^2) = \|h\|_2^2 \le (8\,\|h\|_1)^2.$$

So we've managed to control $\mathrm{Var}(\ell^2)$ by $\|h\|_1$, and that's something we can actually compute — we have

$$\|h\|_1 = \|\ell^2 - \mathbb{E}\ell^2\|_1 \le \|\ell^2 - f^2\|_1 + \|f^2 - \mathbb{E}\ell^2\|_1$$

(by the triangle inequality). And now we can bound each of these terms separately — f^2 is just the constant 1, and $\mathbb{E}\ell^2 = \|f^{=1}\|_2^2 \ge (1-\sqrt{\varepsilon})^2 \ge 1 - O(\sqrt{\varepsilon})$ (by the given assumption). So the second term is small — i.e.,

$$\|f^2 - \mathbb{E}\ell^2\|_1 \le O(\sqrt{\varepsilon}).$$

Now we need to estimate the first term. And for this, we can return to high school; we have a difference of squares, which we can write as

$$\|\ell^2 - f^2\|_1 \le \|(\ell - f)(\ell + f)\|_1\,.$$

And the point of this move is that we know something about $\ell - f$ — that's exactly what we know has small 2-norm. So we'll use Cauchy–Schwarz to get

$$\|\ell^2 - f^2\|_1 \le \|(\ell - f)(\ell + f)\|_1 \le \|\ell - f\|_2\,\|\ell + f\|_2\,.$$

> **Remark 6.8.** You can imagine what'd happen if we tried doing everything with 2-norms instead of 1-norms; one thing that'd break is that here we'd have 4-norms instead of 2-norms, which we don't know how to deal with.

By assumption $\|\ell - f\|_2 \le \sqrt{\varepsilon}$, while $\|\ell + f\|_2 \le 2$. So overall, we get that

$$\|h\|_1 \le 2\sqrt{\varepsilon} + O(\sqrt{\varepsilon}) = O(\sqrt{\varepsilon}).$$

Now we just have to combine our observations, and then we're done. If we plug in our bound for $\|h\|_1$ into our formula for $\mathrm{Var}(\ell^2)$, we get that

$$\mathrm{Var}(\ell^2) \le (9\,\|h\|_1)^2 \le O(\varepsilon).$$

And our whole goal was to prove that $\mathrm{Var}(\ell^2)$ was small, so now we're essentially done — we conclude that $(\sum a_i^2)^2 - \sum a_i^4 \le O(\varepsilon)$, and moving the fourth powers to the other side, we get that $\sum a_i^4 \ge (\sum a_i^2)^2 - O(\varepsilon)$. As mentioned earlier, we have $\sum a_i^2 = \mathbb{E}|f^{=1}|^2 \ge 1 - O(\varepsilon)$. So overall, we get that $\sum a_i^4 \ge 1 - O(\varepsilon)$.

And now we have

$$1 - O(\varepsilon) \le \sum a_i^4 \le \max a_i^2 \sum a_i^2 \le \max a_i^2.$$

So there exists i such that $|a_i| \ge 1 - O(\varepsilon)$. And now we're done (we can finish using a similar argument as in the BLR test). □

Remark 6.9. Here's a corollary of this — recall that earlier we proved that for every balanced function $f\colon\{-1,1\}^n \to \{-1,1\}$ (*balanced* means $\mathbb{E}f = 0$), there is some i with $I_i[f] \geq 1/n$. The proof was essentially that

$$I[f] = \sum |S| \widehat{f}(S)^2 \geq \sum \widehat{f}(S)^2 = 1,$$

so some influence is at least $1/n$. The case where this is tight is when all the mass is on level 1 (otherwise we'd get a lower bound of 2 on $|S|$).

And in this theorem, we've shown that in that case, we actually have a bound much better than $1/n$ — we have a variable with huge influence, close to 1.

If you combine them, you should be able to show that $I_i[f] \geq (1+c)/n$ for some $c > 0$. (This is not very important, because we'll show something much stronger today; but it's nice intuition — where we look at an argument, look when the argument is tight, and see what actually happens in those cases — and here we actually get much better influences.)

Remark 6.10. What's special about degree 1? The answer is that in the exact case, only dictatorships are Boolean functions of degree 1. There also exist theorems for functions with degree 2 and 3, but they're harder. And there's also open questions when you go beyond the uniform Boolean cube.

§6.3 The Fourier spectrum of small sets

Suppose that $S \subseteq \{-1,1\}^n$ has size $\delta \cdot 2^n$, where we think of $\delta > 0$ as small. We'll then consider its indicator function $1_S\colon\{-1,1\}^n \to \{0,1\}$ as a Boolean function; and we can consider the Fourier coefficients of this function.

Lemma 6.11

Suppose that $S \subseteq \{-1,1\}^n$ has size $\delta \cdot 2^n$. Then $\deg(1_S) \geq \Omega(\log(1/\delta))$.

The reason we're talking about this lemma is that we're building up towards the KKL theorem (a lower bound on influences). When we talk about influences, it makes a lot of sense to talk about small sets — the reason is that you look at the set of points x at which coordinate i is influential, and if we're talking about functions with small influence, these sets will be small. So if we can say meaningful stuff about them, then maybe we can say stuff about f. So for the KKL theorem we'd like to say stuff about small sets; and we're starting with this lemma.

This lemma has two proofs. One sort of nukes it, but its benefit is that it tells you more. (This is a sort of weak statement — it tells you there is some character of high degree, but doesn't tell you anything quantitative.) So here's the overkill proof.

Proof. Let $d = \deg(1_S)$. We're going to use Hölder plus hypercontractivity again — we have

$$\delta = \|1_S\|_2^2 = \langle 1_S, 1_S\rangle \leq \|1_S\|_4 \, \|1_S\|_{4/3}$$

by Hölder. Now we use hypercontractivity to reduce the 4-norm to the 2-norm, giving

$$\delta \leq \sqrt{3}^d \, \|1_S\|_2 \, \|1_S\|_{4/3}.$$

And now these are some norms that we can compute — specifically, $\|1_S\|_2 = \sqrt{\delta}$ and $\|1_S\|_{4/3} = \delta^{3/4}$ (this is because 1_S is literally a Boolean function, so we can control any norm that we want). This gives

$$\delta \leq \sqrt{3}^d \delta^{5/4}.$$

And the whole point of this exercise is that $5/4 > 1$, so we get $\sqrt{3}^d \geq \delta^{-1/4}$ (the value of $1/4$ isn't important), and therefore $d \geq \Omega(\log(1/d))$. □

There are several things that are very unsatisfying about this proof. First, it seems like we used a very big hammer to prove something very weak. And secondly, it seems this is not the right conclusion — we could have aimed for something better. (The lemma itself is not too hard — you can prove it by Schwarz–Zippel, which basically tells you that a degree-d function is equal to 0 on at most a $(1-2^d)$-fraction of points.)

Still, we saw this overkill proof because we're now going to strengthen the statement considerably, but with virtually the same proof.

The way to read the previous lemma is that if we have an indicator function of a small set, then it has *some* mass on high levels. We're now going to show that actually, almost *all* the mass lies on high levels. (For technical reasons, we'll consider functions that are $\{0,1,-1\}$-valued instead of just $\{0,1\}$-valued.)

Lemma 6.12

Let $f\colon\{-1,1\}^n \to \{0,1,-1\}$ be a function such that $\mathbb{P}_x[f(x) \neq 0] = \delta$. Then $\sum_{|S|\leq \log(1/\delta)/20} \widehat{f}(S)^2 \leq \delta^{24/20}$.

This states that if we look at the Fourier mass of f on sets which are 'small' — i.e., at most a constant times $\log(1/\delta)$ — then this mass is small. Note that the sum of squares of *all* Fourier coefficients is $\sum_S \widehat{f}(S)^2 = \|f\|_2^2 = \mathbb{E}\,|f(x)|^2 = \mathbb{P}[f(x) \neq 0] = \delta$. But this sum is δ to the power of something larger than 1. So if we think of δ as small, then this is something much smaller than δ (which is the point of the lemma).

Proof. The proof will be almost the same. Let $d = \frac{1}{20}\log(1/\delta)$, and define $f^{\leq d}(x) = \sum_{|S|\leq d} \widehat{f}(S)\chi_S(x)$ (this is the part of f of degree at most d). Then the left-hand side of what we're trying to prove is $\|f^{\leq d}\|_2^2$ by Parseval. So we've expressed what we want to bound as the 2-norm of some function.

Now we're going to use the same Hölder trick — we have

$$\text{LHS} = \langle f^{\leq d}, f^{\leq d}\rangle \leq \|f^{\leq d}\|_4 \|f^{\leq d}\|_{4/3}.$$

We could use hypercontractivity, but we'd have a problem — we have a 4/3-norm, and the rule of thumb in Fourier analysis is that if you're looking at a norm smaller than 2, then it had better be of a Boolean function. Previously, we could compute the 4/3-norm because we had a Boolean function, but here $f^{\leq d}$ is *not* a Boolean function. (The only way we know how to bound this is by the 2-norm, and that won't work.)

So the solution is to be kind of clever — to anticipate this issue and do something about it. We instead write

$$\langle f^{\leq d}, f^{\leq d}\rangle = \langle f^{\leq d}, f\rangle$$

(one way to see this is that $\langle f^{\leq d}, f^{>}d\rangle = 0$ because the Fourier transforms of these functions are disjoint, so by Plancherel their inner product is 0). This is a standard move one does — add something which is 0 to ensure that one of the sides in your inner product is Boolean, and then you can continue the problem as usual. So then we can apply Hölder and then hypercontractivity to get

$$\text{LHS} \leq \|f^{\leq d}\|_4 \|f\|_{4/3} \leq \sqrt{3}^d \|f^{\leq d}\|_2 \, \|f\|_{4/3} \, .$$

Now earlier we said that when we had the 4/3-norm of $f^{\leq d}$, there's nothing we can do about it. There's only one norm where we really can relate $f^{\leq d}$ and f, and that's 2-norms — we know that $\|f^{\leq d}\|_2 \leq \|f\|_2$ by Parseval (since $\|f\|_2^2 = \|f^{\leq d}\|_2^2 + \|f^{>d}\|_2^2$). So now we get

$$\|f^{\leq d}\|_2^2 \leq \sqrt{3}^d \|f\|_2 \|f\|_{4/3} = 3^{d/2}\delta^{1/2}\delta^{3/4} = 3^{d/2}\delta^{5/4} < \delta^{24/20}$$

(the point is that we can take the constant $1/20$ in our definition of d to be small enough, so that $3^{d/2}$ is small — and the term $\delta^{5/4}$ is less than δ, so we'll still end up with something less than δ). □

§6.4 The KKL Theorem

Now we have all the setup we need to prove the KKL theorem — the first major theorem we'll see, which is very influential in more than one sense.

Theorem 6.13 (KKL)

Suppose that $f\colon \{-1,1\}^n \to \{-1,1\}$ has $I[f] \leq k \operatorname{Var}(f)$. Then there exists $I \in [n]$ such that $I_i[f] \geq 2^{-O(k)}$.

This says that if we have a function whose total influence is small, then there's actually a single variable that has a lot of influence.

> **Remark 6.14.** After talking about the FKN theorem, we talked about what happens when teh simple proof with $1/n$ is almost tight. This is something similar, but much more powerful — here the simple proof is off by a factor of k, and we're still able to prove one variable has a lot of influence.

Proof. Let C be some absolute large constant. We'll assume that all the influences are small —- i.e., $I_i[f] \leq 2^{-Ck}$ for all $i \in [n]$. Let $g_i(x) = \partial_i f(x)$.

Now our mindset is that we know all the influences are small; and the way to connect the notion of small influences with small sets is by looking at the derivatives, because we know that

$$\mathbb{P}[g_i(x) \neq 0] = I_i[f],$$

which is small. So you can think of g_i as a small set (but it's ±1-valued, so it's actually a function that fits the previous lemma).

First, informally what's going to happen is — well, we have g_i, which is $\{0, \pm 1\}$-valued, and we know that it's rarely nonzero. So therefore it must be a very high-degree function.

But how can this be the case? All the derivatives are high-degree, but the function itself is low-degree (since the simple bound on influence corresponds to degree). These don't add up; and that'll be the content of the proof.

Now we'll write this more formally. By the previous lemma, we have $\sum_{|S| \leq Ck/20} \widehat{g}_i(s)^2 \leq I_i[f]^{24/20}$ (where Ck is the log of the bound, and 20 comes from the previous lemma).

This tells us about the Fourier coefficients of g, but we want to learn about f. So how can we wrewrite this sum in terms of the Fourier coefficients of f?

We know that the Fourier coefficients of f and g_i are the same, except that we only keep the ones that contain i; so we get that

$$sum_{\substack{i \in S \\ |S| \leq Ck/20}} \widehat{f}(S)^2 \leq I_i[f]^{24/20}.$$

Now summing over all $i \in [n]$, on the left-hand side we're summing over all the small Fourier coefficients, and we end up with

$$\sum_{|S| \leq Ck/20} |S| \, \widehat{f}(S)^2 \leq \sum_{i=1}^n I_i[f]^{24/20} \leq \max I_i[f]^{4/20} \sum_{i=1}^n I_i[f]$$

(the point is that $24/20 > 1$, so we can pull out the leftover). And we assumed that the total influence of f is small and that each individual influence is small, so this is at most $2^{-Ck/5} \cdot k \cdot \operatorname{Var}(f)$.

Now clearing the smoke, we'll first drop $|S|$ (this only decreases the left-hand side); then we get

$$\sum_{|S| \leq Ck/20} \widehat{f}(S)^2 \leq 2^{-Ck/5} k \operatorname{Var}(f) \leq 2^{-Ck/10} \operatorname{Var}(f).$$

We've done a bunch of technical maneuvers that were well-motivated, but now we need to interpret this. If we look at the sum of squares of *all* nonempty Fourier coefficients, we get $\operatorname{Var}(f)$. This mass has to come from *somewhere*. And this inequality tells us not to look at the lower levels — there's nothing there. So this means the mass must be on the higher levels.

But we know that the average size of the Fourier coefficients can't be too large (because otherwise $I[f]$ would be large); so this is how we're going to finish the proof.

Let's ask ourselves how much mass is on the high levels — so we consider $\sum_{|S|\geq Ck/20} \widehat{f}(S)^2$. And we know that the average character size according to this distribution is k, so by Markov it can't be more than $Ck/20$ with probability more than $20/C$. We'll write this slightly differently — we have

$$\sum_{|S|\geq Ck/20} \widehat{f}(S)^2 \leq \frac{\sum_{|S|\geq Ck/20} |S|\,\widehat{f}(S)^2}{Ck/20} \leq \frac{\sum_S |S|\,\widehat{f}(S)^2}{Ck/20} = \frac{I[f]}{Ck/20} \leq \frac{20\operatorname{Var}(f)}{C}$$

(since $I[f] \leq k\operatorname{Var}(f)$). Think of C as big (e.g. 100).

So we've gotten that there's very little mass on the low levels, and there's very little mass on the high levels. But the mass must be somewhere, so this is a contradiction — explicitly, combining these two inequalities, we get that

$$\operatorname{Var}(f) = \sum_{|S|\neq 0} \widehat{f}(S)^2 \leq (2^{-Ck/10} + 20/C)\operatorname{Var}(f) < \operatorname{Var}(f),$$

which is a contradiction (if C is large enough, e.g., $C = 40$ should be good enough). □

Remark 6.15. Why was $2^{-O(k)}$ important? The reason is that when we handle the low-degree part, we get a log (in the bound on degree that we are able to go up to). So if you want a handle on this that is linear in k (which we need in order to do a bound on the high-degree part), we need to start with something that's exponential in k.

Remark 6.16. What motivated the idea of bounding the variance?

The initial motivation is that we know the influences are small, and we want to express this in some way. We know this tells us that the low-degree parts of g dont' have a lot of influence. And we used this to conclude something about the low-degree part of f.

The shift from looking at this quantity and the one where we drop $|S|$ and look at variance-like things is not very well-motivated, but Dor doesn't know how to use the extra $|S|$.

Interpreting total influence as average degree adn doing Markov-type bounds is also well-motivated. Once you stare at these two things and try to make sense of it, you realize it makes sense to drop the $|S|$. (The constant term doesn't contribute anything anywhere, so eventually you have to look at the variance.)

As a sidenote, this is not what is traditionally called the KKL theorem; that's the following statement.

Theorem 6.17 (KKL)
For all $f\colon \{-1,1\}^n \to \{-1,1\}$, we have $\max I_i[f] \geq \frac{\log n}{n}\operatorname{Var}(f)$.

The proof boils down to staring at the previous theorem, and taking $k = O(\log n)$ (you win in either case).

Nicely, this theorem is tight (we'll see the example next time).

§7 February 29, 2024

Last time, we showed a few nice applications of hypercontractivity — the main one was the KKL theorem. Today we're going to see a few more theorems that strengthen the KKL theorem in several ways. But before that, we'll see an important example of a function in this area, which also shows that the KKL theorem is tight (the $\log n$ factor is really all that we can gain).

§7.1 The tribes function

We'll define the function in $\{0, 1\}$ notation (though it can be easily converted to $\{-1, 1\}$).

> **Definition 7.1.** Take subsets $I_1, \ldots, I_k \subseteq [n]$ which are pairwise disjoint and each have size ℓ. We then define the function $f\colon \{0, 1\}^n \to \{0, 1\}$ as $f(x) = \bigvee_{i=1}^k \bigcap_{j \in I_i} x_i$.

The idea is that we imagine each subset as being a tribe, and f as a voting scheme; the outcome ends up being 1 if there is a tribe in which everyone agrees the outcome should be 1, and 0 otherwise.

Let's play around with this function and see what we get.

> **Question 7.2.** What is $\mathbb{P}_x[f(x) = 0]$?

We have $f(x) = 0$ when each one of the tribes says 0; since the tribes are disjoint, these events are independent, so $\mathbb{P}[f(x) = 0] = \prod_{i=1}^k \mathbb{P}[\bigwedge_{j \in I_i} x_j = 0] = \prod_{i=1}^k (1 - 1/2^\ell)$. We don't care about the precise numbers — we want this to be bounded away from both 0 and 1 — so we'll now perform an approximation — we have $1 - 1/2^\ell \approx e^{-1/2^\ell}$, so then

$$\mathbb{P}[f(x) = 0] \approx e^{-k/2^\ell}.$$

More precisely, a long as $k = \Theta(2^\ell)$, we have $0 \ll \mathbb{E} f \ll 1$. (We want to make sure f has constant variance, so that we don't have to worry about the variance term in KKL.)

We want to have k sets of size ℓ inside $[n]$; the best we can do is $k\ell \leq n$. So we want k roughly exponential in ℓ, and $k\ell \leq n$; this means we have to pick $k = n/\log n$ and $\ell = \log n - \log\log n$. So this is a setting of parameters that gives us that f is relatively balanced, but where this function is as spread as possible on as many variables as possible. So this is going to be our setting of parameters.

Now we have a function which is balanced, and our goal is to compute influences. So let's take any $j \in [n]$. First, the influence of each variable is the same — the tribes are all of the same size, so everything is symmetric — so we can assume $j \in I_1$.

Then what's the influence of j? We want to sample an input x, and we imagine that $f(x) = 0$, but if we flip the jth coordinate then f becomes 1. This means that all the tribes except the first must output 0 — otherwise the jth coordinate wouldn't matter. But we also need that for all $\ell' \in I_1 \setminus \{j\}$ we should have $x_{\ell'} = 1$ (so that the outcome of that tribe is up to j). This means $I_j[f]$ is the probability that for all $i \geq 2$ we have $\bigwedge_{\ell \in I} x_\ell = 0$, and for all $\ell' \in I_1 \setminus \{j\}$ we have $x_{\ell'} = 1$. By independence, we can again write this as

$$I_j[f] = \prod_{i \geq 2} \mathbb{P}[\bigwedge x_\ell = 0] \mathbb{P}[\text{for all } \ell \in I_1 \setminus \{j\},\, x_{\ell'} = 1].$$

The first term is constant (by the same computation as before), and the second term is $1/2^{\ell-1} = \Theta(\log n/n)$.

> **Remark 7.3.** This is an important function — often when you have a conjecture on Boolean functions you'll want to try it on several functions, and this is one of them.

So this shows that the KKL theorem is tight. Still, even though it's tight, we're now going to try to improve it — of course we can't improve it quantitatively, but we'll say something more extensive.

The KKL theorem says that if total influence is small, then there is some variable whose influence is large. But this isn't an if and only if; we'll try to get something that's closer to one.

§7.2 Talagrand's theorem

Theorem 7.4 (Talagrand)
Let $f\colon \{-1,1\}^n \to \{-1,1\}$. Then for some constant c, we have

$$\sum \frac{I_i[f]}{1+\log(1/I_i[f])} \geq c\,\mathrm{Var}(f).$$

So here we're looking at the sum of influences, but we *penalize* small ones (by dividing by these logs). Earlier we saw Poincare's inequality, which showed that $\sum I_i[f] \geq \mathrm{Var}(f)$. Talagrand's theorem says this remains true even with this penalty (up to a constant).

Remark 7.5. The extra $+1$ in the denominators is just to avoid division by 0; it doesn't really matter.

Something very nice about this theorem is that it implies both the KKL theorem and Poincare's inequality — for Poincare we literally say that the denominators are at least 1; the fact that it implies KKL is left to us (but not difficult).

Proof. The proof starts with magic, and then uses a reworking of the KKL proof. We'll start with the magic — we have

$$\mathrm{Var}(f) = \sum_{S\neq\emptyset} \widehat{f}(S)^2 = \sum_{i=1}^{n}\sum_{S\ni i} \frac{1}{|S|}\widehat{f}(S)^2.$$

This breaks the variance into the contributions of each one of the coordinates. (With the $1/|S|$ term, this is an exact equality because each Fourier coefficient is counted $|S|$ times.)

Remark 7.6. There's a natural way to arrive at this funny expression using stochastic calculus, so one can make more sense of this step if one knows more.

Now we stare at this and just do the magic of KKL — define $g_i\colon \{-1,1\}^n \to \mathbb{R}$ as

$$g_i(x) = \sum_{S\ni i} \frac{1}{\sqrt{S}}\widehat{f}(S)\chi_S(x)$$

(this is the function whose Fourier coefficients correspond to the ith term in the above expression), so that

$$\mathrm{Var}(f) = \sum_{i=1}^{n} \|g_i\|_2^2\,.$$

The way you can think about g_i is that it's something that 'looks' like the derivative of f (because we only look at characters containing i), but there's some penalization of high-degree terms. (If we didn't have the $1/\sqrt{|S|}$ term, then we'd have exactly the derivative.)

Our goal is to upper-bound $\mathrm{Var}(f)$; we'll do this by upper-bounding $\|g_i\|_2^2$ for each i (by the corresponding term on the left-hand side). Fix i, and let $d_i = \frac{1}{20}\log(1/I_i[f])$. Then we have

$$\|g_i\|_2^2 = \sum_{S\ni i}\frac{1}{|S|}\widehat{f}(S)^2 = \sum_{\substack{S\ni i\\ |S|\leq d_i}} \frac{1}{|S|}\widehat{f}(S)^2 + \sum_{\substack{S\ni i\\ |S|> d_i}} \frac{1}{|S|}\widehat{f}(S)^2$$

(we split our sum into high-degree and low-degree terms).

Let's start with the second sum (which is easier). In the second sum, $|S|$ is always big, so $1/|S|$ is always small; then we can pull it out and we get

$$\sum_{\substack{S\ni i\\|S|>d_i}}\frac{1}{|S|}\widehat{f}(S)^2\leq\frac{1}{d_i}\sum_{\substack{S\ni i\\|S|>d_i}}\widehat{f}(S)^2\leq\frac{1}{d_i}\sum_{S\ni i}\widehat{f}(S)^2\leq\frac{I_i[f]}{d_i},$$

which is exactly what we wanted.

For the first term, we're not so lucky; the $1/|S|$ doesn't buy us much, and in fact we're just going to give it up — we'll upper bound the first term (1) by

$$(1)\leq\sum_{\substack{|S|\leq d_i\\S\ni i}}\widehat{f}(S)^2.$$

And now let's consider the derivative $\partial_i f$. We're only considering sets S containing i, so this sum is *exactly*

$$\sum_{|S|\leq d_i}\widehat{\partial_i f}(S)^2.$$

(The Fourier coefficients of the derivative are the same as those of f, but we only keep the ones that contain i.) And what's nice about this? Last time, the main topic we discussed was that if we look at small sets (or rather their indicators — and the derivative of a Boolean function with small influence is kind of an indicator function of a small set, though it's $\{0,1,-1\}$-valued), then the mass on small levels is small. And this funny number $1/20$ is exactly from that; so we can just apply the lemma from last time, which told us that the total Fourier mass on low levels (where 'low' is defined by d_i) is at most

$$\sum_{|S|\leq d_i}\widehat{\partial_i f}(S)^2\leq I_i[f]^{6/5}=O\left(\frac{I_i[f]}{\log(1/I_i[f])}\right).$$

(Here we even gained a polynomial factor, and all we needed was a $1/\log$ factor.)

To recap, we split $\|g_i\|_2^2$ into high levels and low levels, and dealt with each separately. In the high levels, the penalizing factor itself was good enough. For the low levels, we're looking at the low-degree part of a small set, and we could just use the lemma from last time.

Combining these, we get that

$$\|g_i\|_2^2=O\left(\frac{I_i[f]}{\log(1/I_i[f])}\right)$$

for each i, and we're done. □

Remark 7.7. Could we do better with larger d_i? What'd break is that we can say things up to levels roughly $\log 1/\mu(S)$ in the lemma from last time, but not beyond that. (And if you look at this lemma and plug in the tribes function, it is again tight.)

§7.3 Friedgut's theorem

The Talagrand theorem is a strengthening of KKL; it's nice, but it doesn't tell us something too new (it's hard to say quickly how this is different in applications from KKL). But now we're going to see another theorem; the proof is actually the same, but the theorem tells you something different. It's not an if-and-only-if, but it's as close as we're going to get.

Theorem 7.8 (Friedgut)

Suppose that $f\colon \{-1,1\}^n \to \{-1,1\}$ has $\mathrm{Var}(f) \geq \Omega(1)$ and $I[f] \leq k$. Then for all $\varepsilon > 0$, there exists $J \subseteq [n]$ of size $2^{O(k/\varepsilon)}$ such that f is ε-close to a J-junta.

The way you should read this is that we have a function with small total influence. KKL told us that there's an influential variable. This theorem tells you that all those influential variables basically determine the function f. (This may be true even without the constant-variance assumption, but in that case the result won't be interesting, and it's inconvenient to deal with that case.)

So this theorem tells you that if the total influence is constant, e.g., 100, then your function basically just depends on a constant number of variables. This is stronger than KKL. In particular, it implies KKL — if we look at the J-junta then there's a variable with influence at least $1/\lvert J\rvert$.

Proof. Take $C > 0$ to be a large absolute constant. We begin by defining what our candidate set J is. We have variables and influences, and if a variable has high influence then we probably want to take it, so let's do that — we define

$$J = \{i \in [n] \mid I_i[f] \geq 2^{-Ck/\varepsilon}\}.$$

We have to prove two things — that J is not too big, and that f is close to a J-junta.

Let's start with proving an upper bound on $\lvert J\rvert$. The total influence of f is $I[f] \leq k$ by assumption, but it's certainly at least

$$I[f] \geq \sum_{i\in J} I_i[f] \geq \lvert J\rvert \cdot 2^{-Ck/\varepsilon}.$$

Rearranging this gives $\lvert J\rvert \leq k \cdot 2^{Ck/\varepsilon} \leq 2^{(C+1)k/\varepsilon}$. So J is not too large.

Remark 7.9. Can we strengthen this to saying that f is ε-close to a J-junta with ±1 outputs? Yes, this is what we are going to show — we'll start with a candidate that doesn't have ±1 outputs, and then round it.

Now we need to show f is ε-close to a J-junta. First, we'll define a candidate — we simply just look at the Fourier coefficients completely contained in J. So we define $G\colon \{-1,1\}^n \to \mathbb{R}$ as

$$G(x) = \sum_{S\subseteq J} \widehat{f}(S)\chi_S(x).$$

(This is the natural candidate — if we're a J-junta then we had better only have Fourier coefficients contained in J.) And then we can round it — we define $H(x) = \mathrm{sgn}(G(x))$.

First, we claim that G is a J-junta, and therefore H is. To see that G is a J-junta, all the characters are contained in J, so if we change a variable outside J, then nothing changes. So clearly G is a J-junta, and therefore H is also a J-junta.

Now we want to prove that f and H are close; to do so, we claim that

$$\lVert f - H\rVert_2^2 \leq 4\lVert f - G\rVert_2^2.$$

To see this, let's look at any x where f and H differ — for example, let's say $f(x) = 1$ and $H(x) = -1$. This tells us that $f(x) = 1$ and $G(x) < 0$; so $\lvert f(x) - G(x)\rvert \geq 1$. So for every nonzero contribution to the left-hand side (which has value $2^2 = 4$), we get a contribution of at least 1 to the right-hand side.

And the reason we work with G is because we have its Fourier transform, and Fourier transforms behave nicely with 2-norms. So we'll now bound $\lVert f - G\rVert_2^2$.

We have the Fourier transform of f, and the Fourier transform of G is literally part of it; so $f - G$ is literally the other part of it, and we get

$$\|f - G\|_2^2 = \left\|\sum_{S \not\subseteq J} \widehat{f}(S)\chi_S(x)\right\|_2^2 = \sum_{S \not\subseteq J} \widehat{f}(S)^2$$

(by Parseval). As we've done many times, we're now going to split this sum into high-degree and low-degree parts, and handle each separately — we write this as

$$\sum_{\substack{S \not\subseteq J \\ |S| \leq k/\varepsilon}} \widehat{f}(S)^2 + \sum_{\substack{S \not\subseteq J \\ |S| > k/\varepsilon}} \widehat{f}(S)^2$$

(remember that k is the total influence).

Now we'll bound each of these terms. We'll call these two terms (1) and (2), and we'll first bound (2). For (2), we know that

$$\sum |S| \, \widehat{f}(S)^2 = I[f] \leq k.$$

And we can get a lower bound by only keeping the large sets; so we get

$$\frac{k}{\varepsilon} \sum_{|S| \geq k/\varepsilon} |f| \, (S)^2 \sum_{|S| \geq k/\varepsilon} |S| \, \widehat{f}(S)^2 \leq k.$$

(This is standard Markov — the total influence corresponds to average degree, and the average degree is small, so the total mass on things of high degree has to be small.) When we rearrange, we end up with $(2) \leq \varepsilon$.

Now we need to bound (1). This is harder; a general rule of thumb is trying to bound each Fourier coefficient individually almost never works (Dor knows of exactly one example where it works, which we will see on a problem set). So somehow we need to collectively use the fact that $S \not\subseteq J$.

SO what we're going to do is write

$$(1) \leq \sum_{j \notin J} \sum_{\substack{S \ni j \\ S \leq k/\varepsilon}} \widehat{f}(S)^2.$$

This is clearly an upper bound, because we're going to cover each $S \not\subseteq J$ (it contains at least one element outside J). And what's the point of this move? The point is that we're going towards derivatives again — we can write this as

$$\sum_{j \notin J} \sum_{|S| \leq k/\varepsilon} \widehat{\partial_j f}(S)^2$$

(because Fourier coefficients of the derivative are the same as those of f, except that we only keep those containing j).

Again, j is outside the junta, so its influence is small; this means we can say this derivative has very small mass on low levels. And we've exactly arranged that this degree here is $\log 1/I_j[f]$ (up to a constant). So by the lemma from last lecture, we again get that this is at most

$$\sum_{j \in J} I_j[f]^{6/5}.$$

And this is very good — we have a power larger than 1 of the influences, so we can pull out a bit outside (using the fact that it's small) and bound the rest by the total to get that this is at most

$$2^{-Ck/5\varepsilon} \sum_{j \notin J} I_j[f] \leq 2^{-Ck/5\varepsilon} I[f] \leq k 2^{-Ck/5\varepsilon} < \varepsilon$$

(the term that's exponential in k is clearly much smaller than k).

So now we've bounded $\|f - G\|_2^2 \leq 2\varepsilon$, and we are done. □

Remark 7.10. Note that $\|f - H\|_2^2 = 4\mathbb{P}[f(x) \neq H(x)]$; and when we say ε-close, we're referring to this probability.

§7.4 Isoperimetric inequalities

This is enough technical stuff for one day, so for the rest of the class we'll have storytime about isoperimetric inequalities.

The things we've seen in the last two lectures are all *isoperimetric inequalities*. We can imagine the Boolean cube as the graph with vertices $\{-1,1\}^n$ and edges (x, y) where x and y differ on exactly 1 coordinate.

And then we have a Boolean function $f\colon \{-1,1\}^n \to \{0,1\}$, which corresponds to a subset of vertices. Influences and total influence really capture the *edge-boundary* of this set f — because we're exactly measuring the edges that go between the subset that f represents and the other side.

And if we look at the Poincare inequality, which tells us $I[f] \geq \mathrm{Var}(f)$, this tells us that if we have a roughly equal partition of the graph, then the edge boundary has to be at least somewhat large (it can't be literally 0).

And Friedgut gives you a characterization of cuts in this graph that don't pass a lot of edges — it's telling you these cuts must be like juntas (i.e., they must be local, in the sense that they only depend on a few variables).

Question 7.11. Is this tight? And are there other measures of a boundary you can consider and prove lower bounds on?

Of course, in general this is tight. But for now, let's think of $\mathrm{Var}(f)$ as small. In particular, let's say $\mathbb{E}[f] = 2^{-k}$, where k is largish. Can this still be tight?

First, what's an example of *any* function where Poincare is tight? You can look at dictatorships $f(x) = x_i$ (more precisely $f(x) = \frac{1}{2}(x_1 + 1)$); then $\mathrm{Var}(f)$ is constant and the total influence is constant, so all is good. But of course, dictatorships are balanced.

Question 7.12. Can we engineer a function where $\mathbb{E}f$ is small, but where $I[f]$ and $\mathrm{Var}[f]$ are of the same order of magnitude?

A natural idea to try to get a function with small expectation is to take one of these, and then take intersections of these sorts of cuts. With a dictatorship we're just partitioning $x_1 = 1$ and $x_1 = -1$; so we can try taking $x_1 = \cdots = x_k = 1$. This coresponds to the function

$$f(x) = \prod_{i=1}^{k} \left(\frac{1 + x_i}{2}\right) = 1_{x_1 = \cdots = x_k = 1}.$$

But if we compute influences, for $1 \leq i \leq k$, i is only influential when all the other variables are 1 (among the first k), so that x_i is the one that's deciding; this means

$$\mathbb{E}f = 2^{-k} \text{ and } I[f] = k \cdot \frac{1}{2^{k-1}}.$$

So basically, there's a gap of k.

It turns out that this is actually the right answer.

Theorem 7.13 (Edge isoperimetric inequality)
For all $f\colon \{-1,1\}^n \to \{0,1\}$, we have

$$I[f] \gtrsim \mathrm{Var}[f] \log \frac{1}{\mathrm{Var}[f]}.$$

We're not going to prove this (we might eventually, at some point); but we started with Poincare and argued that for small sets we should be able to get an improvement, and we stated one.

But there's other measures of boundaries we can consider. One very natural one is the *vertex* boundary. Intuitively, we look at f, and we look at all points that are on its boundary — meaning they have a neighbor outside the set.

Definition 7.14. We define $\textsc{VertexBoundary}(f) = \{x \mid S_f(x) > 0\}$.

As an example, let's consider the majority function $f\colon \{-1,1\}^n \to \{0,1\}$ defined as $f(x) = 1_{\sum x_i \geq 0}$. Then what is $\mu(\textsc{VertexBoundary(f)})$? We can imagine the Boolean cube with a bunch of 0's and 1's; the vertex boundary will be the two middle layers, which means

$$\mu(\textsc{VertexBoundary}) = \Theta \frac{\binom{n}{n/2}}{2^n} = \Theta(1/\sqrt{n}).$$

But the edge boundary has to be fairly large. There's no contradiction, because for each vertex on the boundary, its sensitivity is actually *linear* in n; so when we look at the average sensitivity we get at least $\sqrt{n}$, and all is good.

But there are isoperimetric inequalities for vertex boundaries as well. For example, this is what the Kruskal–Katona theorem is about — suppose we have some budget of vertices. It says that if we want to minimize the vertex boundary, then we should shove all these vertices down according to Hamming weight.

Question 7.15. If we look at the examples pushing edge-boundaries and vertex-boundaries to the limit, they look very different. Can you prove a result that mixes the two?

The answer is yes; this was done in two steps.

Theorem 7.16 (Margulis)
We have $I[f] \cdot \mu(\textsc{VertexBoundary}) \gtrsim \mathrm{Var}(f)^2$.

This is really telling you that both theorems can't be tight at teh same time — because if we have a balanced function the minimum vertex boundary is $1/\sqrt{n}$, but in that case our edge-boundary has to be at least $\sqrt{n}$.

We will not see this proof, partly because the paper is in Russian and partly because we don't have to — a few years after that, Talagrand proved an even stronger result.

Recall we looked at average sensitivity, and proved that it's at least the total influence — so $\mathbb{E}_x S_f(x) \geq \mathrm{Var}(f)$ by Poincare. But Talagrand proved that in fact, we can squeeze a square root in here.

Theorem 7.17
We have $\mathbb{E}_x \sqrt{S_f(x)} \gtrsim \mathrm{Var}(f)$.

Remark 7.18. Recall that $S_f(x)$ is the number of coordinates i such that if we flip the value of x_i, then the value of f changes — equivalently, the number of neighbors of x in the graph.

We need to make several comments. First, this is stronger than the theorem by Margulis. To see this (i.e., that Talagrand implies Margulis), we can use Cauchy–Schwarz — we have

$$\left(\mathbb{E}_x\sqrt{S_f(x)}\right)^2 = \left(\mathbb{E}_x 1_{S_f(x)>0}\sqrt{S_f(x)}\right)^2 \leq \mathbb{E}_x 1_{S_f(x)>0}\mathbb{E}S_f(x)$$

by Cauchy–Schwarz, and this is exactly the left-hand side of Margulis.

So that's nice. Talagrand gave a proof of this result; if you've seen several papers of Talagrand you may guess that this is by induction on n, and it is. It's a very nice proof. On a problem set we may see a different, more modern proof.

Now we have two separate threads of discussion. The first is looking at the vertex boundary, and eventually squeezing it into this nice-looking theorem. But we also have the other strengthening of Poincare, which manages to give us something for *small* sets (the edge isoperimetric inequality). And that inequality and Talagrand's theorem are actually incomparable — we can't start with one and deduce the other.

So then Talagrand went on to try to get a unifying result that captures both of them. And he actually managed to do this.

Theorem 7.19 (Talagrand)

We have $\mathbb{E}\sqrt{S_f(x)} \gtrsim \operatorname{Var}[f]\sqrt{\log 1/\operatorname{Var}(f)}$.

(The previous result was a nice induction; this one is a not so nice induction.)

So now we have one theorem that rules all the theorems we currently have on the board; but it doesn't imply KKL. So can we strengthen this further to get something that also implies KKL? Here Talagrand couldn't *quite* make that work, but he did prove something towards this.

How do you squeeze influences into here? We'll consider the quantity $I_i[f]^2$. If all influences are small (e.g., $\log n/n$), then this is polynomially small.

Theorem 7.20 (Talagrand)

There exists $\alpha \in (0, 1/2)$ such that for all Boolean f, we have

$$\mathbb{E}\sqrt{S_f(x)} \geq \operatorname{Var}(f)\log\left(\frac{1}{\operatorname{Var}(f)}\right)^{1/\alpha-\alpha}\left(\log\frac{1}{\sum I_i[f]^2}\right)^{\alpha}.$$

So what Talagrand managed to do is to replace an α-power of the right-hand side with something telling us whether the influences are large or small. This implies a weak version of KKL, where instead of $\log n/n$ you get $(\log n)^c/n$. He also conjectured that $\alpha = 1/2$ could be taken; if that is true, then this theorem would really unify everything (including KKL), which would be really nice.

And this conjecture was actually proven a couple of years ago, by Elden–Gross in 2020. They used stochastic calculus; somehow they formulated everything in terms of martingales and used very nice technology.

But fortunately, there's also a different proof. Earlier Dor mentioned that the first result is a nice induction by Talagrand, but there's also a proof without induction. And that proof without induction sort of gets all this stuff; we're either going to see it in a lecture or problem set.

§8 March 5, 2024

Today our primary topic will be *noise stability*, a new notion we'll define; and we'll see some ways to use it. In particular, we'll see one very nice calculation about the noise stability of the majority function; we'll then see another application, Arrow's impossibility theorem.

§8.1 Noise stability

§8.1.1 Motivation

To motivate the definition of noise stability, let's return to thinking of Boolean functions as voting schemes — given a function $f\colon \{-1,1\}^n \to \{-1,1\}$, we think of the input x as the votes of n people (choosing between $+1$ and -1); then f aggregates these votes and comes to a conclusion.

But in elections, you might not actually get the actual inputs — you'll often just get something *close* to them, i.e., some noisy version y of x. This noise can be anything — it can be malicious (from someone switching votes) or honest (maybe we're passing through some channel, and things flip). So we'd like to use functions f where this noise does not matter so much.

Question 8.1. Can we design voting functions f where this noise doesn't matter so much?

The meaning of 'doesn't matter so much' is up to interpretation. If there's noise, we can't exactly replicate the no-noise case, but we can still try to do the best we can.

So now the two words 'noise stability' make sense — we want to design functions which are stable under noise. This means we want to be able to *measure* how stable a function f is to noise.

§8.1.2 Definitions

We know what a function is, and we know what noise is — we defined the noise operator a few weeks ago. So we'll define noise stability based on this.

Definition 8.2. Let $\rho \in [0,1]$. For $f\colon \{-1,1\}^n \to \mathbb{R}$, we define the *noise stability* of f under correlation ρ as

$$\mathrm{Stab}_\rho(f) = \langle f, T_\rho f\rangle.$$

Remark 8.3. We think of ρ as our *corerlation parameter*; we'll later allow ρ to be negative as well.

First let's do a sanity check to make sure this is actually capturing what we want it to capture. Suppose that $f\colon \{-1,1\}^n \to \{-1,1\}$. We'll try to express the stability in terms of the probability that the value changes — we can write

$$\mathrm{Stab}_\rho(f) = \langle f, T_\rho f\rangle = \mathbb{E}_{x, y\sim T_\rho x} f(x)f(y).$$

And if $f(x) = f(y)$ we get a 1, while if they're not then we get a -1; so then this is

$$\mathrm{Stab}_\rho(f) = \mathbb{P}[f(x) = f(y)] - \mathbb{P}[f(x) \neq f(y)] = 2\mathbb{P}[f(x) = f(y)] - 1.$$

So we've defined a good notion — this really captures what we want.

In the rest of the lecture, we'll play around with this notion and prove some stuff using it.

Remark 8.4. For $y \sim T_\rho x$, this means for each coordinate, we keep it with probability ρ and resample it at random otherwise; in other words, $\mathbb{E}x_iy_i = \rho$.

What we're doing here is modelling noise which is not adversarial — it's completely independent of what x was (because each coordinate is independent).

§8.2 Designing stable functions

Question 8.5. What functions $f\colon \{-1,1\}^n \to \{-1,1\}$ maximize $\operatorname{Stab}_\rho(f)$?

The answer, of course, is constant functions. This is not a very answer, so let's rule them out — let's require that f is balanced, i.e., $\mathbb{E}f = 0$.

Then the answer is a dictatorship — i.e., a function $f(x) = x_1$. We'll soon prove this fact; it's not very hard, but the moral is quite depressing if you look at the motivation — it says that dictatorships are the most stable. So can we rule this out? We want to rule out dictatorships, but how do we rule out dictatorships exactly? We can do so in terms of influences — we'll look at functions where all the influences are small.

Question 8.6. What about functions (still balanced) with $\max_i I_i[f] \leq \tau$?

This problem is much more difficult.

Theorem 8.7 (MOO, Majority is Stablest theorem)
For every $\varepsilon > 0$ and $\rho \in [0,1]$, there exists $\tau > 0$ such that if $f\colon \{-1,1\}^n \to \{-1,1\}$ is balanced and $I_i[f] \leq \tau$ for all i, then $\operatorname{Stab}_\rho(f) \leq \operatorname{Stab}_\rho(\text{majority}) + \varepsilon$.

This is a remarkable theorem for several reasons. First, the statement itself is very nice — once you eliminate constant functions and dictatorships (both of which don't satisfy us for several reasons), then indeed majority is the best you can do. Second, the proof is very interesting — it uses what's called the *invariance principle* (which we won't discuss today, but we will in a few lectures). And third (and what Dor likes the most), the authors don't care about voting schemes or any of this; the reason they proved the theorem (and the reason it was conjectured a year earlier) is because of hardness of approximation.

Question 8.8 (Max-Cut). Given a graph $G = (V, E)$, partition the vertices into two parts such that as many edges cross the cut as possible.

This is one of the classical NP-complete problems; it's very hard to solve (i.e., to find the maximum), so then you can resolve for an approximation.

For simplicity, let's say that G contains a very large cut — specifically, a cut of size $(1-\varepsilon)\left|E\right|$. What's the best cut we can *find*?

You can get a cut with just half the edges (with random sampling), and for a while this was the best known; then GW (1995) showed that you can find a cut of size $(1 - \Theta(\sqrt{\varepsilon}))\left|E\right|$. And it turns out that this is optimal, at least under some complexity assumptions; this was proved by KKMO, and the main technical ingredient of their proof is precisely this result. So the reason they proved this theorem is approximation of Max-Cut. (Goemans–Williamson tells you that you can design an efficient algorithm getting a cut of this size, and KKMO tells you this is the best you can do in polynomial time — assuming the unique games conjecture (some complexity assumption that we'll elaborate on later), doing better is NP-hard.)

We'll discuss both the invariance principle and this connection later in the course, but for now we're just mentioning it as motivation for why noise stability is interesting.

We'll start out by proving that dictatorships are the best you can do among all balanced functions.

Claim 8.9 — If $\mathbb{E}f = 0$, then $\operatorname{Stab}_\rho(f) \leq \operatorname{Stab}_\rho(\text{dictatorship})$.

The only thing we have to do in order to prove this is to get a Fourier analytic formula for the stability.

Lemma 8.10

For a function $f\colon \{-1,1\}^n \to \{-1,1\}$, we have $\mathrm{Stab}_\rho(f) = \sum_S \rho^{|S|}\widehat{f}(S)^2$.

Proof. We have $\mathrm{Stab}_\rho(f) = \langle f, T_\rho f\rangle = \sum_S \widehat{f}(S)\widehat{T_\rho f}(S)$ by Plancherel, and earlier we computed that $\widehat{T_\rho f}(S) = \rho^{|S|}\widehat{f}(S)$, so we're done. □

Proof of claim. If f is balanced, the empty Fourier coefficient is just $\mathbb{E}f = 0$, and so we get

$$\mathrm{Stab}_\rho(f) = \sum_{|S|\geq 1} \rho^{|S|}\widehat{f}(S)^2.$$

And $\rho^{|S|}$ is a decreasing function (the larger S is, the smaller it is), so this is at most

$$\rho \sum_{|S|\geq 1} \widehat{f}(S)^2 \leq \rho.$$

> **Remark 8.11.** The reason we need f to be balanced is because we want to throw away the empty coefficient (which isn't punished by any factor of ρ — this is the reason constants are good).

And the stability of a dictatorship is exactly ρ (by the same computation). □

§8.3 Estimating noise stability of majority

We've mentioned the Majority is Stablest theorem and mentioned that it's relevant to hardness of approximation algorithms; now we'll do some computations to get asymptotics for its stability. To appreciate this, note that if we just apply the Fourier-analytic formula, we won't get very far (the Fourier expansion of the majority function is pretty messy). But there's a nice geometric idea here (and you may even see MaxCut entering the picture).

We'll define majority in a funny way — we can define it as the function $f\colon \{-1,1\}^n \to \{-1,1\}$ as

$$f(x) = \mathrm{sgn}\left(\frac{\sum_{i=1}^n x_i}{\sqrt{n}}\right).$$

(Of course, the $\sqrt{n}$ doesn't affect the sign of the sum, but it'll be useful for what follows.)

Now imagine we sample $x \in \{-1,1\}^n$ and $y \sim T_\rho x$, and see what happens when we plug in both x and y to f. From x we get

$$X = \frac{1}{\sqrt{n}}\sum_{i=1}^n x_i,$$

and similarly from y we get

$$Y = \frac{1}{\sqrt{n}}\sum_{i=1}^n y_i.$$

If we look at X marginally, it looks like a standard normal — so $X \sim \mathcal{N}(0,1)$ (approximately — it's not *exactly* normal, but we'll just roll with this). And the same is true of Y — if we look at y marginally, it's still uniform, so $Y \sim \mathcal{N}(0,1)$. But we have two normals that are correlated, so we can compute the correlation between them — we have

$$\mathbb{E}[X\cdot Y] = \mathbb{E}_{x,y}\left[\frac{1}{n}\sum_{i=1}^n x_iy_i + \frac{1}{n}\sum_{i\neq j} x_iy_j\right].$$

The contribution of the off-diagonal terms is 0 — if $i \neq j$ then x_i and y_j are independent. Meanwhile, the diagonal terms contribute exactly ρ, since $\mathbb{E}[x_i y_i] = \rho$ (and there are n terms). So we get

$$\mathbb{E}[X \cdot Y] = \rho.$$

Now applying the central limit theorem gives that X and Y is each Gaussian; but applying the *multidimensional* central limit theorem (which we will not state) gives that X and Y are (approximately) joint Gaussians with correlation ρ.

We wanted to compute $\mathrm{Stab}_\rho(f)$, which we saw is $2\mathbb{P}[f(x) = f(y)] - 1$; so we'll just consider $\mathbb{P}[f(x) = f(y)]$ (for $x \in \{-1, 1\}^n$ and $y \sim T_\rho x$). By the multidimensional central limit theorem, we can show that

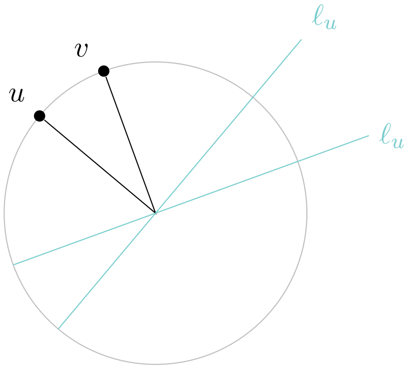
$$\mathbb{P}_{x,y}[f(x) \neq f(y)] = \mathbb{P}_{(G_1,G_2)}[\mathrm{sgn}(G_1) \neq \mathrm{sgn}(G_2)] + o(1),$$

where G_1 and G_2 are ρ-correlated joint Gaussians. Now it seems like we had a nice discrete problem and turned it into a continuous problem where we'll have to compute integrals, but it turns out we actually don't — there's a very nice trick to compute this probability.

It turns out there's a very nice way to generate the distribution of ρ-correlated Gaussians, which will be useful here — pick two vectors $u, v \in \mathbb{R}^2$ with $\|u\|_2 = \|v\|_2 = 1$ and $\langle u, v\rangle = \rho$, and take $z \sim \mathcal{N}(0, 1)^2$. We then take $G_1 = \langle u, z\rangle$ and $G_2 = \langle v, z\rangle$. It's a standard fact that then each of G_1 and G_2 is a Gaussian (we have $\mathbb{E}G_1 = 0$ and $\mathrm{Var}[G_1] = \|u\|_2^2 = 1$), and by the same computation as before, we end up with $\mathrm{Cov}[G_1, G_2] = \langle u, v\rangle = \rho$. So the probability we want to compute is the same as

$$\mathbb{P}_z[\mathrm{sgn}(\langle u, z\rangle) \neq \mathrm{sgn}(\langle v, z\rangle)] + o(1).$$

And now we've made huge progress, and you can see the answer geometrically:



The vector u partitions the space into two halves — the half with positive inner product with u, and the half with negative inner product. And the same is true of v. (Here ℓ_u and ℓ_v are the normals to u and v.)

Then the inner products have different signs if and only if z is between these two normals.

And z is a Gaussian, which has a direction and a length; we don't care about the length, only the direction, which is completely uniform. So we get that this probability is

$$\frac{\measuredangle(\ell_u, \ell_v)}{\pi} = \frac{\measuredangle(u, v)}{\pi}$$

(there are two angles, which is why we divide by π instead of 2π). And we have $\cos\measuredangle(u, v) \cdot \|u\|_2 \|v\|_2 = \langle u, v\rangle = \rho$, so we end up with that our probability is

$$\frac{\arccos(\rho)}{\pi} + o(1).$$

Corollary 8.12 (Sheppard's formula)
We have $\text{Stab}_\rho(\text{majority}) = 1 - \frac{2}{\pi}\arccos(\rho) + o(1)$.

Remark 8.13. We could have done this thing with z even in the original setting — instead of thinking about the majority, we can think about $\text{sgn}(\sum a_i x_i)$ where $a_i \in \{\pm 1\}$. This is the analog of moving to z. But then you're still discrete instead of continuous; we moved to Gaussian space to avoid that. And then this becomes a very nice geometric problem.

This actually gives you the precise number for the max-cut problem.

And as mentioned before, trying to find the Fourier coefficients of the majority function is difficult; but once you have this formula, you can actually reverse-engineer the weights of the majority function. We get a formula $\sum_{k=0}^n \rho^k W^{=k}[\text{maj}]$ that holds for *every* ρ; and so now we have two polynomials in ρ that are basically equal (we know $\text{Stab}_\rho(\text{majority})$ is supposed to be a polynomial in ρ depending on the Fourier coefficients, and you can use the Taylor series of arccos), and you can compare them.

§8.4 Arrow's impossibility theorem

The last thing we'll do today is prove Arrow's impossibility theorem, which we stated in the first lecture. (Now we have more notation, so we can more easily state the theorem.)

Imagine that there are three candidates A, B, and C in an election, and we have n voters. The voters submit their preferences between each two candidates — each voter i casts their vote via three bits $x_i, y_i, z_i \in \{-1, 1\}$. These are interpreted in the following way: if voter i prefers A over B, then $x_i = 1$; otherwise $x_i = -1$. Similarly, if i prefers B over C, then $y_i = 1$; otherwise $y_i = -1$. And finally, if i prefers C over A, then $z_i = 1$; otherwise $z_i = -1$. (This is the way we'll formalize voting between three candidates using Boolean bits.)

We talked about the notion of a valid ranking — we don't want someone to say they prefer A over B, B over C, and C over A (that makes no sense). This gives us a condition on what the possible (x_i, y_i, z_i) tuples are — they shouldn't be all 1's or all -1's.

Definition 8.14. The *not-all-equal* function $\text{NAE}\colon \{-1,1\}^3 \to \{0,1\}$ is defined as

$$\text{NAE}(a, b, c) = 1 \iff a, b, c \text{ are not all the same.}$$

We then say that the ranking of voter i is *valid* if it is in the support of NAE — i.e., $(x_i, y_i, z_i) \in \text{NAE}^{-1}(1)$.

Now each voter has a vote, so we get (x_1, y_1, z_1), (x_2, y_2, z_2), and so on, which we can arrange in a table.

$$\begin{array}{ccc} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{array}$$

And now we want to aggregate these votes. The way we'll do this is by applying a function on each of the columns.

In other words, to figure out whether the overall election should prefer A over B, we look at each of the individual preferences between A and B and apply f.

Theorem 8.15

Suppose that f is a function such that:

(1) Whenever (x_i, y_i, z_i) are all valid, then $(f(x), f(y), f(z))$ is valid as well.

(2) $f(1) = 1$ and $f(-1) = -1$ (i.e., if everyone prefers A over B, then the final ranking will too, and vice versa).

Then f is a dictatorship.

We're going to prove this now. The proof is kind of odd, in the sense that the main idea is to cast the problem in the right way on the Boolean hypercube, and once you do this it more or less follows by itself.

We'll need some facts. First, we want to apply Fourier analysis, which means we want to express NAE as a polynomial in a, b, and c.

Fact 8.16 (1) — We have $\mathrm{NAE}(a, b, c) = \frac{3}{4} - \frac{1}{4}(ab + bc + ac)$.

Proof. Once you guess the formula, you can check it by hand. But guessing is hard; alternatively, we have

$$\mathrm{NAE}(a, b, c) = 1 - 1_{a=b=c} = 1 + \frac{(a-1)(b-1)(c-1)}{8} - \frac{(a+1)(b+1)(c+1)}{8},$$

and then we expand and we're done. □

Now we need a definition. At the beginning of the lecture, we talked about stability when $\rho \in [0, 1]$, but sometimes it makes sense to take negative ρ; this is what we'll do now.

Definition 8.17. The distribution $(a, b) \in \{-1, 1\}^2$ where a is uniform, b is marginally uniform, and $\mathbb{E}[ab] = \rho$ is called *ρ-correlated*, for any $\rho \in [-1, 1]$.

When ρ is nonnegative, we already saw how to generate this distribution. For $\rho < 0$, here's a way to do so — we can sample a uniformly. Then with probability $|\rho|$ we take $b = -a$; otherwise we sample b uniformly. (We can check that $\mathbb{E}[ab]$ is indeed ρ.)

We'll still use the notation $b \sim T_\rho a$, even when $\rho < 0$.

Proof of Arrow's theorem. We know that when (x_i, y_i, z_i) is valid, so is f. So we'll sample (x_i, y_i, z_i) from the set of all possible rankings, independently for all i — this means we sample $(x_i, y_i, z_i) \sim \mathrm{NAE}^{-1}(1)$. We know that the result of f is also valid, so

$$\mathrm{NAE}(f(x), f(y), f(z)) = 1.$$

On the other hand, we have an expression for what NAE does, so we can write this as

$$\mathrm{NAE}(f(x), f(y), f(z)) = \frac{3}{4} - \frac{1}{4}(f(x)f(y) + f(x)f(z) + f(y)f(z)).$$

Now we're going to combine these two observations, and take the expectation over x, y, and z — we have

$$1 = \mathbb{E}_{x,y,z}\left[\frac{3}{4} - \frac{1}{4}(f(x)f(y) + f(y)f(z) + f(x)f(z))\right] = \frac{3}{4} - \frac{3}{4}\mathbb{E}_{x,y}[f(x)f(y)]$$

(since the marginal distribution of any of the three pairs (x, y), (y, z), and (x, z) is the same). And now comes the miracle, which is that we're going to look at the distribution of (x, y) and try to figure out what it is. We can write down the support of NAE as $\{(1,1,-1),(1,-1,1),(-1,1,1),(1,-1,-1),(-1,-1,1),(-1,1,-1)\}$. Each bit by itself is uniform (there's 3 places where it's $+1$, and 3 where it's -1), while

$$\mathbb{E}[x_1y_1] = \frac{1}{6}(2-4) = -\frac{1}{3}.$$

And so we've learned that x and y are ρ-correlated for $\rho = -1/3$ — so

$$1 = \frac{3}{4} - \frac{3}{4}\operatorname{Stab}_{-1/3}(f).$$

Now we can just rearrange to get that

$$\operatorname{Stab}_{-1/3}(f) = -\frac{1}{3}.$$

(It's some sort of miracle that we got a $-1/3$ in both places, and we'll soon see why.)

Now if we just look at the Fourier analytic formula for the stability, we'll realize something very special about f — we get that

$$\operatorname{Stab}_{-1/3}(f) = \sum_S \rho^{|S|}\widehat{f}(S)^2,$$

where $\rho = -1/3$. When $|S| = 0$, $\rho^{|S|}$ is positive, so it doesn't help us get to $-1/3$. When $|S| = 1$ it's $-1/3$. In general even $|S|$ gives a positive coefficient and odd $|S|$ gives a negative one, but the coefficient gets smaller as $|S|$ gets larger. And so the only way to get to $-1/3$ is if $W^{=1}[f] = 1$. (In other words, we have $\operatorname{Stab}_{-1/3}(f) \geq -\frac{1}{3}W^{=1}[f] - \frac{1}{27}(1 - W^{=1}[f])$.)

And now we're done — because we have a Boolean function which we know is of degree 1, and by a problem from the problem set, f must be a dictatorship, i.e., $f = b \cdot x_i$. (The second condition is just to say that $b = 1$.) □

This is kind of a magical proof — somehow you get exactly the right $-1/3$. But if you stare at this proof, we've actually proved more — we've proved a *robust* version, that you can weaken condition 1 to say that $(f(x), f(y), f(z))$ is valid with probability $1 - \varepsilon$. (What you'll find is that almost all of the weight of f is on level 1, and then you can use the FKN theorem to conclude that f is very close to a dictatorship.)

§9 March 7, 2024

§9.1 Noise sensitivity

Definition 9.1. Fo a function $f\colon \{-1,1\}^n \to \mathbb{R}$ and $\varepsilon > 0$, the *noise sensitivity* of f is $\operatorname{NS}_\varepsilon(f) = \frac{1}{2}(1 - \operatorname{Stab}_{1-2\varepsilon}(f))$.

This is a weird definition, so to make sense of it, let's first consider the case where f is boolean — i.e,. $f\colon \{-1,1\}^n \to \{-1,1\}$. We know $\operatorname{Stab}_\rho(f) = 1 - 2\mathbb{P}[f(x) \neq f(y)]$, which is the reason for the $\frac{1}{2}$ and $1 - \bullet$ — then we get that

$$\operatorname{NS}_\varepsilon(f) = \mathbb{P}_{x\in\{-1,1\}^n, y\sim T_{1-2\varepsilon}x}[f(x) \neq f(y)].$$

(We typically think of ε as small.)

This should explain why this is called noise sensitivity — we're measuring how sensitive f is to noise.

If we take x and y to be fully independent, then what is $\mathbb{P}[f(x) \neq f(y)]$? By symmetry this is

$$2\mathbb{P}[f(x) = 1]\mathbb{P}[f(y) = -1] = \frac{1}{2}\operatorname{Var}[f]$$

(if you play around with the definition of variance a bit). So we'll call a function noise-sensitive if when we choose x and y to be fairly correlated, the values of $f(x)$ and $f(y)$ act as if x and y were fully independent.

Definition 9.2. We say f is (ε, δ)-noise sensitive if

$$\left|\mathrm{NS}_\varepsilon(f) - \frac{1}{2}\operatorname{Var}[f]\right| \le \delta.$$

The noise sensitivity of functions or processes is something that comes up often in probability and percolation theory, where people ask a bunch of such questions. The main result we'll see today is by Benjamini–Kalai–Schramm; this is the motivation they had in mind.

§9.2 Fourier analytic formulas

We want to prove results of the form 'if f satisfies —, then it is noise sensitive.' But from our perspective, we know about total influence and degrees, so let's first see what noise sensitivity tells us about stuff we already know.

First, there's a nice Fourier analytic formula for the noise sensitivity of f. We have a formula for stability, so we can just plug it into the definition.

Claim 9.3 — We have $\mathrm{NS}_\varepsilon = \frac{1}{2}\sum_S((1-(1-2\varepsilon)^{|S|}))\widehat{f}(S)^2$.

Proof. We proved last class that $\mathrm{Stab}_{1-2\varepsilon}(f) = \sum_S (1-2\varepsilon)^{|S|}\widehat{f}(S)^2$. Noting that $\sum \widehat{f}(S)^2 = 1$ gives the desired result. □

Now we can rephrase the condition for (ε, δ) noise sensitivity in terms of Fourier coefficients: Recall that $\operatorname{Var}[f] = \sum_{S \neq \emptyset} \widehat{f}(S)^2$ — and this sum doesn't include the empty Fourier coefficient (because $1-(1-2\varepsilon)^0 = 0$). So the condition becomes

$$\frac{1}{2}\sum_{S\neq\emptyset}(1-2\varepsilon)^{|S|}\widehat{f}(S)^2 \le \delta,$$

which we can rewrite as

$$\frac{1}{2}\sum_{k=1}^{n}(1-2\varepsilon)^k W^{=k}[f] \le \delta.$$

The factor of $(1-2\varepsilon)^k$ is exponentially decaying in k, so if we take the sum from large k onwards (e.g., $k \gg \varepsilon^{-1}\log(1/\delta)$), it's going to basically be 0 (since $(1-2\varepsilon)^k$ is tiny, and $\sum_k W^{=k}[f] = 1$). But for small k, the $(1-2\varepsilon)^k$ factor won't matter too much. So morally, a function is noise sensitive if and only if it has $o(1)$ weight on low Fourier levels.

When we do math we have to be precise with definitions, but this is the way we should think of noise sensitivy — as having very little weight on low levels.

§9.3

It turns out that there's one parameter that's really important for studying this — we define

$$M(f) = \sum_{i=1}^{n} I_i[f]^2.$$

This is an odd-looking parameter; what does it mean? Here's one reason people look at it (and the primary reason that BKS looked at it):

Claim 9.4 — If f is monotone and Boolean, then $M(f) = W^{=1}[f]$.

Proof. On the second problem set, we'll prove that for a monotone function, $I_i[f] = f(\{i\})$. □

Now with the earlier intuition in mind, if we want a monotone function to be noise-sensitive, then at the very least the weight at level 1 should be small, so at the very least this parameter should be small. Remarkably, it turns out that this is actually enough!

This is the reason this parameter makes sense, but now that it makes sense, we can talk about it for *any* function, not just monotone ones.

Theorem 9.5

If a function $f\colon \{-1,1\}^n \to \{-1,1\}$ has $M(f) = o(1)$, then f is noise sensitive.

(For now we'll state the theorem informally; we'll write out the ε's and δ's later on.)

This will be our main theorem today.

Remark 9.6. If f is monotone, this is an if and only if. For monotone functions, it's saying that if you have weight on any low level, then you necessarily have weight on level 1.

§9.4 The level d inequality

The journey to prove this theorem will be pretty challenging; we'll need some tools, in particular the level d inequality — an important result that has many applications outside this as well. We'll see two proofs — the book proof, and then a more complicated proof that's morally the same (for reasons we'll see later).

Lemma 9.7

Let $g\colon \{-1,1\}^n \to \{-1,0,1\}$, and suppose that $\delta = \mathbb{P}_x[g(x) \neq 0]$. Then for all $d \in \mathbb{N}$, we have

$$W^{\leq d}[g] \lesssim \delta^2 \log^d(4/\delta).$$

(We think of δ as small.) We saw this last class with $d \leq \frac{1}{20}\log 1/\delta$, and some funny number. To figure out where the right-hand side comes from, we have $W^{=0}[g] = (\mathbb{E}g)^2$, which could be up to δ^2. This inequality says that if we look up to level d, then we almost get the same thing; of course you can't exactly get δ^2 (that would be too good), so you have to lose some log factors.

Remark 9.8. We use $W^{\leq d}[f]$ to denote $\sum_{|S|\leq d} \widehat{f}(S)^2$.

Proof. Let $g^{\leq d}(x) = \sum_{|S|\leq d} \widehat{g}(S)\chi_S(x)$ be the part of the Fourier transform of g with degree at most d. Then we have

$$W^{\leq d}[g] = \langle g^{\leq d}, g^{\leq d}\rangle.$$

As before, we'll perform a clever trick (replacing one of these with g, using the fact that $g^{\leq d}$ and $g^{>d}$ are orthogonal) and then apply Hölder — this is

$$W^{\leq d}[g] = \langle g^{\leq d}, g^{\leq d}\rangle = \langle g, g^{\leq d}\rangle \leq \|g\|_{\frac{q}{q-1}} \left\|g^{\leq d}\right\|_q \leq \|g\|_{\frac{q}{q-1}} \sqrt{q-1}^{\,d} \left\|g^{\leq d}\right\|_2.$$

And the second term is $\sqrt{W^{\leq d}[g]}^2$ by definition. Dividing by this and squaring, we get

$$W^{\leq d} \leq (q-1)^d \left\lVert g \right\rVert_{\frac{q}{q-1}}^2 .$$

And now because g only takes values in $\{0, -1, 1\}$, we can calculate what this norm is — and we get that

$$W^{\leq d}[g] \leq (q-1)^d \delta^{2 \cdot \frac{q-1}{q}} .$$

We now see that if we choose q large, then our power on the right approaches 2 (which is what we want), but the first term starts becoming big. So we have to optimize — we pick $q = \log(2/\delta)$ and get

$$W^{\leq d}[g] \leq \log^d(2/\delta) \cdot \delta^2 \cdot \delta^{2/\log(2/\delta)}.$$

Using log rules, the last term is $O(1)$ (this is a standard fact), which proves the theorem. □

That's the level d inequality. It's a very nice result. What's the context for our discussion? We want to prove that a function f is noise sensitive, so we have to control the level k weight of it by some means. And this is a tool that lets you control the level d weight of something, given that you know the measure of it is small.

§9.5 A cheap version of BKS

First we'll make a first attempt at BKS; using this result we can already establish something nontrivial, that looks very similar to BKS but is weaker.

Lemma 9.9

Let $f \colon \{-1, 1\}^n \to \{-1, 1\}$ and $d \geq 1$. Then

$$W^{=d}[f] \lesssim \sum_{i=1}^{n} I_i[f]^2 \log^d \frac{4}{I_i[f]^2}.$$

The first term is essentially what we want; what we don't want is all the log factors.

Proof. We'll use the lemma on the derivatives of f — for every i, applying the level $d-1$ inequality to $\partial_i f$, we get that

$$W^{=d-1}[\partial_i f] \lesssim I_i[f]^2 \log^d \frac{4}{I_i[f]^2}.$$

(The lemma refers to $W^{\leq d-1}$, but this is certainly an upper bound.) Now summing this up gives that

$$\sum_{i=1}^{n} W^{=d-1}[\partial_i f] \lesssim \sum_{i=1}^{n} I_i[f]^2 \log^d \frac{4}{I_i[f]^2}.$$

The right-hand side is exactly what we want. Meanwhile, for the left-hand side we know the Fourier coefficients of $\partial_i f$ are the same as those of f, but we just take the ones containing i; this gives that the left-hand side is

$$\sum_{i=1}^{n} \sum_{|S| \geq 1, |S|=d} \widehat{f}(S)^2 = \sum_{|S|=d} |S| \, \widehat{f}(S)^2 \geq W^{=d}[f].$$

(The reason we have $d-1$ vs. d is because when we take a derivative, we delete i from our sets.) □

This is a nontrivial statement but doesn't get us as far as BKS. This is because basically what we're going to prove is that this result holds, except that we can replace the *individual* influences squared by the sum of all of them (which is better, and is going to give us the theorem).

So this is off. And why? If we think about all the level-d inequalities, if the proof goes south then all these inequalities are tight. But they can't all be tight, which is what we're going to exploit in the end. We'll write down some argument that resembles this in a sense, but where we treat all the variables at the same time.

§9.6 Proof of level d inequality

There's two reasons we're seeing a proof of BKS. It's a nice theorem, but we're also going to see two techniques — one on how to use tail bounds in proofs, and one on decoupling. We're going to present these two ideas separately; so first we'll see how one uses tail bounds in proofs. And to demonstrate that, we'll give a different proof of the level d inequality (which is going to look more complicated than the first, but gives you a sense of the types of arguments one can make).

We want to bound the level at most d weight of g. First, we have

$$W^{\leq d}[g] = \langle g^{leqd}, g\rangle = \mathbb{E}_x g^{\leq d}(x) g(x) \leq \mathbb{E}\left|g^{\leq d}(x)\right| \cdot |g(x)|.$$

What's going to be the point of this argument? Let's look at the term $|g^{\leq d}(x)|$. The 2-norm of $g^{\leq d}(x)$ is $\sqrt{W^{\leq d}[g]}$. And we know low-degree functions aren't crazy — they're not typically too much more than their 2-norm. For a moment, let's pretend that this is actually happening — because g is low-degree, we expect that typically $\left|g^{\leq d}(x)\right| \leq T\sqrt{W^{\leq d}[g]}$ (this is informal; we're going to make it formal soon). Let's work under the assumption that this is *always* true for now. Assuming that this inequality is always true, we would get

$$W^{\leq d}[g] \leq T\sqrt{W^{\leq d}[g]}\mathbb{E}\,|g(x)|.$$

And $\mathbb{E}\,|g(x)| = \delta$; dividing and squaring gives that

$$W^{\leq d}[g] \leq T^2\delta^2,$$

which is the type of thing we're shooting for.

If we ever run into low-degree functions, before we do things formally with ε's and δ's, it's useful to think of them as being bounded, try to do a sanity check like this, and if it works then we try to formalize it later.

So then, how do we formalize this?

Let E be the event that what we wanted is true — i.e., that

$$|g^{\leq d}(x)| \leq T\sqrt{W^{\leq d}[g]}.$$

Now we can look at our expectation and just split it according to whether E holds or not — we can write

$$(*) = \mathbb{E}_x\left|g^{\leq d}(x)\right| 1_E \cdot |g(x)| + \mathbb{E}_x\left|g^{\leq d}(x)\right| 1_{\overline{E}}\,|g(x)|.$$

Let's call these terms (1) and (2). We can bound (1) easily, in the way we just did — because E holds we can bound $|g^{\leq d}(x)|$ in the way we did, and following your nose, you get

$$(1) \leq T\sqrt{W^{\leq d}[g]} \cdot \delta.$$

And what's the point of (2)? We'll look at $\overline{E}$; this is a very rare event. So if we could somehow isolate $\mathbb{P}[|E|]$ and just be left with an average, we'd be happy. There are many ways to do this; we'll do the crudest (it's not necessarily the tightest, but it's good enough). Since g is $\{0, \pm 1\}$-valued we always have $|g(x)| \leq 1$, so we'll throw that away; then

$$(2) \leq \mathbb{E}_x \left| g^{\leq d}(x) \right| \cdot 1_E \leq \sqrt{\mathbb{E}_x \left| g^{\leq d}(x) \right|^2} \sqrt{\mathbb{P}[\overline{E}]}$$

by Cauchy–Schwarz. Now $\mathbb{E}_x \left| g^{\leq d}(x) \right|^2 = W^{\leq d}[g]$, and for the second term we can use our Chernoff bound — a few classes ago, we saw that the probability a low-degree function exceeds its 2-norm by a factor of T is exponentially decaying in some power of T, precisely giving

$$(2) \leq \sqrt{W^{\leq d}[g]} \cdot e^{-\frac{1}{4}T^{2/d}}.$$

When we put these together, we get that this is at most

$$\sqrt{W^{\leq d}[g]}(T\delta + e^{-\frac{1}{4}T^{2/d}}).$$

We now want to choose T that roughly balances these two things — the first term is roughly δ, so we want the second term to also be roughly δ, and so we set $T = (100 \log(2/\varepsilon))^{2/\delta}$ (this ensures that the first term is actually the main guy).

Now combining everything, we get that

$$W^{\leq d}[g] \lesssim \sqrt{W^{\leq d}[g]} T\delta,$$

and after dividing and squaring, we're done.

This proof is unarguably more complicated than the previous one, but it kind of gives you a feeling for the intuition that low-degree functions are 'bounded' and how you turn this intuition into a proof. And this is what'll happen in the proof of the actual BKS theorem.

Remark 9.10. We're not concerned about recovering exactly the level d inequality — this may get it up to a factor of 100^d, though you can probably recover the actual statement if you do things more carefully.

§9.7 Decoupling

Now that we've isolated one idea in the proof, we'll work towards the second idea, which is called decoupling.

Let's imagine we're interested in computing the level-d weight of some function (which is what we're doing in all of today). Then decoupling tells you that you can break the variables into two sets such that if you look at the level d weight and compare it to the level d weight where you take one variable from I and the rest from J, you recover a constant fraction of the original weight.

Claim 9.11 — There exists a partition of $[n]$ into two sets I and J such that

$$\sum_{\substack{|S|=d \\ |S \cap I|=1}} \widehat{f}(S)^2 \geq \frac{1}{e} \sum_{|S|=d} \widehat{f}(S)^2.$$

To avoid annoying cases, you can think of $d > 1$ (it doesn't matter). How are we going to choose I? It makes sense to choose it randomly, but let's try to think about what size we want. If we include every element with probability p, then what's the expected size of $|S \cap I|$? It's dp, and we want this to be 1; so let's take $p = \frac{1}{d}$.

Proof. Assume $d > 1$, set $p = 1/d$, and pick a partition $I \cup J$ by including each $i \in [n]$ independently in I with probability p, and in J otherwise. Now what we're going to do is calculate the expectation of the left-hand side and hope for the best. We have

$$\mathbb{E}_{I,J} \sum_{\substack{|S|=d \\ |S\cap I|=1}} \widehat{f}(S)^2 = \mathbb{E}_{I,J} \sum_{|S|=d} \widehat{f}(S)^2 \cdot 1_{|S\cap I|=1}.$$

And now the things we're summing over don't depend on I and J, so we can interchange the sum and expectation to rewrite this as

$$\sum_{|S|=d} \widehat{f}(S)^2 \mathbb{P}_{I,J}[|S \cap I| = 1].$$

And what is this probability? We have d elements, so the probability the intersection is some particular element is $p(1-p)^{d-1}$; since there are d elements in S, this is

$$d \cdot p \cdot (1-p)^{d-1} = \left(1 - \frac{1}{d}\right)^{d-1} \geq \frac{1}{e},$$

which gives that

$$\mathbb{E}(*) \geq \frac{1}{e} \sum_{|S|=d} \widehat{f}(S)^2.$$

So we've computed the expectation and got that it's what we wanted; and if the expectation of a random variable is some quantity, then with positive probability it's at least that quantity. So there exist I and J with the desired property. □

We don't have enough time to finish the proof of BKS today, so instead we'll just say one additional fact and then finish.

The BKS is a *sufficient* condition to prove noise sensitivity, and for monotone functions it's also necessary. BUt for general functions, maybe influences are hard to compute; and sometimes people want to use other tools to prove noise sensitivity, Today we'll state two results which are sometimes useful. There will be a final project in this course where you read research papers and summarize them; if these interest you, they are some of the options you can pick.

The *revealment* method is a very elegant way of trying or prove a function is noise sensitive, by actually presenting an algorithm for it. Suppose $f\colon \{-1,1\}^n \to \{-1,1\}$ is a function, and $\mathcal{A}$ is a randomized query algorithm computing f. What this means is there's some input x, and the algorithm doesn't have access to x. Instead, it can just say 'please give me coordinate number i,' and then it gets x_i. So $\mathcal{A}$ asks for the values of coordinates and gets them. And in the end, it's supposed to figure out the value of f.

We can imagine such an algorithm is *always* correct, but we typically ask such an algorithm to figure out $f(x)$ with high probability. Now define $\delta_i(\mathcal{A})$ as the probability that $\mathcal{A}$ reads the coordinate i — i.e.,

$$\delta_i(\mathcal{A}) = \mathbb{P}_x[\mathcal{A} \text{ reads coordinate } i].$$

(The probability is over both x and $\mathcal{A}$.)

Theorem 9.12 (S–S)
If $\max_i \delta_i(f) = o(1)$, then f is noise-sensitive.

The intuition is that if the algorithm only looks at a few places in the input but there's no one particular input it cares about, then the Fourier mass has to be spread and on high levels. The proof is elegant, and there are several nice applications.

> **Remark 9.13.** Note that constant functions are very noise-sensitive, because two constant random variables are independent. (This is a sort of glitch in the matrix.)

Finally, here's a related result — not about noise sensitivity, but about query algorithms.

Theorem 9.14 (OSSS)

Suppose that there is a query algorithm for $f\colon \{-1,1\}^n \to \{-1,1\}$ that always makes at most d queries. Then there exists a coordinate $i \in [n]$ with

$$I_i[f] \geq \frac{\mathrm{Var}[f]}{d^{O(1)}}.$$

This theorem is related directly to what we discussed — another method of proving lower bounds on the number of queries an algorithm makes is by inspecting the influences (if they're all small, then the number of queries has to be large).

While we're here, we'll mention one open problem. The above theorem is for functions which are ±1-valued.

Theorem 9.15

The result should be true for functions $f\colon \{-1,1\}^n \to [-1,1]$ of degree at most d, i.e., if $\deg(f) \leq d$, then

$$I_i[f] \geq \frac{\mathrm{Var}[f]}{d^{O(1)}}.$$

This conjecture says that if you have a bounded function with low degree, you should be able to make the same conclusion about influences. This is a famous conjecture, due to Aaronson and Ambairis. The motivation of it is quantum computing.

§10 March 12, 2024

The main goal for today is to finish the noise sensitivity business we started last week, and give a proof of the Benjamini–Kalai–Schramm theorem. Then we'll start a new topic, where we'll see some applications of what we've already seen to extremal combinatorics.

§10.1 The BKS theorem

Last time, we considered Boolean functions $f\colon \{-1,1\}^n \to \{-1,1\}$, and we defined

$$M(f) = \sum_{i=1}^{n} I_i[f]^2.$$

The main content of last week was to prove some results about when a function is noise sensitive or not. The main theorem we discussed is the BKS theorem:

Theorem 10.1 (BKS)

If $M(f) = o(1)$, then f is noise sensitive.

We discussed the notion of noise sensitivity a lot, and concluded a function is noise sensitive if it mostly lives in high degrees. So really, the content of BKS is saying that if $M(f)$ is small, then there's very little weight on low levels. This is the actual theorem we'll prove.

Theorem 10.2 (BKS)

For all k and all Boolean f, we have

$$W^{=k}[f] \le C^k M(f)\left(\log \frac{1}{M(f)}\right)^{k-1}.$$

It's straightforward to derive the first theorem from the second — if $M(f) = o(1)$, then for every constant k, the right-hand side is also $o(1)$. So we have no weight on low levels.

So we'll actually prove the second theorem. Towards this end, we developed two ideas last time (which we'll briefly recall).

One idea, which we demonstrated last time, is how you can think about low-degree functions as bounded, do some bogus argument, and then fix it using tail bounds. We demonstrated this via the level-d inequality.

§10.1.1 Decoupling

The second idea is decoupling — a general idea that's useful in lots of places. Specified to our purposes, it says the following:

Lemma 10.3 (Decoupling)

For all $f\colon \{-1,1\}^n \to \mathbb{R}$ and k, there exists a partition $I \cup J = [n]$ such that

$$\sum_{\substack{|S|=k \\ |S\cap I|=1}} \widehat{f}(S)^2 \ge \frac{1}{e} W^{=k}[f].$$

Remark 10.4. Recall that we prove this by choosing I and J randomly, such that $|I| \approx \frac{n}{k}$; in particular, I certainly depends on k.

Why is this partition useful? Here's how Dor thinks about it, informally (though what we're going to write down now is incorrect): terms on the left-hand side are kind of like degree-1 things in I, times degree-$(k-1)$ things in J, i.e., $(\sum_{i\in I} a_i x_i)(\sum_{|T|=k-1} a_T z_T)$. So you're sort of able to play separately with the variables of I and J. This isn't actually the case, but it's useful intuition to keep in mind.

§10.1.2 Proof of BKS

Now we're going to prove (the second version of) BKS. The proof we'll give now is different from teh one in the lecture notes (the lecture notes have a proof using a clever trick with integrals; it gives better bounds but is more confusing).

The theorem asks us to bound the level-k weight of f — we need to upper-bound $W^{=k}[f]$. And we don't really know what to do with this, so let's apply decoupling — by the decoupling lemma, we can find a partition (I, J) such that

$$W_{=k}[f] \lesssim \sum_{\substack{|S\cap J|=k-1 \\ |S\cap I|=1}} \widehat{f}(S)^2.$$

Now what we're going to do is think about our probability space $\{-1,1\}^n$ as a product of two probability spaces $\{-1,1\}^I \times \{-1,1\}^J$.

Now if we look at these Fourier coefficients, when we project onto I they're sort of like degree-1 things. So let's define the associated degree-1 functions. First, let's establish some notation: we have an input $x \in \{-1,1\}^n$ in the whole probability space, but we'll write it as $x = (y,z)$ where $y \in \{-1,1\}^I$ and $z \in \{-1,1\}^J$.

Now, as is always the case, we get to look at the individual contribution of each $i \in I$, write our expression as a sum of each of these contributions, and then bound those contributions separately. So let's fix some $i \in I$, and define a function that captures its contributions — naively, we might define

$$f_i' = \sum_{\substack{|S\cap J|=k-1 \\ |S\cap I|=1 \\ i\in S}} \widehat{f}(S)\chi_S(x).$$

But we really want to decouple the space as much as possible. And the only dependency of these coefficients on I is in x_i. So we can just lop that off, and then we only have a dependency on J — so we actually define

$$f_i'(z) = \sum_{|S\cap J|=k-1, |S\cap I|=1 i\in S} \widehat{f}(S)\chi_{S\setminus\{i\}}(z).$$

This looks a lot like the discrete derivative of f along the direction i, but we only look at Fourier coefficients relevant to the sum that we care about. Finally, we actually want to work with a normalized version — so we define

$$f_i = \frac{f_i'}{\|f_i'\|_2}$$

(this is not necessary, but simplifies life later on).

To recap what we did, we used decoupling, looked at the sum that we have, and then broke it down — we looked at the contribution of each i, and now we'll rewrite what we want to bound using this function f_i.

Now let's look at the sum we care about — we have

$$\|f_i'\|_2^2 = \sum_{\substack{|S\cap J|=k-1 \\ S\cap I=\{i\}}} \widehat{f}(S)^2$$

by Parseval. But this is equal to

$$\langle f, y_i f_i'\rangle$$

by Plancherel ($y_i f_i'$ is just the sum of the corresponding subset of Fourier coefficients of f — the effect of multiplication by y_i is removing $S \setminus \{i\}$, giving us x instead of z; and the rest of the things in f are just orthogonal to this). And since we renormalized, we get that

$$\langle f, y_i f_i\rangle = \|f_i'\|_2 .$$

We like squares of 2-norms, so we can square this; then we have

$$\langle f, y_i f_i\rangle^2 = \|f_i'\|_2^2 = \sum_{\substack{|S\cap J|=k-1 \\ S\cap I=\{i\}}} \widehat{f}(S)^2.$$

We've now analyzed the contribution of each i separately, and we want to bound the total contribution, so we sum this over i, giving exactly the sum we want; so we conclude that

$$\sum_{\substack{|S\cap J|=k-1 \\ |S\cap I|=1}} \widehat{f}(S)^2 = \sum_{i\in I}\langle f, y_i f_i\rangle^2.$$

Now for the rest of the proof, we're going to bound each $\langle f, y_i f_i\rangle^2$ separately. Fix $i \in I$, and consider $\langle f, y_i f_i\rangle$, which by definition is $\mathbb{E}_{y,z} f(y, z) y_i f_i(z)$. Now we're going to reshuffle these random variables — so we can rewrite this as

$$\mathbb{E}_z f_i(z) \mathbb{E}_y y_i f(y, z).$$

Now we've gotten to a nice point, where one could actually see the light at the end of the tunnel. If we look at the function f_i', it's degree $k - 1$; that's a constant. And f_i comes from dividing it by some number, so it's still constant degree. And we normalized so that $\|f_i\|_2 = 1$.

So our general principle says that a function with bounded 2-norm and constant degree is morally bounded; nd we exactly have such a thing. So let's pretend for a moment that f_i is actually bounded, and see why this actually gives the bound we want; and then we'll fix that.

Let's consider the dream case where f_i is bounded by some T. Then we can just use the triangle inequality — we have

$$|\langle f, y_i f_i\rangle| \le \mathbb{E}_z |f_i(z)| |\mathbb{E}_y y_i f(y, z)| \le T \mathbb{E}_z |\mathbb{E}_y y_i f(y, z)|.$$

Now, what can we say about $\mathbb{E}_y y_i f(y, z)$? We've put z outside; so we're kind of looking at $f_{J\to z}$; and what we have left is the singleton Fourier coefficient at $\{i\}$, really by definition — and we have

$$\left|\widehat{f_{J\to z}}(\{i\})\right| \le I_i[f_{J\to z}]$$

by the triangle inequality — explicitly, we can pull out the expectation over y_{-i} outside, and then we get

$$|\widehat{f_{J\to z}}(\{i\})| \le \mathbb{E}_{z,y_{-i}} |\mathbb{E}_{y_i} y_i f(y, z)|,$$

and the inner expectation is 0 if y_i is not influential. (If we fix z and everything except y_i, then we're looking at the values of f when $y_i = 1$ and -1.)

So then we get that

$$|\langle f, y_i f_i\rangle| \lesssim T \cdot I_i[f],$$

which is exactly what we wanted (plugging this in gives us the sum of squares of influences).

So that's how the dream would work; this is a good way to try to actually arrive at such thing s(it's very hard to play around with indicators when you don't even know what you're shooting for; but now we know what we're shooting for, namely that the inner expectation should act like an influence, and the outer thing should be kind of bounded).

That's a bogus proof assuming something which is morally correct; now we're going to remove that assumption, and make the dream come true. We wanted our function to be bounded, so let's define an event where it is bounded — let E be the event that $|f_i(z)| \le T$ (where T is something to be determined).

> **Remark 10.5.** If we didn't normalize, then this event would look uglier; this is the only thing that would change.

Now we still want to look at $\langle f, y_i f_i\rangle$, which we can write as above as

$$\langle f, y_i f_i\rangle = \mathbb{E}_z f_i(z) 1_E(z) \mathbb{E}_y y_i f(y, z) + \mathbb{E}_z f_i(z) 1_{\overline{E}}(z) \mathbb{E}_y y_i f(y, z).$$

Let's call these two terms (1) and (2); we'll bound each one of them separately. For (1), this is the dream case — when E happens we can just repeat this argument, and we get a bound of

$$|(1)| \lesssim T \cdot I_i[f]$$

as in the dream case. Meanwhile, for (2), the first thing we'll do is rewrite the inner expectation as a Fourier coefficient of the restriction again — this gives

$$|(2)| = \left|\mathbb{E}_z f_i(z) 1_{\overline{E}}(z) \cdot \widehat{f_{J\to z}}(\{i\})\right|.$$

Now we have some nice stuff, and an indicator of an event that's very rare. We'd like to isolate this, and the only way we know how to do this is by Cauchy–Schwarz; so we're going to use Cauchy–Schwarz to isolate $1_{\overline{E}}$ from the rest. This gives

$$|(2)| \leq \sqrt{\mathbb{E}_z \left|f_i(z)\right|^2 1_{\overline{E}}(z)} \cdot \sqrt{\mathbb{E}_z \left|\widehat{f_{J\to z}}(\{i\})\right|^2}.$$

This indicator isn't fully decoupled from everything else, so we apply Cauchy–Schwarz again, to get that

$$(2) \leq \left(\mathbb{E}_z \left|f_i(z)\right|^4\right)^{1/4} \mathbb{P}[\overline{E}]^{1/4} \sqrt{\mathbb{E}_z \widehat{f_{J\to z}}(\{i\})^2}.$$

The first term is $\|f_i\|_4$, which we can bound using hypercontractivity — we know $\|f_i\|_2 = 1$, so

$$\|f_i\|_4 \leq \sqrt{3}^{k-1} \|f_i\|_2 = \sqrt{3}^{k-1}.$$

For $\mathbb{P}[\overline{E}]$, we have our tail bound for low-degree functions; and we get that

$$(2) \leq \sqrt{3}^{k-1} \cdot 2^{-\Omega(T^{2/(k-1)})} \cdot \sqrt{\mathbb{E}_z \widehat{f_{J\to z}}(\{i\})^2}.$$

So we've got upper bounds on (1) and (2), and plugging this into the thing $(*)$ that we want to bound (and using the fact that $(a+b)^2 \leq 2(a^2+b^2)$) gives

$$(*) \lesssim \sum_{i \in I} T^2 I_i[f]^2 + 3^{k-1} 2^{-\Omega(T^{2/(k-1)})} \mathbb{E}_z \widehat{f_{J\to z}}(\{i\}).$$

And now we're going to separate this sum into two sums to simplify our life. The first term, summed over i, is exactly $T^2 M(f)$. And for the second term, the final point — why we kept this Fourier coefficient flailing around the entire time — is that

$$\sum_i \widehat{f_{J\to z}}(\{i\})^2 \leq \|f_{J\to z}\|_2^2 \leq 1$$

(f is a Boolean function, so its 2-norm is also a Boolean function; note that when you take restrictions of non-Boolean things, 2-norms can become crazy). So we get a bound of

$$(*) \lesssim T^2 M(f) + 3^{k-1} 2^{\Omega(T^{2/(k-1)})}.$$

Big T makes the second term happy but the first term sad, so we want to balance them out; we want them to be roughly equal, so we take $T = C \log^{(k-1)/2} 1/M(f)$, and we're done.

This is one of the more technical proofs that we'll see in the course, but once you see it, you realize that the number of tools we have is quite small, but there are infinite possibilities to combine them, and they're very powerful if you combine them in the right way.

> **Remark 10.6.** To bound $\mathbb{P}[\overline{E}]$, we used a lemma from a previous lecture that if $g\colon \{-1,1\}^n \to \mathbb{R}$ has $\deg(g) \leq d$, then
> $$\mathbb{P}_x[|g(x)| \geq T \|g\|_2] \leq 2^{-\Omega(T^{2/(d-1)})}.$$

> **Remark 10.7.** Dor likes this conceptual view that low-degree functions with bounded 2-norms are morally bounded; this simplifies many things a lot, and if this intuition helps you, then the tail bounds route is the natural thing to try. Otherwise, you have to be very lucky and do Cauchy–Schwarz very carefully.

> **Remark 10.8.** Why is decoupling useful here? Without decoupling we wouldn't have been able to break $\langle f, y_i f_i \rangle$ into an inner and outer expectation. What's really happening is $\mathbb{E}_y y_i f(y,z)$ is measuring whether i is influential, and $f_i(z)$ is measuring some sort of magnitude. And the point is that these are kind of independent, because of the decoupling.

§10.2 Extremal combinatorics

Now we're going to shift to a completely different topic, extremal combinatorics; and we're going to see one of the simplest examples of applications of analysis of Boolean functions in it.

First, here's an overview of the types of problems you can tackle here. The most basic problem in extremal combinatorics is the Erdős–Ko–Rado problem.

Question 10.9. Suppose we have integers n and $k \leq n/2$, and we look at a collection of subsets $\mathcal{F} \subseteq \binom{[n]}{k}$ (a family of subsets of $[n]$ of size k). Suppose we know that $\mathcal{F}$ is intersecting; then how large can it be?

Definition 10.10. We say $\mathcal{F}$ is intersecting if for all $A, B \in \mathcal{F}$, we have $A \cap B \neq \emptyset$.

Example 10.11

We can fix some element $i \in [n]$, and take $\mathcal{F}$ to be all the sets containing i — i.e.,

$$\mathcal{F} = \{A \subseteq [n] \mid |A| = k,\, i \in A\}.$$

Then we get $|\mathcal{F}| = \binom{n-1}{k-1}$.

In fact, it can be proven that this is the best you can do. We're not going to do that. But here's a hint why analysis of boolean functions has anything to do with this type of problem. This construction is a dictatorship — membership in $\mathcal{F}$ only depends on one element. And whenever you have a problem in extremal combinatorics where the answer looks like a dictatorship, then you might expect that what we're about to say may work.

This is an interesting question, but one can ask a more refined question. This is about $|\mathcal{F}|$, but can we prove any *structure* theorem for intersecting families — that any intersecting family of roughly this size has to be close to a dictatorship? (We can prove the above theorem with some cheap tricks, but to get more robust conclusions Dor doesn't know any other approach.)

§10.2.1 Some generalizations

This is the problem we'll study today in this lecture. But before that, we'll mention some more problems of similar flavor that can be tackled using this approach. (Of course, combinatorics has infinitely many problems, so we won't talk about all of them.)

For one thing, instead of talking about intersecting families, we can talk about *t-intersecting families*:

Question 10.12. What if instead of just requiring that all $A, B \in \mathcal{F}$ have $|A \cap B| \neq \emptyset$, we require that $|A \cap B| \geq t$?

Taking inspiration from our earlier example, we're led to consider the family

$$\mathcal{F} = \{A \mid |A| = k, \{i_1, \ldots, i_t\} \subseteq A\}$$

(for some fixed $i_1, \ldots, i_t \in [n]$). This is a nice family, but fortunately or unfortunately, it's not always a tight example. There's another family to consider —

$$\mathcal{F}_{t,r} = \{A \mid |A \cap [t + 2r]| \geq t + r\}$$

(i.e., we look at sets A such that we have almost all of the first $t + 2r$ elements). You can check that this family is also t-intersecting — for any A and B, A contains all but r of the elements of $[t + 2r]$, and the same is true for B; even if all the elements they omit are different, we still get t. We won't get into the sizes of the two families, but it turns out that for different regimes of k and t, one of these two examples is the best.

So things get more complicated.

Another type of problem is the t-avoiding problem.

> **Question 10.13.** What if we replace the condition with $|A \cap B| \neq t - 1$ for all $A, B \in \mathcal{F}$?

This is a weakening of the earlier condition. It turns out (maybe surprisingly) that the best examples you can get are still the same. (OF course if two sets intersect in size at least t, they don't intersect in size $t - 1$; but the converse is not true. But it turns out the answers are the same, at least for some regimes of parameters.)

A final problem, which also looks similar:

> **Question 10.14.** For $n \geq sk$, how large can $\mathcal{F} \subseteq \binom{[n]}{k}$ be if it doesn't contain s pairwise disjoint sets $\{A_1, \ldots, A_s\}$?

We won't discuss these problems; but the point is we'll see some very nice tools, and there are extensions of them that solve these problems (and others). In some sense, the best indicator of whether what Dor will say now is relevant to a problem is if the suspected optimal example looks like a dictatorship in some sense.

§10.3 Translation

In all the discussion so far, we looked at subsets of size k of $[n]$. We don't really like this — we like product spaces and stuff like that. So first we're going to cheat; instead of looking at subsets of size k, we'll look at a product measure that makes sense (there's a tight relation between these, but we won't do anything formal; we'll just change the problem instead).

So instead of $\binom{[n]}{k}$, we'll take $p = \frac{k}{n}$, and think about the Boolean cube $\{0, 1\}^n$ with the p-biased measure $\mu_p^{\otimes n}$. The point of this is that if we sample a point x here, then its expected number of 1's is k; and the standard deviation is roughly $\sqrt{k}$. So if we take a point here and take its support, this morally corresponds to a subset of size k. This lets us translate the subset language to a product set language, and this is what we'll stick with.

Second, what's a family of subsets $\mathcal{F} \subseteq \binom{[n]}{k}$? This is really just a Boolean-valued function on this space, and we like Boolean-valued functions, so that's fine — this will correspond to a function $f\colon (\{0, 1\}^n, \mu_p^{\otimes n}) \to \{0, 1\}$.

(Right now, we're just writing down a dictionary of how to translate from subsets to product sets.)

And the normalized size of $\mathcal{F}$ is

$$\frac{|\mathcal{F}|}{\binom{n}{k}} = \mathbb{P}_{|A|=k}[A \in \mathcal{F}].$$

In Boolean function terms, this is just the average of f — i.e.,

$$\mu_p(f) = \mathbb{E}_{x \sim \mu_p^{\otimes n}} f(x).$$

Finally, what does intersecting mean? In subset language, we need that all $A, B \in \mathcal{F}$ have $A \cap B \neq \emptyset$. In our new language, this says that whenever $f(x) = f(y) = 1$, there exists i for which $x_i = y_i = 1$.

So we've translated our problem to product spaces; but now we get to our second issue.

§10.4 The p-biased cube

Throughout this course, we've only talked about the Boolean cube with the uniform distribution; so now we need a disclaimer about what happens when we look at the p-biased distribution. We won't prove things; we'll just state the results.

It turns out that there are two regimes. The first regime is when p is bounded away from both 0 and 1 — i.e., $\zeta \leq p \leq 1 - \zeta$ (for some constant $\zeta > 0$). In this regime, nothing of interest happens (or rather, many things happen) — all of what we saw generalizes. In particular, the KKL theorem, hypercontractivity, Friedgut, and so on all still apply. The only catch is that the constants will be worse (they'll depend on ζ); but we shouldn't care about that. So if p is bounded away from 0 and 1, everything generalizes and we don't need to worry about anything.

This is the regime in which we'll work now. But because we started this discussion, let's talk about the other regime — where p is close to 0 or 1. For concreteness, let's suppose that $p = 1/\sqrt{n}$. Here everything breaks.

This regime is still interesting (sometimes more interesting, depending on what you're trying to solve), but all the tools we saw so far don't apply (e.g. hypercontractivity, KKL, Friedgut). We may talk about this more in the end of the course, but for now we'll only work with p bounded away from 0 and 1.

§10.5 The junta approximation theorem

We'll now prove the following theorem.

Theorem 10.15 (Dinur–Friedgut)

Suppose that we have $\mathcal{F} \subseteq \{0,1\}^n$ which is intersecting, and $\zeta \leq p \leq \frac{1}{2} - \zeta$. Then for every $\varepsilon > 0$, there exists J (depending on ζ and ε) such that $\mathcal{F}$ is nearly contained in an intersecting J-junta — i.e., there exists a J-Junta $\mathcal{J}$ such that $\mathcal{J}$ is intersecting, and

$$\mu_p(\mathcal{F} \setminus \mathcal{J}) \leq \varepsilon.$$

Remark 10.16. What does a J-junta look like in this case? It's the same thing — we're phrasing this in terms of collections, but you can think about their indicator vectors.

Let's look at this again. We saw one example of an intersecting family; and if you have an intersecting family, any subset of it is also going to be intersecting. So the best thing you can hope for is that you're contained in some special intersecting family. And that's what this theorem is telling you.

We don't have time to prove this today, but we'll mention something that'll be useful.

Definition 10.17. A family $\mathcal{F} \subseteq \{0,1\}^n$ is called *monotone* if whenever $x \in \mathcal{F}$ and $x \leq y$, we also have $y \in \mathcal{F}$.

If we have a function, we sometimes want to turn it into something monotone; and we can do that by taking the *upper shadow.*

Definition 10.18. If $\mathcal{F} \subseteq \{0,1\}^n$, the *upper shadow* of $\mathcal{F}$ is defined as

$$\mathcal{F}\uparrow = \{y \in \{0,1\}^n \mid \exists x \in \mathcal{F} \text{ s.t.} x \leq y\}.$$

In other words, the upper shadow of $\mathcal{F}$ is the collection of points y which are above *some* point in $\mathcal{F}$.

Claim 10.19 — For any $\mathcal{F} \subseteq \{0, 1\}$, the upper shadow $\mathcal{F} \uparrow$ is monotone.

Proof. If $y \leq z$ and $y \in \mathcal{F} \uparrow$, then there exists $x \in \mathcal{F}$ with $x \leq y$; but then $x \leq z$ as well (by transitivity), so $z \in \mathcal{F} \uparrow$. □

So this is a way to make any given family monotone. The reason this is relevant to us is the following thing (which we'll just state for now, and prove next time):

Claim 10.20 — If $\mathcal{F}$ is intersecting, then $\mathcal{F} \uparrow$ is also intersecting.

So in words, being intersecting is closed under this monotone closure operation.

This is a very nice feature which, as we'll see later, tells us that it's enough to work with monotone ufnctions. And we'll see that of the other properties we've talked about, the first and third are also closed under taking monotone closure; this is an important feature of the two problems. (The second is not, which is why it is more complicated.)

§11 March 14, 2024

§11.1 The Dinur–Friedgut theorem

Today our main goal is to prove a theorem we stated last time, which makes the following assertion:

Theorem 11.1 (Dinur–Friedgut)
Let $\zeta < p < \frac{1}{2} - \zeta$, and suppose that $\mathcal{F} \subseteq \{0, 1\}^n$ is intersecting. Then for all $\varepsilon > 0$, there exists $J \in \mathbb{N}$ (depending on ε and ζ) such that there exists a J-junta $\mathcal{G} \subseteq \{0, 1\}^n$ such that:

- $\mathcal{G}$ is intersecting;
- $\mathcal{G}$ almost contains $\mathcal{F}$ — i.e., $\mu_p(\mathcal{F} \setminus \mathcal{G}) \leq \varepsilon$.

The significance is that if you try to come up with examples of intersecting families, the ones you naturally come up with are juntas. And this theorem says there's a good reason for that — those are essentially all the good examples.

The proof of this theorem is very nice; it'll use a bunch of tools we've already seen, and a bunch of tools we'll see today. But before we do anything fancy, we'll start by making the following observation.

Claim 11.2 — It is enough to prove the theorem for *monotone* families $\mathcal{F}$.

Proof. Given any $\mathcal{F}$, we can replace it with $\mathcal{F}' = \mathcal{F} \uparrow$ — the upwards closure of $\mathcal{F}$, which we defined as

$$\mathcal{F} \uparrow = \{y \in \{0, 1\}^n \mid \text{exists } x \in \mathcal{F} \text{ with } x \leq y\}.$$

Last time, we stated that if $\mathcal{F}$ was intersecting, then $\mathcal{F}'$ is also intersecting. To prove this, suppose we have $y, y' \in \mathcal{F}'$; we want to show there exists i where they're both 1. To see this, we can drop down to the elements $x, x' \in \mathcal{F}$ that produced them — i.e., such that $x \leq y$ and $x' \leq y'$. Then since $\mathcal{F}$ is intersecting, there is some i at which $x_i = x'_i = 1$; and then $y_i = y'_i = 1$ as well.

And of course $\mathcal{F}'$ is monotone (we proved this last time).

So if we've proven the theorem for monotone functions, then we can apply it on $\mathcal{F}'$ — there exists a J-junta $\mathcal{G}$ such that $\mu_p(\mathcal{F}' \setminus \mathcal{G}) \leq \varepsilon$. And now we're done, because $\mathcal{F} \subseteq \mathcal{F}'$, and therefore $\mu_p(\mathcal{F} \setminus \mathcal{G}) \leq \mu_p(\mathcal{F}' \subseteq \mathcal{G}) \leq \varepsilon$. □

So from now on, we'll assume that $\mathcal{F}$ (in the theorem) is monotone.

We'll put that aside for now; monotonicity will enter the picture in about 20 minutes.

§11.2 Quasirandomness

At a very high level, the way that the proof of the theorem will go is that we're looking for a junta; this means we want to decompose our family into a constant number of families, where each is either full or not. And how do we come up with this? We have one tool at our disposal talking about juntas, which is Friedgut; this will play a role, but not in the way you expect.

Nevertheless, we have to find a junta, and the notion of quasirandomness is going to be helpful in that regard.

Definition 11.3. We say that a function $f\colon\{0,1\}^n \to \{0,1\}$ is (r,ε)-*quasirandom* with respect to p if for all $R \subseteq [n]$ with $|R| \le r$, and all $z \in \{0,1\}^R$, we have

$$|\mu_p(f_{R\to z}) - \mu_p(f)| \le \varepsilon.$$

In words, quasirandomness means that whatever coordinates we restrict to, the expectation of f shouldn't change too much. So over all sets of size R and all possible restrictions, the expectations of f with and without the restriction are about the same.

As an example, let's think about $p = \frac{1}{2}$ and $r = 1$. Then we have

$$\widehat{f}(\{i\}) = \mathbb{E} f(x)(-1)^{x_i} = \frac{1}{2}\left(\mu(f_{i\to 1}) - \mu(f_{i\to 0})\right).$$

And so the condition is almost equivalent to all the singleton Fourier coefficients being small; more generally, this is morally equivalent to all the Fourier coefficients of size at most r being small. (This actually is true for more general p as well, but we haven't defined the p-biased characters, so it's more annoying to state.) But this motivates this.

§11.3 A regularity lemma

Of course, not every function is quasirandom — e.g., a dictatorship or a junta is not quasirandom (there's going to be some variable that makes a difference). But what the following lemma says is that after you restrict constantly many variables, you basically are quasirandom.

Lemma 11.4 (Regularity lemma)

For all r and all $\varepsilon, \zeta, \delta > 0$, there exists some $J \in \mathbb{N}$ such that the following holds: for any $\zeta < p < \frac{1}{2} - \zeta$ and any function $f\colon\{0,1\}^n \to \{0,1\}$, one can find a set of variables $T \subseteq [n]$ such that:

(1) $|T| \le J$.

(2) We have $\mathbb{P}_{z\sim\mu_p^T}[f_{T\to z}$ is not (r,ε)-quasirandom$] \le \delta$.

(This should be true even with $\zeta < p < 1-\zeta$, but it doesn't matter.)

In other words, there's a set T which is not too big such that once we *restrict* all of the variables in T, the remaining function is almost always quasirandom.

When we say words like 'quasirandom,' we mean that what we expect to be able to show is functions satisfying this act random in some sense. And what this lemma says is that inside any big Boolean function, we can find this quasirandom structure. This is kind of like the Szemerédi regularity lemma for graphs, and the proof and bound are going to be very similar.

Proof. We're going to start with $T = \varnothing$. The idea of the proof is — suppose f violates this condition. Then we can find some restrictions violating it. Then we can take these coordinates, insert them into T, and try again. And we keep doing this.

The point is, when do you halt? And the trick is to define some natural potential function.

So we define a potential function

$$\mathcal{P}(T) = \sum_{z \sim \mu_p^T} \left| \mu_p(f_{T \to z}) - \mu(f) \right|^2 .$$

Why is this natural to take? The reason is when we find violations, we've found some large gaps between what the restriction gives and what the average is. So it makes sense this will increase as we find more and more gaps. (This is how you come up with these things — you think about what's the interesting thing, and then you try it.)

Note that $\mathcal{P}(\emptyset) = 0$, and $\mathcal{P}(T) \leq 1$ for all T.

We'll show that if T fails to satisfy the conditions of the theorem, then we can find T' (which will contain T) such that:

- T' is not too much larger than T — the bound isn't that important, but the specific bound is
$$\left|T'\right| \leq r \cdot 2^{2^{|T|}}.$$
- More importantly, $\mathcal{P}(T') \geq \mathcal{P}(T) + \delta \zeta^r \varepsilon^2$ (so the potential increases by some fixed quantity, which is not very important).

What's important is that when we look at this gain, this means we can't run this process forever — we can run for at most $1/(\delta \zeta^r \varepsilon^2)$ iterations. At this point, we'll have T', whose size is some tower of 2's whose height only depends on the fixed parameters; so we'll be done (taking J to be this tower of 2's).

Before we do this, let's do a warmup:

> **Claim 11.5** — Whenever $T' \supseteq T$, we have $\mathcal{P}(T') \geq \mathcal{P}(T)$.

Proof. We have $\mathcal{P}(T') = \mathbb{E}_{z \in \mu_p^{T'}} (\mu_p(f_{T' \to z}) - \mu_p(f))^2$. And we can split the variables in T' into two parts — those in T and those not in T — to rewrite this as

$$\mathbb{E}_{y \sim \mu_p^T, y' \sim \mu_p^{T' \setminus T}} (\mu_p(f_{T \to y, T' \setminus T \to y'}) - \mu_p(f))^2.$$

And now we can pull the expectation over y' inside, using the fact that $\mathbb{E}X^2 \geq (\mathbb{E}X)^2$ (e.g., Jensen or Cauchy–Schwarz), to get that this is at least

$$\mathbb{E}_{y \sim \mu_p^T} \left(\mathbb{E}_{y' \sim \mu_p^{T' \to T}} \mu(f_{T \to y, T' \to y'}) - \mu_p(f) \right)^2 .$$

And the first term inside is just $\mu_p(f_{T \to y})$ (we're first restricting the variables in T', but then we're averaging over all possible restrictions, which nullifies that). So this is precisely $\mathcal{P}(T)$. □

So that's the warmup. And there's exactly one inequality we used here, which is $\mathbb{E}X^2 \geq (\mathbb{E}X)^2$. The whole point of the proof is to use the violation of quasirandomness to get a gain here.

We first need to describe how to construct T' from T. Suppose that T fails, and let

$$Z = \{z \in \{0,1\}^T \mid f_{T \to z} \text{ is not quasirandom}\}$$

be the set of all restrictions of T where quasirandomness fails. We know that $|Z| \leq 2^{|T|}$ (this is where the $2^{|T|}$ term will come from).

What does it mean that $f_{T \to z}$ is not quasirandom? This means there's some further set that shows it's not quasirandom — for each $z \in Z$, we can choose some restriction $R_z \subseteq [n] \setminus T$ and $w_z \in \{0,1\}^{R_z}$ showing that quasirandomness fails, meaning that $|R_z| \leq r$ and

$$|\mu_p(f_{T \to z, R_z \to w_z}) - \mu_p(f_{T \to z})| \geq \varepsilon.$$

(We've just spelled out what it means to be not quasirandom — if the restricted function $f_{T \to z}$ is not quasirandom, then there exists some further subset of the variables and a fixing of them that makes the average tilt by at least ε.)

For each $z \in Z$ we can find such a set R_z. And we also know the measure of Z is not that small, because T failed, which means that many restrictions fail to b equasirandom — specifically, we have $\mu_p(Z) \geq \delta$ (otherwise we would have been done).

So with all of this in mind, what's the most sensible way to choose T'? We take T along with each one of these sets R_z — so we define

$$T' = T \cup \bigcup_{z \in Z} R_z.$$

(We already know by the warmup that the more variables we take, the better our situation will be; so it makes sense to just take all of these.)

Now we can bound $|T'|$ — we have

$$|T'| \leq |T| + |Z| \cdot r \leq (r+1) \cdot 2^{|T|}$$

(since $|Z| \leq 2^{|T|}$). And now we want to show that the potential increases, so let's write down the change in potential — we have

$$\mathcal{P}(T') - \mathcal{P}(T) = \mathbb{E}_{y \sim \mu_p^{T'}} (\mu_p(f_{T' \to y}) - \mu(f))^2 - \mathbb{E}_{z \sim \mu_p^T} (\mu_p(f_{T \to z}) - \mu_p(f))^2.$$

And now we're going to use the same trick from earlier — breaking y into two parts, one which is over T and one which is the rest. This allows us to rewrite this as

$$\mathbb{E}_{z \sim \mu_p^T} \left[\left(\mathbb{E}_{y' \sim \mu_p^{T' \setminus T}} (\mu_p(f_{T \to z, T' \setminus T \to y}) - \mu(f))^2 \right) - (\mu_p(f_{T \to z}) - \mu_p(f))^2 \right].$$

So the difference in potentials is an expectation over z of some difference-looking thing; and we're going to analyze the inner expression for each z. Let

$$X_z = \left(\mathbb{E}_{y' \sim \mu_p^{T' \setminus T}} (\mu_p(f_{T \to z, T' \setminus T \to y}) - \mu(f))^2 \right) - (\mu_p(f_{T \to z}) - \mu_p(f))^2$$

denote this inner expression.

Claim 11.6 — For each z, we have $X_z \geq 0$.

Proof. If you look at the first expectation and put the expectation over y' inside, then you get exactly the other term; and putting the expectation inside can only decrease things. (This is the same application of $\mathbb{E}X^2 \geq (\mathbb{E}X)^2$ as from earlier.) □

This is good, because at least we've managed to show that in this convoluted way of writing the difference, we still didn't lose the fact that potential increases when we go to T'. The next claim says that if z is one of the annoying ones, then not only is X_z nonnegative, but it's actually also bounded below.

> **Claim 11.7** — For each $z \in Z$, we have $X_z \geq \zeta^r \varepsilon^2$.

Proof. We have y', but $T' \setminus T$ contains a bunch of variables that z doesn't care about — z only cares about R_z (the rest of the elements are there for other reasons). So we can write

$$X_z = \mathbb{E}_{w \in \mu_p^{R_z}, w' \sim \mu_p^{T \setminus (T \cup R_z)}} (\mu_p(f_{T \to z, R_z \to w, T' \setminus (T \cup R_z) \to w'}) - \mu_p(f))^2 - (\mu_p(f_{T \to z}) - \mu(f))^2.$$

Now we're going to use Jensen again to get rid of the unnecessary variables — we can push the expectation over w' inside. This is only going to decrease X_z; so we have

$$X_z \geq \mathbb{E}_{w \sim \mu_p^{R_z}} (\mu_p(f_{T \to z, R_z \to w}) - \mu(f))^2 - (\mu_p(f_{T \to z}) - \mu_p(f))^2.$$

And now it's time to make some nice observations. Define a random variable

$$X_z(w) = \mu_p(f_{T \to z, R_z \to w}) - \mu(f).$$

(where we think of z as fixed, and w as bieng sampled).

In terms of the random variable $X_z(w)$, the first expectation is just $\mathbb{E} X_z(w)^2$, while the second is $(\mathbb{E} X_z(w))^2$ (we've done this trick several times, where we shove the expectation inside and the second term is what we get). And so

$$X_z = \mathbb{E}_w X_z(w)^2 - (\mathbb{E}_w X_z(w))^2 = \mathrm{Var}(X_z(w)).$$

Now we're almost done — intuitively, a violation to quasirandomness means you have variation somewhere, so if we've managed to get a variance that should be good. If we just look at the definition of the variance, it's

$$\mathrm{Var}[X_z(w)] = \mathbb{E}\left[X_z(w) - \mathbb{E}_w X_z(w)\right]^2 = \mathbb{E}_w \left|\mu(f_{T \to z, R_z \to w}) - \mu(f_{T \to z})\right|^2$$

(because the expectation of the random variable is $\mu(f_{T \to z}) - \mu(f)$, and the random variable itself is $\mu(f_{T \to z, R_z \to w}) - \mu(f)$).

And we claim that this is at least $\zeta^r \varepsilon^2$. To see this, by our choice of R_z, this difference is larger than $|eps$ for at least one setting of w_z. So now we sample w ;what's the chance that we pick this exact w_z? Well, it's at least $p^r \geq \zeta^r$. And when we look at this special one, the difference squared is at least ε^2. (For any other w, the square is nonnegative; we don't worry about them.) This completes the proof. □

And now we're done — because plugging in these two claims to our expression for the difference in potentials, we get that

$$\mathcal{P}(T') - \mathcal{P}(T) \geq \mathbb{E}_z \left[1_{z \notin Z} 0 + 1_{z \in Z} \zeta^r \varepsilon^2\right] \geq \delta \zeta^r \varepsilon^2,$$

and we're done. □

> **Remark 11.8.** This bound is somewhat ridiculous (we run for $\delta \zeta^r \varepsilon^2$ steps, and each exponentiates the size of T), but it is certainly constant.

§11.4 Sharp thresholds

We have to find a junta, and the way we're going to find a junta is actually by using this regularity lemma. But still, Friedgut's junta theorem is going to enter the picture, and that's now.

Recall that we've seen it suffices to prove the theorem for monotone families, and we've proven the regularity lemma. It stands to reason that now one should say something about monotone families that are quasirandom, and that is the following lemma.

Lemma 11.9

For all $\zeta, \alpha > 0$, there exists $r \in \mathbb{N}$ and $\varepsilon > 0$ such that if $\zeta < p < \frac{1}{2} - \zeta$ and $f\colon \{0,1\}^n \to \{0,1\}$ is monotone and (r, ε)-quasirandom, and $\mu_p(f) \geq \alpha$, then $\mu_{p+\zeta/2}(f) \geq 0.9$.

So we have a quasirandom family; the way you should read this is as the average of f is at least 1% and f is monotone. Then this says that f has a sharp threshold, in the sense that if we slightly increase p then we already end up close to 1.

Ignoring all the quantifiers, the way to read the lemma is that any monotone quasirandom function has a sharp threshold — we increase from a little bit to almost 1 in a very narrow window (of p).

Proof. The proof is very cool. It's by contradiction — suppose that $\mu_{p+\zeta/2}(f) < 0.9$. (We're going to choose r and ε later.)

If we consider the difference, we have

$$\frac{\mu_{p+\zeta/2}(f) - \mu_p(f)}{(p + \frac{\zeta}{2}) - p} \leq \frac{2}{\zeta}.$$

(The numerator is at most 1, and the denominator is $\zeta/2$.) Now comes undergrad calculus — by the mean value theorem, the left-hand side is the average value of the derivative, so there exists some $p' \in [p, p + \frac{\zeta}{2}]$ at which

$$\frac{d\mu_p(f)}{dp}(p') = \frac{\mu_{p+\zeta/2}(f) - \mu_p(f)}{(p + \frac{\zeta}{2}) - p} \leq \frac{2}{\zeta}.$$

But we know what this derivative is — it's exactly the total influence at p'. So by Russo–Margulis, we have

$$I_{p'}[f] = \frac{d\mu_p(f)}{dp} \leq \frac{2}{\zeta}.$$

This means we've got that at *some* point in this interval, the total influence of our function is small.

And now we can unravel the chain — Friedgut tells us that a function with small total influence is close to being a junta. So by Friedgut, there exists J (depending only on ζ and α) and a function $g\colon \{0,1\}^n \to \{0,1\}$ such that g is a J-junta and f is close to g, i.e.,

$$\mathbb{P}_{x \sim \mu_p^n}[f(x) \neq g(x)] \leq \frac{\alpha}{1000}.$$

(The 1000 is just a random number.)

Now here's how we choose r and ε — we just take $r = |J|$ and $\varepsilon = \alpha/4$. Now let's see what we can say about the junta g. First, we claim that the average of g is not too close to 1 — the point is that g is 1 only if f is 1 or $g \neq f$, giving

$$\mu_{p'}(g) \leq \frac{\alpha}{1000} + \mu_{p'}(f) \leq \frac{\alpha}{1000} + \mu_{p+\frac{\zeta}{2}}(f) \leq \frac{\alpha}{1000} + 0.9 \leq 0.99$$

(using monotonicity — we have $p' \leq p + \zeta/2$, and so increasing p' to $p + \zeta/2$ is only going to increase the p-biased average of f).

Now let $R \subseteq [n]$ be the set of coordinates that g depends on (so $|R| \leq J = r$), and choose $x \sim \mu_{p'}^R$. Now we're going to consider two events. The first event is — g is a R-junta, so $g_{R\to x}$ is a constant function; and we'll consider the event that $g_{R\to x} \equiv 0$. And secondly, generally speaking f and g are close; we'll consider the event that even after the restriction, f and g are still close. So we consider the event E_1 defined as $g_{R\to x} \equiv 0$ and E_2 defined as

$$\mathbb{P}_{y \sim \mu_p^{[n]\setminus R}}[f_{R\to x}(y) \neq g_{R\to x}] \leq \frac{\alpha}{2}.$$

(Both of these events are functions of x.)

What we're going to show is that we can find x satisfying both these events simultaneously. First, we can stare at E_1 — $g_{R\to x}$ is either the constant 1 function or the constant 0 function. And we know $\mu_{p'}(g) \leq 0.99$, so g isn't almost always 1; and this means

$$\mathbb{P}[E_1] \geq 1 - 0.99 = 0.01.$$

(Again, this is just because $g_{R\to x}$ is constant, and $\mu_{p'}(g)$ is exactly the probability that it's the constant 1.) Meanwhile, we'll upper-bound $\mathbb{P}[\overline{E_2}]$ — we can look at this probability as a random variable V_x. We know V_x is nonnegative, and $\mathbb{E}[V_x] = \mathbb{P}[f(x) \neq g(x)] \leq \frac{\alpha}{1000}$. So by Markov, we have

$$\mathbb{P}[\overline{E_2}] \leq \frac{\alpha/1000}{\alpha/2} = \frac{1}{500}.$$

Therefore $\mathbb{P}[E_2]$ is very close to 1.

(If we have a bunch of events that you want to hold, if they're all close to 1 you're happy; but if one of them is small, you can still tolerate it. If two of them are small, then you're in bad shape.)

So we can say that $\mathbb{P}[E_1 \cap E_2] \geq 0.01 - \frac{1}{500} > 0$. Now pick x satisfying both E_1 and E_2. And let's stare at this x. We know $g_{R\to x}$ is completely 0, and g and f are close when we restrict them; that means f is very close to being 0 on that x, i.e.,

$$\mathbb{P}_{y\sim\mu_p^{[n]\setminus R}}[f_{R\to x}(y) \neq 0] \leq \frac{\alpha}{2}.$$

In other words, we have

$$\mu_{p'}(f_{R\to x}) \leq \frac{\alpha}{2}.$$

Now we'll use monotonicity — we don't know anything about p', but thanks to monotonicity, when we take p' and reduce it to p, the average of the function just decreases — so we have $\mu_p(f_{R\to x}) \leq \alpha/2$ as well.

And this is a contradiction to the fact that f is (r, ε)-quasirandom — we just found that generally speaking $\mu_p(f) = \alpha$, but we found this restriction giving us $\alpha/2$, so the difference here is at least $\alpha/2$. □

This is a very cute argument using a whole bunch of stuff; but what we'll use is just the statement itself, which is very intuitive if you think about it. If you take some random monotone function (whatever that means), you'll see that it's going to have a sharp threshold (if you look at some layer of the cube and then the layer above it, it's going to very rapidly go up). And what this lemma tells you is you don't really need randomness, but just this more combinatorial notion of quasirandomness. That's a nice result.

It's not clear yet how on earth this is related to intersecting families. It is clear that it's related to monotonicity and to quasirandomness, and with 10 more minutes we could see the whole thing, but this will have to wait for next week. Briefly, what we're going to prove using this lemma is that if you have a quasirandom family, it cannot be intersecting.

§12 March 19, 2024

Today we'll do two things. One is to finish what we started last time — to prove that intersecting families are almost contained in juntas. (This will take us 20–30 minutes.) The second thing is that we're going to start discussing the *invariance principle*, which is heavy armor and is very nice.

§12.1 Intersecting families

In the first half of the class, we'll prove the following theorem:

Theorem 12.1 (Dinur–Friedgut)

For all $\varepsilon, \zeta > 0$, there exists $J \in \mathbb{N}$ such that if $\mathcal{F} \subseteq \{0,1\}^n$ is an intersecting family and $\zeta \leq p \leq \frac{1}{2} - \zeta$, then there exists $\mathcal{G} \subseteq \{0,1\}^n$ such that $\mathcal{G}$ is an intersecting J-junta and $\mu_p(\mathcal{F} \setminus \mathcal{G}) \leq \varepsilon$.

We've made one observation last time about this problem — it's enough to prove this when $\mathcal{F}$ is monotone. Then we started developing some tools. One tool we developed is the regularity lemma, which we'll briefly restate:

Lemma 12.2

For every $r \in \mathbb{N}$ and $\varepsilon > 0$, there exists J such that for every $f\colon \{0,1\}^n \to \{0,1\}$, there exists $T \subseteq [n]$ such that $|T| \leq J$,

$$\mathbb{P}_{z \sim \mu_p^T}[f_{T \to z} \text{ is } (r, \varepsilon)\text{-quasirandom}] \geq 1 - \varepsilon.$$

In other words, we can find a reasonably small set T such that when we restrict T, we most likely get a quasirandom family. The proof is very similar to the Szemerédi regularity lemma.

Then we started talking about quasirandomness, and proved a sharp threshold result:

Theorem 12.3

If $\mathcal{F}$ is monotone, $\mu_p(\mathcal{F}) \geq \alpha$, and $\mathcal{F}$ is (r, ε)-quasirandom (for appropriate r and ε chosen based on ζ and α), then $\mu_{p+\zeta/2}(\mathcal{F}) \geq 0.9$.

(We're being a bit sloppy with the quantifiers here; but the point is that if have nontrivial measure at p and we go just a bit beyond p, then we get really large measure.)

Now everything's going to come together, and we'll see how these things give the theorem.

The amazing thing is that it starts with the following easy claim (the easy proof of Erdős–Ko–Rado):

Claim 12.4 — Suppose that $\mu_{1/2}(\mathcal{F}) + \mu_{1/2}(\mathcal{H}) > 1$. Then there exist $x \in \mathcal{F}$ and $h \in \mathcal{H}$ that are disjoint.

(This means the subsets of $[n]$ corresponding to x and h — namely $\operatorname{supp}(x)$ and $\operatorname{supp}(h)$ — are disjoint.)

Proof. Let's say we have some $x \in \mathcal{F}$. Then there's a very special point that we know for sure is not in $\mathcal{H}$ — namely $\overline{x}$ (where we flip all 0's to 1's and vice versa). So in particular, when $x \notin \mathcal{F}$, we have $\overline{x} \notin \mathcal{H}$. This gives tension — every point in $\mathcal{F}$ discounts one in $\mathcal{H}$ — and this gives the conclusion.

In other words, let $x \sim \mu_{1/2}^n$, and let $\overline{x} = 1 - x$. Then formally speaking, the main observation (and what's special about 1/2) is that the marginal distribution of $\overline{x}$ is $\mu_{1/2}^n$ as well — each coordinate of $\overline{x}$ is also 1 with probability 1/2. (This is only true for the uniform distribution — if x is p-biased then $\overline{x}$ is $(1-p)$-biased, but for 1/2 they're the same.)

So then by linearity of expectation

$$\mathbb{E}_x\left[1_{x \in \mathcal{F}} + 1_{\overline{x} \in \mathcal{H}}\right] = \mu_{1/2}(\mathcal{F}) + \mu_{1/2}(\mathcal{H}) > 1.$$

So the expectation of some random variable is more than 1, which means the random variable is, with positive probability, more than 1 — this means there exists x such that $1_{x \in \mathcal{F}} + 1_{\overline{x} \in \mathcal{H}} > 1$, and this gives disjoint elements $x \in \mathcal{F}$ and $\overline{x} \in \mathcal{H}$. □

This is a nice and simple proof with no relation to what we saw so far; but now things are going to start clicking. Note that this claim is talking about families with very large measure (close to $1/2$). And we have some tools that allow us to go from some p to slightly higher p and jump from a little bit to a lot. Now we're going to combine these things and get some nice results.

Claim 12.5 — For every $\alpha, \zeta > 0$, there exists (r, ε) such that if $\zeta \leq p \leq \frac{1}{2} - \zeta$ and $\mathcal{F}, \mathcal{H} \subseteq \{0,1\}^n$ are monotone and (r, ε)-quasirandom, and $\mu_p(\mathcal{F}), \mu_p(\mathcal{H}) \geq \alpha$, then there exist $x \in \mathcal{F}$ and $h \in \mathcal{H}$ which are disjoint.

(Now it's going to be clear why we keep p bounded away from $\frac{1}{2}$ — this is the only reason.)

Remark 12.6. The (r, ε)-quasirandomness is with respect to p (in both this and the earlier statement).

Proof. By the sharp threshold theorem for quasirandom families, we have that $\mu_{p+\zeta/2}(\mathcal{F}), \mu_{p+\zeta/2}(\mathcal{H}) \geq 0.9$. But now notice that p is bounded away from $1/2$, so $p + \zeta/2$ is still less than $1/2$; and since $\mathcal{F}$ is monotone and $p + \zeta/2 \leq 1/2$, when we increase $p + \zeta/2$ to $1/2$ the measure of $\mathcal{F}$ only increases, and we get that

$$\mu_{1/2}(\mathcal{F}) \geq \mu_{p+\zeta/2}(\mathcal{F}) \geq 0.9,$$

and similarly $\mu_{1/2}(\mathcal{H}) \geq 0.9$.

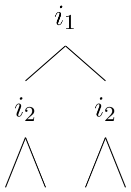
So we've used sharp thresholds to go up a bit in probability and increase our measures by a lot; and then we can simply use the claim to find disjoint $x \in \mathcal{F}$ and $h \in \mathcal{H}$ in these families. □

Now we're going to combine this with the regularity lemma to finish.

Proof of theorem. There's too many ε's on the board; now the only ε is going to be the one in the theorem. So fix ε and ζ to be the ones in the theorem statement, and take $\alpha = \varepsilon/2$. Now take (r', ε') from the second claim. Now we apply the regularity lemma to find J. (These parameter games aren't important; the important thing comes now.)

We want to prove the theorem, so fix a family $\mathcal{F}$. Then we'll apply the regularity lemma to find a set T — by the regularity lemma, we can find $T \subseteq [n]$.

Visually, imagine a tree where we read off all the variables of T, one by one: for example, say $T = \{i_1, \ldots, i_J\}$. Then at the top node we read i_1; we either go to the left or right depending on x_{i_1}; then we read off i_2, and so on. So each one of the leaves in the trees corresponds to an assignment $z \in \{0,1\}^T$.



Now we want to design a family $\mathcal{G}$ which is a T-junta — meaning after we read all variables of T, we know whether our element is in the family or not. To do so, we're going to look at this tree — we need to do something based on x_T. And we know that when we restrict the coordinates of T, most likely we're quasirandom. We're only going to include x's where the corresponding restriction is quasirandom. Also, sometimes when we hit a leaf the function will be really close to 0, and we don't know anything about such functions, so we'll exclude them — let $f = 1_{\mathcal{F}}$, and define

$$\mathcal{G} = \{x \in \{0,1\}^n \mid f_{T \to x_T} \text{ is } (r', \varepsilon')\text{-quasirandom, and } \mu_p(f_{T \to x_T}) \geq \alpha\}.$$

This is the junta we'll define.

Now we need to prove two assertions — one is that $\mathcal{G}$ nearly contains $\mathcal{F}$, and the other is that $\mathcal{G}$ is intersecting. We also need to prove that $\mathcal{G}$ is a J-junta; this is true simply because $|T| \leq J$, and the formula we used to define $\mathcal{G}$ only depends on the coordinates in T. (Whether x belongs to $\mathcal{G}$ or not only depends on its coordinates in T, and $|T| \leq J$.)

Next, we'll check that $\mathcal{F}$ is nearly contained in $\mathcal{G}$, meaning that $\mu_p(\mathcal{F} \setminus \mathcal{G}) \leq \varepsilon$. To prove this, we can write

$$\mu_p(\mathcal{F} \setminus \mathcal{G}) = \mathbb{E}_x 1_{x \in \mathcal{F} \setminus \mathcal{G}}.$$

If we have an element in $\mathcal{F}$ but not $\mathcal{G}$, there's two cases — either the restriction we find is not quasirandom, or the restriction has really small measure. So

$$\mu_p(\mathcal{F} \setminus \mathcal{G}) \leq \mathbb{E}_{x \sim \mu_p^n} 1_{x \in \mathcal{F} \setminus \mathcal{G}} 1_{f_{T \to x_T} \text{ not } (r', \varepsilon')\text{-quasirandom}} + \mathbb{E}_{x \sim \mu_p^n} 1_{x \in \mathcal{F} \setminus \mathcal{G}} 1_{\mu_p(f_{T \to x_T}) < \alpha}.$$

Let's bound each one of these things individually. For the first term, we can ignore the first indicator and just use the fact that the probability we're not quasi-random is very small — it's at most $\varepsilon/2$ by the regularity lemma.

For the second term, we know that the average of $f_{T \to x_T}$ has small measure, so this indicator $1_{x \in \mathcal{F} \setminus \mathcal{G}}$ can't give us too much — we can rewrite this by first taking the expectation over x_T, as

$$\mathbb{E}_{x_T} 1_{\mu_p(f_{T \to x_T})} \cdot \mu_p((\mathcal{F} \setminus \mathcal{G})_{T \to x_T}).$$

But the measure of $\mathcal{F} \setminus \mathcal{G}$ is of course at most the measure of $\mathcal{F}$; and by the indicator we know that this term is less than α, so overall this entire quantity is less than α.

So we get a bound of $\varepsilon/2 + \alpha = \varepsilon$ (we chose $\alpha = \varepsilon/2$). (Technically there's a bound of ε' for the first term, but you can think of ε' as being much smaller than ε.)

Now we're going to finish off the proof by showing that $\mathcal{G}$ is intersecting, which is really the cool point of this proof. Let's suppose $\mathcal{G}$ is *not* intersecting — this means $\mathcal{G}$ contains x and x' that are disjoint. Since $\mathcal{G}$ is a T-junta, we only really care about the coordinates in T; let $z = x_T$ and $z' = x'_T$. We know that x and x' are both in $\mathcal{G}$, so this tells us two things about the restrictions $f_{T \to z}$ and $f_{T \to z'}$: we know $\mu_p(f_{T \to z}) \geq \alpha$ and $f_{T \to z}$ is (r', ε')-quasirandom, and the same is true for $f_{T \to z'}$.

This is looking promising — we're almost in the position of applying the second claim. This wants us to have two families that are large and quasirandom; we already have that. So we just need to translate things from functions to families, and then we can apply it — let $\mathcal{F}' = \{y \in \{0,1\}^{[n] \setminus T} \mid f_{T \to z}(y) = 1\}$ and $\mathcal{G}' = \{y \in \{0,1\}^{[n] \setminus T} \mid f_{T \to z'}(y) = 1\}$ (in other words, we just take the families which are the supports of our functions $f_{T \to z}$ and $f_{T \to z'}$).

Then $\mu_p(\mathcal{F}')$ and $\mu_p(\mathcal{G}')$ are both at least α, adn both are (r', ε')-quasirandom. And of course they're also monotone (this is the point in the proof where the initial assumption that f is monotone comes into play). So they satisfy all the requirements of the second claim, and therefore there are $y \in \mathcal{F}'$ and $\widetilde{y} \in \mathcal{G}'$ that are disjoint. But now we're done — the points $(x_T = z, x_{\overline{T}} = y)$ and $(x_T = z', x_{\overline{T}} = \widetilde{y})$ are two points in $\mathcal{F}$ that are disjoint, which is a contradiction. □

This is a very cool argument, showing how you can combine some sort of sharp threshold machinery to get a result in extremal combinatorics.

> **Remark 12.7.** There are many things here that are quite generic — you define some notion of quasirandomness, show that they behave like random families in terms of sharp thresholds, and then you glue things together to get a result. This method actually has a name — the *junta method* — and you can do a bunch of things with it; this is just one of them.

§12.2 The invariance principle

We'll now move to the next topic, which will take us 2.5 lectures, the *invariance principle*. We stated earlier in the course that the main motivation was actually hardness of approximation in computer science; but we'll take a non-historical route where we first see the principle itself, and later on see some of the applications in computer science.

§12.2.1 Motivation

Let's start with a theorem that we'll all see at some point in our lives, the central limit theorem.

Theorem 12.8 (CLT)
Spupose that $x_1, \dots, x_n \in \{-1, 1\}$ are i.i.d. random variables with $\mathbb{E} x_i = 0$. Then

$$\frac{x_1 + \cdots + x_n}{\sqrt{n}} \sim \mathcal{N}(0, 1).$$

In other words, if we take the sum of our random variables and normalize so that the variance is 1, then the distribution converges to something really nice — the normal distribution.

So far this has nothing to do with analysis of Boolean functions. But now we'll restate the same theorem in somewhat funny notation to motivate what we're going to see.

Define $f \colon \mathbb{R}^n \to \mathbb{R}$ as

$$f(z_1, \dots, z_n) = \frac{z_1 + \cdots + z_n}{\sqrt{n}}.$$

Then the content of the central limit theorem is the same as the following fact:

Theorem 12.9
The following two distributions are 'close' to each other:

(1) $f(x_1, \dots, x_n)$ where $x \in \{-1, 1\}^n$ are uniform and independent.

(2) $f(z_1, \dots, z_n)$ where $z_i \sim \mathcal{N}(0, 1)$ are independent.

The reason these two theorems are morally the same is that a sum of independent Gaussians is also Gaussian (and its variance is the number of summands). So $f(z_1, \dots, z_n) \sim \mathcal{N}(0, 1)$; and therefore these two theorems are really identical. But the second might be a more 'honest' formulation — the first theorem really is this claim, plus the additional miracle that the distribution of $f(z_1, \dots, z_n)$ is nice. You don't expect that applying functions to Gaussians will always give you a Gaussian — for example, even if you multiply two Gaussians the result isn't Gaussian anymore.

Now we'll see what 'invariance' means — it essentially means that the distribution of values of f is invariant under whatever distribution you plug into it. Of course invariance cannot hold for all functions. We're now going to see two types of functions for which it *doesn't* hold; from this we'll learn a little bit, and then we can formulate what the invariance principle actually states.

§12.3 Some counterexamples

Let $f \colon \mathbb{R}^n \to \mathbb{R}$ be a polynomial (we'll only work with polynomials).

Question 12.10. When can invariance hold?

We'll see some examples where invariance *doesn't* hold (this is because if you take a nice polynomial that doesn't have any 'nasty' features, it's going to hold).

Example 12.11

Let $f(z_1, \dots, z_n) = z_1$. Then invariance doesn't hold, since a bit is not close to a Gaussian.

Example 12.12

Let $f(z_1, \dots, z_n) = \prod_{i=1}^{n/2} z_i$. This still doesn't work — if we plug in $z_i \in \{-1, 1\}$ then the result is always ±1, but if we plug in Gaussians then it won't be.

Example 12.13

Let p be any polynomial with $\mathbb{E}p = 0$ and $\mathbb{E}p^2 = 1$ (it's not hard to cook up some polynomial). Let $f(z_1, \dots, z_n) = p(z_1, \dots, z_{n/2}) + \varepsilon \prod_{i=n/2+1}^{n} z_i$. Then invariance still fails.

You might wonder what's the difference between these examples, and why we're looking at them.

We're trying to look at these functions and identify what is the property that makes invariance fail. In the first case, we have a variable 1 that has way too much influence. Of course if a singular variable has large influence, there's going to be a difference between $\{\pm 1\}$ and a Gaussian.

This actually takes care of the second example as well — whenever you have too much influence, invariance isn't going to hold.

Now we get to the third example, where we took a nice polynomial, and then a nasty one times ε. Now we can arrange that the influences are small (if we pick nice p and small ε). But the issue is that if we look at $\prod_{i=n/2+1}^{n} z_i$, in $\{-1, 1\}$ it'll just be ±1. But in Gaussian space it'll be something nasty.

Here's one way to see that — suppose we sample $z \sim \mathcal{N}(0, 1)^n$, and look at the fourth moment of this second term. The z_i's are independent, so this is

$$\mathbb{E}_{z \sim \mathcal{N}(0,1)^n} \prod_{i=n/2+1}^{n} z_i^4 = (\mathbb{E}\mathcal{N}(0, 1)^4)^{n/2} = 3^{n/2}.$$

So in Gaussian space, this second term completely blows up. But that's not about influence, but something else.

So the reason to discard this example is that $\prod_{i=n/2+1}^{n} z_i$ is a high degree polynomial.

§12.4 The invariance principle

So these are the excuses we've came up with from these examples — if you have an influential variable or high degree, you may not be invariant. So now let's look at any other function — any function that has small influences and low degree. Does any such function have to be invariant?

The answer is yes, and this is the invariance principle.

We want to compare two distributions, but one is discrete and the other is continuous, so we need to formalize what it means for them to be close. To do so, we take some test function $\psi \colon \mathbb{R} \to \mathbb{R}$; and the point is that when we apply this test function to f on Booleans or Gaussians, the result is necessarily close.

Theorem 12.14 (Invariance principle, MOO)

Let $f(x_1, \ldots, x_n) = \sum_{|S| \le d} a_S \prod_{i \in S} x_i$ be a multilinear polynomial of degree at most d. Then if $\psi\colon \mathbb{R} \to \mathbb{R}$ is a smooth test function — specifically, $\|\psi'''\|_\infty \le C$ for some constant C — then

$$\left|\mathbb{E}_{x \sim \{-1,1\}^n} \psi(f(x_1, \ldots, x_n)) - \mathbb{E}_{z \sim \mathcal{N}(0,1)^n} \psi(f(z_1, \ldots, z_n))\right| \lesssim C \cdot 2^{3d/2} \cdot \sum_{i=1}^n I_i[f]^{3/2}.$$

Importantly, the sum of influences has a power greater than 1.

First, it's not clear that the right-hand side is small, so let's start by formalizing the intuition that when influences are small, this really is small.

Corollary 12.15

For all $\varepsilon > 0$ and $d \in \mathbb{N}$, there exists $\tau > 0$ such that if $f(x) = \sum_{|S| \le d} a_S \prod_{i \in S} x_i$ has $\max I_i[f] \le \tau$, $\deg f \le d$, and $\|f\|_2 \le 1$, then

$$|\mathbb{E}\psi(f(x)) - \mathbb{E}\psi(f(z))| \le \varepsilon.$$

Proof. By the theorem, the left-hand side of this is at most $C \cdot 2^{3d/2} \cdot \sum_{i=1}^n I_i[f]^{3/2}$. For this sum of influences, we can pull out $I_i[f]^{1/2}$ and be left with

$$\text{LHS} \lesssim C \cdot 2^{3d/2} \max_i \sqrt{I_i[f]} \sum_{i=1}^n I_i[f].$$

And we have $\sum I_i[f] = \sum_{|S| \le d} a_S^2 \, |S| \le d$ (as seen earlier), while we can bound the square-root factor by $\sqrt{\tau}$; this gives

$$\text{LHS} \le C \cdot 2^{3d/2} \cdot d \cdot \sqrt{\tau}.$$

Fially, we can take $\tau = 2^{-6d} \cdot 1/C^2 \cdot \varepsilon^2$, and this should work. □

§12.5 A few remarks

We're not going to prove this theorem today; we are going to formulate the statement that we'll prove next lecture, which will give the main idea of the proof. But first we'll make a few remarks.

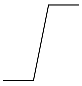
You may wonder what's so special about Gaussians and Boolean bits. The answer is that there is nothing special about them. What's really happenign here is that the average of each coordinate is the same — the average of a bit is 0, and the average of $\mathcal{N}(0,1)$ is also 0. And the variances are also the same — the square of a bit is always 1, and the variance of a Gaussian is also 1. Meanwhile, the third and fourth moments are bounded (but you can see the fourth moments are already not the same).

So what's important is that you have matching 1st and 2nd moments, and bounded 3rd and 4th moments; if you have two distributions that have these properties, then you can get such a result.

§12.5.1 Test functions

What are some good examples of test functions? Morally speaking, polynomials are good test functions (of degree e.g. 1 or 2 or 10). But sometimes, what you really want to do is look at functions that tell you something about the tails or typical values — so you want to consider functions like $\psi(t) = 1_{t \ge 0}$. The only trouble is that this is not smooth. But this is not a big deal — it turns out you can also apply invariance in this case, and we'll also see that.

Extending thse things to non-smooth functions actually has to do with stuff that isn't core to the material in this course, but that we may have seen elsewhere — the idea is that we want to approximate ψ with polynomials. We can certainly approximate ψ by something smooth, like the following:



But then there's tiny intervals with a huge difference between ψ and ψ'. Maybe it's the case that f always lands in these small intervals, and that would make you unhappy. But it turns out that this doesn't happen, due to *anticoncentration* (the only fact that we'll use in this course and not prove):

Theorem 12.16 (Anticoncentration)
If $p\colon \mathbb{R}^n \to \mathbb{R}$ is a degree d polynomial in Gaussian space with $\mathbb{E}p = 0$ and $\mathbb{E}p^2 = 1$, then

$$\mathbb{P}_{z\sim\mathcal{N}(0,1)^n}[p(z) \in (-\varepsilon, \varepsilon)] \le \varepsilon^{\Omega_d(1)}.$$

The message is that Gaussian polynomials can't camp all day around 0 (or around any value); this is another fact that you're going to use if you want to extend to things that are not smooth.

§12.5.2 The Berry–Essen theorem

We'll now write down a statement that we'll prove next time; the proof will give us the ideas to prove the invariance principle. This is a specialization of the statement to linear functions.

Theorem 12.17 (Berry–Essen)
If $f(x_1, \dots, x_n) = \sum_{i=1}^n a_i x_i$, then

$$|\mathbb{E}\psi(f(x)) - \mathbb{E}\psi(f(z))| \lesssim C \cdot \sum_{i=1}^n a_i^3.$$

There's a very nice proof of this, which we'll see next time.

§13 March 21, 2024

Last time, we stated the invariance principle; today our main goal is to prove it.

§13.1 The Berry–Essen theorem

We'll start with the Berry–Essen theorem, which can be thought of as a baby version of the invariance theorem but is very important in its own right.

Theorem 13.1 (Berry–Essen)
Let f be a function of the form $f(x_1, \dots, x_n) = \sum_{i=1}^n a_i x_i$, and let $\psi\colon \mathbb{R} \to \mathbb{R}$ be a smooth test function with $\|\psi'''\|_\infty \le C$. Then

$$\left|\mathbb{E}_{x\sim\{-1,1\}^n}\psi(f(x_1, \dots, x_n)) - \mathbb{E}_{z\sim\mathcal{N}(0,1)^n}\psi(f(z_1, \dots, z_n))\right| \lesssim C\sum_{i=1}^n a_i^3.$$

So for any test function ψ with bounded 3rd derivative, the action of ψ on f when we plug in Boolean variables and independent Gaussians are roughly the same.

There are many proofs; the proof that Dor saw used continuous Fourier analysis. The proof we'll see is an odd proof using a method called the *reflection method*; but it comes very naturally in computer science (where it's called the *hybrid method*).

Proof. Throughout the proof, we'll let $x = (x_1, \ldots, x_n) \in \{0,1\}^n$ and $z = (z_1, \ldots, z_n) \in \mathcal{N}(0, \mathrm{Id}_n)$. The idea is that we'll try to go from x to z slowly — x contains all Booleans and z all Gaussians, and we'll go from one to the other one variable at a time. So we'll define

$$U_t = (x_1, \ldots, x_t, z_{t+1}, \ldots, z_n)$$

for $t = 0, \ldots, n$ (when $t = 0$ it's just all z_i's, and when $t = n$ it's all x_i's). For notational convenience, we'll let $U_{-(t+1)} = (x_1, \ldots, x_t, z_{t+2}, \ldots, z_n)$ denote the above vector, where we remove the $(t+1)$th coordinate.

As mentioned before, we have $U_0 = z$ and $U_n = x$. In this notation, the left-hand side of the theorem is

$$\left|\mathbb{E}_{x,z}\psi(f(U_n)) - \mathbb{E}_{x,z}\psi(f(U_0))\right|.$$

The point is that we have U_0 and U_n and we want to go from one to the other, and we'll do this one step at a time — we can write this as

$$\left|\sum_{t=0}^{n-1} \mathbb{E}_{x,z}\psi(f(U_{t+1})) - \mathbb{E}_{x,z}\psi(f(U_t))\right|$$

(the sum is telescoping), and by the triangle inequality this is at most

$$\sum_{t=0}^{n-1} \left|\mathbb{E}_{x,z}\psi(f(U_{t+1})) - \mathbb{E}_{x,z}\psi(f(U_t))\right|.$$

Now the point is that when we look at f and where we're comparing it, U_{t+1} and U_t only differ in one coordinate; and looking at our error term, it makes sense to think of the a_i's as small. So we're evaluating f at two points which are very close to each other; and let's think about how these things change. We can write

$$f(U_{t+1}) = g(U_{-(t+1)}) + a_{t+1}x_{t+1},$$

and similarly

$$f(U_t) = g(U_{-(t+1)}) + a_{t+1}z_{t+1}$$

(where $g(s_1, \ldots, s_t, s_{t+2}, \ldots, s_n) = \sum_{i \neq t+1} a_i s_i$).

For intuition, we haven't yet done much — we've just written down what f is. But the point is that these two values of f are really the same, up to a bit of shift that only depends on what happens in the $(t+1)$st coordinate; and this is a small shift, because we think of a_i as small. So let's try to see what happens to ψ around $g(U_{-(t+1)})$, and see how this shift tilts it. This is where Taylor comes in — by the Taylor approximation, we can write

$$\psi(f(U_{t+1})) = \psi(g(U_{-(t+1)}) + a_{t+1}x_{t+1}).$$

And now we expand ψ around $g(U_{-(t+1)})$ — we can write this as

$$\psi(f(U_{t+1})) = \psi(g(U_{-(t+1)})) + \psi'(g(U_{-(t+1)})) \cdot a_{t+1}x_{t+1} + \frac{1}{2}\psi''(g(U_{-(t+1)}))a_{t+1}^2x_{t+1}^2 + \frac{1}{6}\psi'''(\zeta)a_{t+1}^3x_{t+1}^3$$

for some $\zeta \in [g(U_{-(t+1)}), g(U_{-(t+1)}) + a_{t+1}x_{t+1}]$ depending on everything else (this last term is our error term — we only do the precise expansion up to order 2, and then fold everything else into an error term). Similarly, we have

$$\psi(f(U_t)) = \psi(g(U_{-(t+1)})) + \psi'(g(U_{-(t+1)})) \cdot a_{t+1}z_{t+1} + \frac{1}{2}\psi''(g(U_{-(t+1)}))a_{t+1}^2z_{t+1}^2 + \frac{1}{6}\psi'''(\zeta')a_{t+1}^3z_{t+1}^3.$$

And now we plug in this Taylor expansion into the differece we're interested in, which we'll call

$$(1) = \left|\mathbb{E}_{x,z}\psi(f(U_{t+1})) - \mathbb{E}_{x,z}\psi(f(U_t))\right|.$$

When we do this, the constant terms are the same, so they'll disappear. The linear terms are kind of the same, except that we have one x_{t+1} and one z_{t+1}; the same is true for the quadratic terms, and then there's the error term, where we can't pull out anything. And we end up with the expectation (over x and z) of

$$\psi'(g(U_{-(t+1)}))a_{t+1}(x_{t+1} - z_{t+1}) + \frac{1}{2}\psi''(g(U_{-(t+1)}))a_{t+1}^2(x_{t+1}^2 - z_{t+1}^2) + \frac{1}{6}\psi''(\zeta)a_{t+1}^3x_{t+1}^3 + \frac{1}{6}\psi'''(\zeta')a_{t+1}^3z_{t+1}^3.$$

But the linear term is 0 — g doesn't depend on the $(t+1)$st coordinate (by how we defined $U_{-(t+1)}$), so $g(U_{-(t+1)})$ and $x_{t+1} - z_{t+1}$ are independent. And $\mathbb{E}x_{t+1} = \mathbb{E}z_{t+1} = 0$, so this term is just 0.

And the quadratic term is 0 for the same reason — $\mathbb{E}x_{t+1}^2 = \mathbb{E}z_{t+1}^2 = 1$, and these things are again independent.

For the remaining error terms, we're not so lucky (we have no control over ζ). So we can't do anything, and we'll just use the triangle inequality; we conclude that

$$(1) \le \frac{1}{6}Ca_{t+1}^3\left|\mathbb{E}\left|x_{t+1}\right|^3 + \mathbb{E}\left|z_{t+1}\right|^3\right|.$$

You can compute exactly what these expectations are — they're some finite numbers — and we get $(1) \lesssim Ca_{t+1}^3$. And that completes the proof. □

Remark 13.2. If you wanted, you could have expanded this Taylor series to one more order (if you had a bound on ψ'''' — the third moments of $\{-1, 1\}$ and $\mathcal{N}(0, 1)$ have the same third moments (both are bounded)). You can see that the more matching moments you have, the further you can get. But that's as far as you can go here, since a Gaussian and bit don't have the same fourth moments.

Remark 13.3. Here we're using the fact (the Lagrange error theorem) that you can for a smooth enough function, you can truncate the Taylor series at a point, at the cost of the last term becoming a derivative at an arbitrary point you don't control.

Remark 13.4. To get CLT, you're interested in the probability of being greater than t. For this, you need ψ to be a non-smooth function; we'll see how to do that today.

Remark 13.5. The error bound of $C\sum a_i^3$ may seem huge at first. But for a typical function we'd have $a_i \approx 1/\sqrt{n}$, because $\sum a_i^2 \le 1$. And then the error bound is $\sum a_i^3 \lesssim 1/\sqrt{n}$, so this is a good bound (it goes to 0 with n).

§13.2 Generalizing to low-degree functions

Now what we're going to do is try to inspect this proof and see how on earth we could extend it in the case where f is not linear, but rather a low-degree function.

There's a natural way to try to generalize this — the main point is, what's the correct analog of writing f as some function depending on all the variables except one, plus one that you expect to be small?

Now that $f(x) = \sum_{|S|\le d} a_S \prod_{i\in S} x_i$, what should be the analog of the statement $f(U_{t+1}) = g(U_{-(t+1)}) + a_{t+1}x_{t+1}$? We split this sum up into characters that do contain $t+1$ and ones that don't — we define

$g(U) = \sum_{t+1 \notin S} a_S \prod_{i \in S} U_i$ (as our analog of g). And all the characters we haven't yet counted all contain $t+1$. So we can write

$$f(x) = g(x) + x_{t+1} \cdot \partial_{t+1} f(x)$$

(note that the term we want here is all the characters containing $t+1$ but with $t+1$ removed, and we have a name for this — it's precisely the partial derivative).

You can now imagine how the proof would continue. The fact that the a_i's were small in the original proof has to e replaced by the fact that $\partial_i f$ is small, which corresponds to the influences being small.

But things don't work exactly as before, because we'll get stuck with derivatives to the third power. And then we'll have to use hypercontractivity to get back to 2. But there's a slight catch — we'll have both Booleans and Gaussians, and we've only proven hypercontractivity for Boolean values. So first we'll show that you can do hypercontractivity for Gaussians, and then we're going to see the argument through.

§13.3 Hypercontractivity in Gaussian space

We're going to talk a little bit about hypercontractivity in Gaussian space. One could spend hours on this, though we won't. It's interesting to note that historically, this is what people were interested in first; to prove it, they first proved it for the Boolean cube, and reduced the Gaussian space case to that one.

Here we're working with functions $f\colon (\mathbb{R}, \mu) \to \mathbb{R}$, where our input space is equipped with the Gaussian measure

$$\mu(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}.$$

We can define an inner product in the same way, as

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f(x) g(x)\, d\mu.$$

Similarly, we can define norms — we define

$$\|f\|_p = \left(\int_{-\infty}^{\infty} |f(x)|^p \, d\mu \right)^{1/p}.$$

Next, we need to find a basis for $(\mathbb{R}, \mu)$. This basis $(h_k(z))_{k=0}^{\infty}$ is called the *Hermite polynomials*. The first few are defined as

$$\begin{aligned} h_0(z) &= 1 \\ h_1(z) &= z \\ h_2(z) &= z^2 - 1 \\ h_3(z) &= z^3 - 3z \\ &\vdots \\ h_k(z) &= (-1)^k e^{z^2/2} \frac{d^k}{dz^k} e^{-z^2/2}. \end{aligned}$$

(Another way to define this basis is to start with $\{1, z, z^2, z^3, \ldots\}$ and apply Gram–Schmidt.)

This formula, despite looking scary, is sometimes very easy to work with. The only property of these that we're going to use is that $(h_k(z))_{k=0}^{\infty}$ is an orthonormal basis for $L_2(\mathbb{R}, \mu)$.

So far, we've dealt with functions in Gaussian space with one variable. But we're working with functions with n variables, so we have to extend all of this; naturally, we can do so by taking a tensor basis. For $L_2(\mathbb{R}^n, \mu)$, we can take the functions

$$h_{k_1, \ldots, k_n}(z_1, \ldots, z_n) = \prod_{i=1}^{n} h_{k_i}(z_i)$$

(this is called *tensorization*; we have some probability space, we take a product of these spaces, and we just tensorize our functions). These functions are again an orthonormal basis.

Today we're only going to care about multilinear functions — which only involve h_0 and h_1 — and so we're going to use words like Parseval (we'll only need the fact that these are orthonormal for $k_i \in \{0, 1\}$).

Finally, we need to define degrees (to talk about hypercontractivity). We define $\deg(h_{k_1,\dots,k_n}) = k_1 + \cdots + k_n$.

Remark 13.6. When we write $d\mu$ in integrals, we mean $\mu(x)\,dx$.

Theorem 13.7

Let $f \colon \mathbb{R}^n \to \mathbb{R}$ be a function of degree at most d. Then for all $q \geq 2$, we have

$$\|f\|_q \leq \sqrt{q-1}^d \, \|f\|_2 \, .$$

Proof. We'll give a very soft argument, where we don't make any calculations — we'll just reduce this to the Boolean case, by approximating the Gaussians by sums of bits. Take N to be large, and choose $x^{(1)}, \dots, x^{(n)} \in \{-1, 1\}^N$. Then we define $F(x^{(1)}, \dots, x^{(n)})$ by plugging in stuff to f — f expects a Gaussian, but we're going to instead shove in something that *looks* like a Gaussian — so we define

$$F(x^{(1)}, \dots, x^{(n)}) = f\left(\frac{1}{\sqrt{N}} \sum_{i=1}^{n} x_i^{(1)}, \dots, \frac{1}{\sqrt{N}} \sum_{i=1}^{N} x_i^{(n)}\right).$$

We can make two observations. First, for all $p \geq 1$, we have

$$\|F\|_p \to \|f\|_p$$

as $N \to \infty$ — this is because f is a reasonable function (it's a polynomial, and therefore smooth), and by the central limit theorem $\frac{1}{\sqrt{N}} \sum_{i=1}^N x_i^{(s)} \to \mathcal{N}(0, 1)$ for each s.

Now let's look at $\|F\|_q$. This is a function on a discrete cube — a *huge* discrete cube (it's a function $\{-1, 1\}^{n \cdot N} \to \mathbb{R}$). But it still has degree at most d (we're plugging in linear forms into some polynomial of degree at most d, so the result is still a polynomial of degree at most d). And so by hypercontractivity we have $\|F\|_q \leq \sqrt{q-1}^d \, \|F\|_2$ (by hypercontractivity for the Boolean cube, as $\deg(F) \leq \deg(f) \leq d$).

And when we send $N \to \infty$ and use the first fact, we conclude the theorem. □

Remark 13.8. There is also a noise formulation of this, which we'll kind of see later on.

Now we have hypercontractivity in bits and Gaussians, but in our proof we'll want functions that have both. But we can establish hypercontractivity for those more general functions in the same way.

Theorem 13.9

The same hypercontractive inequality holds for functions $f \colon \{-1, 1\}^t \times \mathbb{R}^{n-t} \to \mathbb{R}$ that have some Boolean input and some Gaussian input.

We can see this by the exact same proof (just replacing the Gaussian parts by some bits that approximate them).

§13.4 Proof of invariance principle

Finally, now we're ready to prove the invariance principle.

Theorem 13.10

Let $f(x) = \sum_{|S| \leq d} a_S \prod_{i \in S} x_i$ be a multilinear polynomial of degree at most d, and let $\psi \colon \mathbb{R} \to \mathbb{R}$ be smooth with $\|\psi'''\|_\infty \leq C$. Then

$$\left|\mathbb{E}_{x \in \{-1,1\}^n} \psi(f(x)) - \mathbb{E}_{z \sim \mathcal{N}(0, \mathrm{Id}_n)} \psi(f(z))\right| \lesssim C \cdot 2^{3d/2} \sum_{i=1}^n I_i[f]^{3/2}.$$

Proof. We'll use the same notation as before — let $x = (x_1, \ldots, x_n) \in \{-1,1\}^n$ and $z = (z_1, \ldots, z_n) \sim \mathcal{N}(0, \mathrm{Id}_n)$, and let $U_t = (x_1, \ldots, x_t, z_{t+1}, \ldots, z_n)$ and $U_{-(t+1)} = (x_1, \ldots, x_t, z_{t+2}, \ldots, z_n)$ (where we have t x_i's, no $(t+1)$st coordinate, and the rest are z_i's). As before, the left-hand side of the theorem is at most

$$\text{LHS} \leq \sum_{t=0}^{n-1} |\mathbb{E}_{x,z} \psi(f(U_{t+1})) - \mathbb{E}_{x,z} \psi(f(U_t))|$$

(again by telescoping and the triangle inequality).

The proof is going to be very similar to Berry–Essen; we're only going to modify the intuition that when we change one variable, things change only by a little bit in expectation. We can still write f at U_{t+1} and U_t in a similar way — here we get

$$f(U_{t+1}) = g(U_{t+1}) + x_{t+1} \partial_{t+1} f(U_{-(t+1)}),$$
$$f(U_t) = g(U_t) + z_{t+1} \partial_{t+1} f(U_{-(t+1)}).$$

where $g(U) = \sum_{S \not\ni t+1} a_S \prod_{i \in S} U_i$ (so g takes all the monomials that do not contain the coordinate $t+1$). Then we need to include all the monomials that do contain $t+1$; in U_{t+1} this gives an x_{t+1}, and in U_t it gives a z_{t+1}.

(So we've split our terms into monomials that contain $t+1$ and ones that don't; the ones that don't are precisely g, and for the ones that do we pull out x_{t+1} or z_{t+1}, and the rest is precisely a derivative.)

Note that $g(U_{t+1}) = g(U_t)$ (since g doesn't depend on the $t+1$th coordinate, and U_t and U_{t+1} only differ on that coordinate).

Now we push forwards using Taylor — it's really the same thing, but we unfortunately have to make things more complicated-looking. We then end up with

$$\psi(f(U_{t+1})) = \psi(g(U_{t+1}) + x_{t+1} \partial_{t+1} f(U_{-(t+1)})).$$

We expect these derivatives to be small because we think of the influences as small, and then we do Taylor expansion around this point, to get

$$\begin{aligned}\psi(f(U_{t+1})) = \psi(g(U_{t+1})) &+ \psi'(g(U_{t+1})) \cdot x_{t+1} \partial_{t+1} f(U_{-(t+1)}) \\ &+ \frac{1}{2} \psi''(g(U_{t+1})) \cdot x_{t+1}^2 (\partial_{t+1} f(U_{-(t+1)}))^2 + \frac{1}{6} \psi'''(\zeta) x_{t+1}^3 (\partial_{t+1} f(U_{-(t+1)}))^3\end{aligned}$$

for some $\zeta \in (g(U_{t+1}), g(U_{t+1}) + x_{t+1} \partial_{t+1} f(U_{-(t+1)}))$, and the same is true for $\psi(f(U_t))$ with the x_{t+1}'s replaced by z_{t+1}'s — i.e., we get

$$\begin{aligned}\psi(f(U_t)) = \psi(g(U_t)) &+ \psi'(g(U_t)) \cdot z_{t+1} \partial_{t+1} f(U_{-(t+1)}) \\ &+ \frac{1}{2} \psi''(g(U_t)) \cdot z_{t+1}^2 (\partial_{t+1} f(U_{-(t+1)}))^2 + \frac{1}{6} \psi'''(\zeta') z_{t+1}^3 (\partial_{t+1} f(U_{-(t+1)}))^3\end{aligned}$$

As we observed, $g(U_t) = g(U_{t+1})$. So now we're going to take the difference between these two, cancel things as before, and arrive at a similar conclusion. The constant terms are the same, so they go away; and we get an expectation (over x and z) of

$$\begin{aligned}\psi'(g(U_t))\partial_{t+1}f(U_{t+1})(x_{t+1} - z_{t+1})&\\ &+ \frac{1}{2}\psi''(g(U_t))\partial_{t+1}f(U_{-(t+1)})^2(x_{t+1}^2 - z_{t+1}^2)\\ &+ \frac{1}{6}\psi'''(\zeta)\partial_{t+1}f(U_{-(t+1)})^3x_{t+1}^3 + \frac{1}{6}\psi'''(\zeta')\partial_{t+1}f(U_{-(t+1)})^3z_{t+1}^3.\end{aligned}$$

And the same reasoning as before tells you that everything but the error term is just 0. For example, for the first term, $\psi'(g(U_t))$ and $\partial_{t+1}f(U_{-(t+1)})$ don't depend on x_{t+1} and z_{t+1}. And the average of a Gaussian and bit are still the same. The same happens for the second term. So we can again conclude that only the error terms survive. Again using the triangle inequality, we get

$$(1) \leq \frac{1}{6}C \cdot \left(\mathbb{E}\left|\partial_{t+1}f(U_{-(t+1)})\right|^3 (\mathbb{E}\left|x_{t+1}\right|^3 + \mathbb{E}\left|z_{t+1}\right|^3)\right)$$

(here we're using independence to separate out the expectations over x_{t+1} and z_{t+1} — the derivative doesn't depend on the $(t+1)$st coordinate). The expectations of x_{t+1} and z_{t+1} are both some constants, which we don't care about; and this gives us

$$(1) \lesssim C\left\|\partial_{t+1}f\right\|^3_{L^3(\{-1,1\}^t\times\mathbb{R}^{n-t})}$$

(we get a 3-norm over a mixed measure, with some booleans and some Gaussians).

We want to go from L^3 to L^2, so we need to use hypercontractivity with $q = 3$; this gives us

$$(1) \lesssim C\left(\sqrt{2}^d\left\|\partial_{t+1}f\right\|_{L^2(\{-1,1\}^d\times\mathbb{R}^{n-t})}\right).$$

We're still stuck with this mixed measure, but with L^2 we have Parseval — and the first two Hermite polynomials 1 and z are the same as the corresponding Fourier characters, so it doesn't matter which measure we use in this case, and by Parseval we get that this is

$$C \cdot 2^{3d/2} \cdot I_{t+1}[f]^{3/2}$$

(here we're defining influence by the formula $I_{t+1}[f] = \sum_{S\ni t+1} a_S^2$ — the point is that if you take a derivative and measure the 2-norm in either Gaussian or Boolean space, it's still going to be the same thing — this is because the derivative is a multilinear polynomial). □

§13.5 Extensions

So far, we've done things for smooth functions, but oftentimes you might want non-smooth functions — for example, maybe you want a function $\psi_t(s) = 1_{s>t}$. We'd like to prove invariance for this type of function.

It turns out that you still have invariance.

Theorem 13.11

Invariance holds for ψ_t, with slightly worse parameters.

There are two facts one needs for this. One fact, which is a standard calculus thing we may have seen in the past, is that you can get smooth approximations to ψ_t. The second fact is that this is actually good enough — behind that is something nice and deep, which we'll discuss next time.

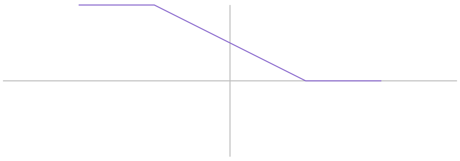
First, getting a smooth approximation for ψ_0 is a standard recipe (if you've seen this in calculus before you know it exists; it's not the best, but it works). You can define the function

$$h(y) = \begin{cases} \alpha e^{-1/(1-y^2)} & \text{if } |y| < 1 \\ 0 & \text{otherwise.} \end{cases}$$

You can check that this is smooth, and has compact support — only on $[-1, 1]$. And then you can choose some constant α such that $\int_{-\infty}^{\infty} h = 1$. Once you have this function, you can use standard convolutions to get a smooth approximation for ψ_0 — take

$$\psi(y) = (1_{(-\infty,0]} * h)(y).$$

Convolution is a type of operator such that if one function is very smooth and nice and the other is not too terrible, then the convolution inherits the nice properties — so ψ is smooth and has bounded derivatives, and you can check that it also has nice support, in that if $y \leq -1$ then $\psi(y) = 1$, and if $y > 1$ then $\psi(y) = 0$.



So below -1 we're 1, and above 1 we're 0. And in between, we're something we don't really care about. This means we kind of got what we wanted, except that this shifting interval should be more narrow (rather than $[-1, 1]$, we need the transition to happen more quickly). And we can get that just by scaling. So shifting ψ and scalign gives ψ_δ such that $\|\psi_\delta'''\|_\infty = O(1/\delta^3)$ and the transition window has length at most δ.

Now we have a smooth approximation and we can try applying the invariance principle on ψ_δ; but it turns out this doesn't quite work. There's some glitch, and to fix it we'll need a nice property of Gaussians.

§14 April 2, 2024

Today we'll pick up where we left off last time; we'll do a bit more regarding the invariance principle, enough so that we can prove the majority is stablest theorem. Then depending on how much time we have left, we'll try to do a brief introduction to complexity theory. At this point in the course, we're moving from more classical topics to some applications of what we've seen in complexity; and then we'll see where we go next.

§14.1 The invariance principle

Last time, we proved the invariance principle, in the following form:

Theorem 14.1 (Invariance principle)

For all $\varepsilon > 0$, $d \in \mathbb{N}$, there exists $\tau > 0$ such that if $f(x_1, \dots, x_n) = \sum_{|S| \leq d} a_S \prod_{i \in S} x_i$ has $\max_i I_i[f] \leq \tau$, then for all smooth $\psi\colon \mathbb{R} \to \mathbb{R}$ with $\|\psi'''\|_\infty \leq C$, we have

$$\left|\mathbb{E}_{x \in \{-1,1\}^n} \psi(f(x)) - \mathbb{E}_{z \sim \mathcal{N}(0, I_n)} \psi(f(z))\right| \leq C\varepsilon \|f\|_2^2 .$$

So if f is low-degree and all influences are small, then for every smooth test function ψ with bounded 3rd derivative, the behavior when we plug in bits and Gaussians is roughly the same. Last time, we proved the concrete bound of $C \cdot 2^{3d/2} \sum_{i=1}^n I_i[f]^{3/2}$.

Often, though, we'll want to work with test functions which are not smooth, and whose third derivative may not be bounded (or even well-defined). In particular, we want to work with the function $\psi_0\colon \mathbb{R} \to \mathbb{R}$ defined as

$$\psi_0(t) = \begin{cases} 1 & \text{if } t \geq 0 \\ 0 & \text{otherwise.} \end{cases}$$

We'd like to prove an invariance principle for this function, and it turns out that we can.

Theorem 14.2

For all ε and d, there exists τ such that for f as above, if $I_i[f] \leq \tau$ for all i, then

$$|\mathbb{E}_x \psi_0(f(x)) - \mathbb{E}_z \psi_0(f(z))| \leq \varepsilon.$$

We're now going to derive this theorem from the previous one. For this, we'll need two components (one of which we saw last time).

The first fact is that ψ_0 has smooth approximations:

Fact 14.3 — For every $\delta > 0$, there exists a function $\psi_\delta\colon \mathbb{R} \to \mathbb{R}$ which is smooth and has the following properties:

(1) $\psi_\delta(t) = 0$ for all $t < 0$.

(2) $\psi_\delta(t) = 1$ for $t \geq \delta$.

(3) $\|\psi_\delta'''\|_\infty \lesssim 1/\delta^3$.

So the first condition says we agree with ψ_0 for all $t < 0$, and the second says we agree with ψ_0 for all t a bit greater than 0. And we jump from 0 to 1 in an interval of length δ, so our third derivative has to be at least $1/\delta^3$; and so the third condition says our third derivative is basically as small as it can be.

The second component has a name — it's called *anticoncentration* in Gaussian space.

§14.1.1 Anticoncentration in Gaussian space

We have an interval where there's a gap between ψ_0 and ψ_δ; we need to consider what happens there. So the first question is:

> **Question 14.4.** What is the probability that a Gaussian lies in that interval (or more generally, in any interval with small length)?

You can show that

$$\mathbb{P}[G \in I] \lesssim |I|$$

for an interval I of length at most δ, just by writing an integral.

We don't just care about a Gaussian though, but about a function of degree d.

> **Question 14.5.** What is $\mathbb{P}[G^d \in I]$?

This is simply (asymptotically at most) $|I|^{1/d}$, since $G^d \in [0, d]$ if and only if $G \in [0, \delta^{2/d}]$.

But what happens if instead of looking at a simple function of degree d, we have a complicated one — maybe a function of *several* Gaussian random variables?

> **Theorem 14.6** (Carbeny–Wright)
> Suppose that $f(z) = \sum_{0<|S|<d} a_S \prod_{i\in S} z_i$ with $\sum a_S^2 \geq 1$. Then
>
> $$\mathbb{P}_{z\sim\mathcal{N}(0,I_n)}[|f(z)| \leq \varepsilon] \lesssim d\varepsilon^{1/d}.$$

So we have some multilinear function of degree at most d, whose 2-norm is not too small (i.e., there should be some variance — if there's no variance then you can't really do anything). This theorem states that then the probability $f(z)$ is in a small interval about 0 is small.

The proof is not that bad — it's a rather old paper which is kind of short — but it is kind of mysterious. There was an alternative proof given later that proved a worse bound, but using tools more in the spirit of what we've seen or will see in this course.

§14.1.2 Invariance principle for ψ_0

Now we'll prove the invariance principle for ψ_0, using these two components.

We want to prove that a certain absolute value is at most ε. We'll prove one direction — that $\mathbb{E}_x\psi_0(f(x)) - \mathbb{E}_z\psi_0(f(z)) \leq \varepsilon$. The other direction is similar. The reason for the asymmetry is that ψ_δ is always a lower bound on ψ_0; for the other direction you'll have to shift it around so that ψ_δ is actually an upper bound on ψ_0, but the idea is the same.

So we want to show that

$$\mathbb{E}\psi_0(f(x)) \leq \mathbb{E}\psi_0(f(z)) + \varepsilon.$$

First we find a function ψ_δ, which we're going to shift around a bit — we take $\widetilde{\psi_\delta}$ to be the shift of ψ_δ such that the jump goes from $-\delta$ to 0. (So we define $\widetilde{\psi_\delta}(t) = \psi_\delta(t+\delta)$ — this makes it an upper bound on ψ_0.) Then $\psi_0 \leq \widetilde{\psi_\delta}$ pointwise, which means

$$\mathbb{E}_x\psi_0(f(x)) \leq \mathbb{E}\widetilde{\psi_\delta}(f(x)).$$

And now we have a smooth function on our hands, so we can apply the invariance principle — the invariance principle tells us that this is at most

$$O\left(\xi \cdot \frac{1}{\delta^3}\right) + \mathbb{E}_z\widetilde{\psi_\delta}(f(z))$$

(here $1/\delta^3$ comes from the third derivative, and $\xi \cdot 1/\delta^3$ is meant to be the small difference coming from the invariance principle). We can see that taking δ small means you win in anticoncentration, but lose in the bound from the invariance principle; so you need to balance these out.

And now that we're in Gaussian space, we can actually move from the approximator to the actual function ψ_0, at some cost — so this is at most

$$O(\zeta/\delta^3) + \mathbb{E}_z\psi_0(f(z)) + \mathbb{P}_z[|f(z)| \leq \delta]$$

(because ψ_0 and our approximator only differ when $f(z)$ is in $[-\delta, 0]$, and they only differ by at most 1). Now we can use our anticoncentration bound. (We can ignore the constant term — of course the constant term doesn't change between $f(x)$ and $f(z)$, so we can ignore it — and if f has small variance, then the principle is trivial.) Plugging in the bound of $O(d\delta^{1/d})$, we get

$$\mathbb{E}_x\psi_0(f(x)) \leq O\left(\xi \cdot \frac{1}{\delta^3} \cdot d\delta^{1/d}\right) + \mathbb{E}_z[\psi_0(f(z))].$$

Finally, we can pick our parameters — we need this to be something like ε, so we pick $\delta = (\varepsilon/d)^d \cdot c$ for some small constant c.

Remark 14.7. Here $\zeta \leq C \cdot 2^{3d/2} \leq \sqrt{\tau}$ is the thing coming from the invariance principle; the only thing to keep in mind is that as $\tau \to 0$ this goes to 0. So the order of quantifiers is we first choose δ so that $d\delta^{1/d}$ is small, and then send take τ sufficiently small so that $\xi \cdot 1/\delta^3 \leq \varepsilon/2$.

Remark 14.8. We should ignore the stuff with the parameters; the important thing is that we saw how to use smooth approximation in a nice way. With Boolean functions we don't have this anticoncentration result — it's actually false. But in Gaussian space we can freely move between different things, as long as they're not too unreasonable.

Remark 14.9. This example shows that even if we have a function which is mostly smooth, we're still fine. And in fact you can do this for any piecewise smooth function — if you can partition ψ into intervals such that it's smooth within each interval, it's still fine (you can get a smooth approximation in the same way and do all this stuff). So what we did above works for *any* piecewise smooth function.

§14.1.3 Another extension

It turns out that this also works for functions that aren't exactly degree d, but are 'almost' degree d.

Theorem 14.10

For all $C, \varepsilon > 0$ and $d \in \mathbb{N}$, there exists τ such that if $f(x_1, \dots, x_n) = \sum_{S \subseteq [n]} a_S \prod_{i \in S} x_i$ and $\psi\colon \mathbb{R} \to \mathbb{R}$ is C-Lipschitz with $\|\psi'''\|_\infty \leq C$, then

$$|\mathbb{E}_x\psi(f(x)) - \mathbb{E}_z\psi(f(z))| \leq \varepsilon + 2C \cdot \|f^{>d}\|_2.$$

So if f is very high-degree then we don't get anything; but if there's very little weight at high degrees then we still get a bound. The way to think about this is that f is mostly degree d, but it has small Fourier weight above degree d. The original theorem wouldn't say anything about f; but this theorem says that as long as our test function ψ is sufficiently nice (i.e., it's Lipschitz), we still get a similar result.

(We're being a bit sloppy with the hypotheses, but always x is Boolean, z is Gaussian, and τ is a bound on the influences.)

Proof. We write $f = f^{\leq d} + f^{>d}$, and because ψ is Lipschitz we can split f inside ψ — the point is that invariance is about low-degree functions, so we'll split f into the low-degree and high-degree parts, and try to replace f with its low-degree parts. By the Lipschitz property, we have

$$\left|\psi(f) - \psi(f^{\leq d})\right| \leq C\left|f - f^{\leq d}\right| = C \cdot \left|f^{>d}\right|.$$

And so on the left-hand side of the theorem, we can replace each f by $f^{\leq d}$, paying this term for each input — so

$$\text{LHS} \leq \left|\mathbb{E}_x \psi(f^{\leq d}(x)) - \mathbb{E}_z \psi(f^{\leq d}(z))\right| + C\mathbb{E}_x \left|f^{>d}(x)\right| + C\mathbb{E}_z \left|f^{>d}\right|(z).$$

The first term is small by standard invariance (we can make it at most ε). Meanwhile, for the second term, by Cauchy–Schwarz we have $\mathbb{E}_x|f^{>d}(x)| \leq \|f^{>d}\|_2$ (and similarly with the second term — here we're using the fact that the 2-norm of a multilinear function is the same in Boolean space as Gaussian space, which we've used several times but swept under the rug). □

§14.2 Majority is stablest

Now with all this machinery, we'll finally talk about the majority is stablest theorem — which was actually the motivation for all of this (it was why the invariance principle was even invented).

§14.2.1 Some setup

We'll start by defining an analogous noise operator to the standard noise operator on the cube, but in Gaussian space.

Definition 14.11. Let $\rho \in [0,1]$, and define $U_\rho\colon L_2(\mathbb{R}^n;\gamma) \to L_2(\mathbb{R}^n;\gamma)$ (as an operator in Gaussian space) as follows: for $f\colon (\mathbb{R}^n,\gamma) \to \mathbb{R}$, we define

$$U_\rho f(x) = \mathbb{E}_{y\sim\mathcal{N}(0,I_n)} f(\rho x + \sqrt{1-\rho^2}y).$$

(We use $L^2(\mathbb{R}^n,\gamma)$ to denote Gaussian space.)

What we're doing is we're taking an average of Gaussians that are ρ-correlated with x. And there's a nice way to do this with Gaussians (by sampling independent y and using the above formula).

Remark 14.12. It's useful to think about what this random variable $\rho x + \sqrt{1-\rho^2}y$ looks like; you can see that marginally it's Gaussian, and its correlation with x is ρ.

You can prove that if f is a multilinear polynomial $f(z) = \sum_S a_S \prod_{i\in S} z_i$, then the action of U_ρ on f is exactly the same as the standard noise operator in Boolean space — we have

$$U_\rho f(z) = \sum_S a_S \rho^{|S|} \prod_{i\in S} z_i.$$

So this is the reason that we're presenting this now — majority is stablest is something about the noise operator on the cube, and if we're reducing it to a Gaussian question then we had better have an analog of that noise operator.

Next, majority is stablest is about stability, so we need to define an analogous notion of stability in Gaussian space.

Definition 14.13. For $f\colon (\mathbb{R}^n,\gamma) \to \mathbb{R}$ and $\rho \in [0,1]$, we define

$$\text{Stab}_\rho(f) = \langle f, U_\rho f\rangle.$$

This is exactly the same as in the Boolean case.

§14.2.2 Proof idea

We want to prove a statement about the Boolean world using invariance. How this works is we take our statement in the Boolean world, and use invariance to reduce it to a statement in Gaussian space. Often that statement is either easier to prove, or already has been proven; and that's what happened here. All this work (in Boolean space) is from 2005; the analog in Gaussian space was proved in the 1870s. It's the following result by Borel, essentially saying that majority is stablest works in Gaussian space.

Theorem 14.14

If $f\colon (\mathbb{R}^n, \gamma) \to [-1, 1]$ has $\mathbb{E}f = 0$, then

$$\mathrm{Stab}_\rho(f) \le 1 - \frac{2}{\pi} \arccos(\rho).$$

This number on the right-hand side is exactly equal to $\mathrm{Stab}_\rho(2 \cdot 1_{z_1 \ge 0} - 1)$ (the stability of the indicator function of a half-space, converted to a ±1-function).

Remark 14.15. This theorem is not easy to prove; but surprisingly there are some values of ρ for which it's just the triangle inequality applied in a clever way. (It's an old theorem, but people sometimes try to reprove them; there's a paper that managed to prove this result for $\rho = \frac{1}{k} \cdot 2\pi$ for $k \in \mathbb{N}$, which is a bit weird.)

Remark 14.16. You can wonder what the Gaussian space buys you. The main thing is that you can symmetrize — Gaussians are symmetric with respect to rotating, while Booleans are not. At a high level this is what the proof does — if f is ±1-valued, you use symmetry to show that all the 1's and -1's are better bunched together, and otherwise you start losing. (You have to do this carefully, though.)

§14.2.3 Proof of theorem

Theorem 14.17 (Majority is stablest)

For all $\varepsilon > 0$ and $\rho \in [0, 1]$, there exists $\tau > 0$ and $d \in \mathbb{N}$ such that if $f\colon \{0, 1\}^n \to \{-1, 1\}$ satisfies $\mathbb{E}f = 0$ and $\max_i I_i^{\le d}[f] \le \tau$, then $\mathrm{Stab}_\rho(f) \le 1 - \frac{2}{\pi} \arccos(\rho) + \varepsilon$.

So this states that if f has zero expectation and small low-degree influences (we didn't discuss low-degree influences in class, but it was on the second problem set), then the stability of f is at most this funny number plus a little bit.

We're going to prove this theorem, which will involve both invariance and being very careful. Applying invariance is often a bit painful; it's very easy to make mistakes.

The first thing which is kind of surprising is that the theorem is talking about *any* function, and all our tools are really about low-degree functions or mostly low-degree functions. So we first have to conceptually deal with that.

The idea here is very simple — take $\delta > 0$ (which is a small parameter that is to be determined — we won't explicitly determine it, but we'll say that you can pick it). And we define $f' = T_{1-\delta} f$ to be f with a *tiny* bit of noise applied to it. Now f' is actually in the ballpark of things we can deal with — because the noise operator hits characters *exponentially* fast, so the weight on tails is going to be very small. So that gets us in the right ballpark. But how do we relate $\mathrm{Stab}_\rho f$ to $\mathrm{Stab}_\rho f'$?

Fact 14.18 — We have $\mathrm{Stab}_\rho(f) \le \mathrm{Stab}_\rho(f') + O_\rho(\delta)$.

Intuitively this makes sense because when you use stability you're already applying noise, which means you're already penalizing the high levels. So if you use a bit more noise, you're penalizing the high levels a tiny bit more, but it's not a serious difference.

So for the rest of the proof, we're really only going to work with f', the slightly noisy version — we'll only bound $\mathrm{Stab}_\rho(f')$.

Now we have a function which is kind of like a low-degree function, which is good. But now we need to apply the invariance principle. There are extensions of it (e.g., a multivariate invariance principle where you have several correlated inputs, which we may see in a problem set in the future) which you can also apply here, but here we'll try to stick to the basics.

Note that $\mathrm{Stab}_\rho(f') = \langle f', T_\rho f'\rangle$. We want to turn this from an inner product to something that only has one input; there's a trick where we can split T_ρ into two noise operators $T_{\sqrt{\rho}}T_{\sqrt{\rho}}$, and now because T is a symmetric operator we can move one to the other side and actually get a norm — so we get

$$\mathrm{Stab}_\rho(f') = \langle f', T_\rho f'\rangle = \langle f', T_{\sqrt{\rho}}T_{\sqrt{\rho}} f'\rangle = \left\| T_{\sqrt{\rho}} f'\right\|_2^2.$$

And now we make another move which right now looks trivial but we'll see the reason fro — we can write this as

$$\mathbb{E}_x \mathrm{Square}(T_{\sqrt{rho}} f'(x)).$$

The point is that we're going to move to Gaussian space. If we have a function f which is ± 1-valued, then f' is bounded (it's always in $[-1,1]$). It's also a multilinear polynomial. We know it's bounded when we plug in Boolean inputs, but when we plug in Gaussian inputs it may be completely crazy. So we'll take care of that here — we define

$$\mathrm{Square}(t) = \begin{cases} t^2 & \text{if } t \in [-1,1] \\ 1 & \text{else} \end{cases}$$

(this is just a truncated square function, because we don't want things to get completely crazy in Gaussian space).

And Square is piecewise-smooth and Lipschitz, so we can use invariance to say that we can plug in Gaussian random variables instead — so this is at most

$$\mathbb{E}_z \mathrm{Square}(T_{\sqrt{\rho}} f'(z)) + \frac{\varepsilon}{2} + O(\left\|(T_{\sqrt{\rho}} f')^{>d}\right\|_2).$$

Remark 14.19. To clarify what we mean by this, f' is a multilinear function, so when we talk about $T_{\sqrt{\rho}} f'(z)$, this means we look at f' as a multilinear function and then plug in Gaussians (otherwise this would make no sense).

The last term $(T_{\sqrt{\rho}} f')^{>d}$ is at most $(1-\delta)^d$, since f' is f with a bit of noise to make the high-degree things exponentially vanishing. So this is small, and we can kind of ignore it.

Why did we do this business with the function Square, rather than writing $\mathbb{E}_z(T_{\sqrt{\rho}} f'(z))^2 = \langle f', U_\rho f'\rangle$ and applying Borel? The point is that Borel requires the function to be bounded — f' is bounded as a Boolean function but not as a Gaussian one, and so we can't apply Borel in this way directly. So all these shenanigans are to get somewhere where we can apply Borel.

We need to make $T_{\sqrt{\rho}} f'(z)$ into something bounded (right now it isn't), and to do that we need some more force. We define the function trunc: $\mathbb{R} \to \mathbb{R}$ to be

$$\mathsf{trunc}(t) = \begin{cases} -1 & t \le -1 \\ t & -1 \le t \le 1 \\ 1 & t \ge 1 \end{cases}$$

(so to make our function bounded, we just apply truncation), and we define

$$F(z) = \mathsf{trunc}(f'(z))$$

(the thing we really want is f', but that's not bounded, so we make it bounded). Now we claim that F and f' are close to each other. THis is where the shenanigans with invariance mentioned earlier enter the picture. It's not clear in what sense they're close, but we claim that $\mathbb{E}_z \left|F(z) - f'(z)\right|$ is small — the point is we can write this as

$$\mathbb{E}_z \text{dist}(f'(z), [-1, 1]).$$

And we know literally nothing about f' in Gaussian space, but we do know something about f' in the Boolean world — it is the noise of a Boolean function, so it is in particular bounded. So if instead of z we had x, this distance would always be 0. And so what we do — which is kind of cute — is that we move *back* to Boolean space, again using invariance. This is at most

$$\mathbb{E}_z \text{dist}(f'(x), [-1, 1]) + O(\left\|(f')^{>d}\right\|_2) + \varepsilon/4$$

(the $\varepsilon/4$ is from the influences, and the second term is because of the tail; this distance function is a Lipschitz function). And as before, $\left\|(f')^{>d}\right\|_2$ is small (at most $(1-\delta)^d$). And importantly, the first term is identically 0 (this was the point). And so we get

$$\mathbb{E}\left|F(z) - f'(z)\right| \leq \frac{\varepsilon}{4} + (1-\delta)^d.$$

So f' is close to F, and Square is a reasonable function (it's also bounded, so there's never contributions of a gazillion — it's always at most 1). So we can bound $\text{Square}(T_{\sqrt{\rho}} f'(z))$ by the same thing with F, plus something depending on the distance — by the above thing, this is at most

$$\mathbb{E}_z \text{Square}(T_{\sqrt{\rho}} F(z)) + \frac{3\varepsilon}{4} + 2(1-\delta)^d.$$

(Here we're using the fact that Square is Lipschitz.)

And now we're kind of done, almost — now we can rewrite Square in a normal way, because the thing inside it is always between -1 and 1. So we can rewrite this as

$$\mathbb{E}_z (T_{\sqrt{\rho}} F(z))^2 + \frac{3\varepsilon}{4} + 2(1-\delta)^d.$$

(Strictly speaking, we had to move from T to U at some point — when we converged from $\mathbb{E}_x$ to $\mathbb{E}_z$.)

And now we're in a position to apply Borel — as before, this is $\langle F, U_\rho F\rangle + 3\varepsilon/4 + 2(1-\delta)^d$. And now we can apply Borel, so this is at most

$$1 - \frac{2}{\rho}\arccos(\rho) + \frac{3\varepsilon}{4} + 2(1-\delta)^d.$$

Remark 14.20. Actually we're lying slightly, because F is not balanced — $\mathbb{E}[F]$ is not really 0. But $\mathbb{E}f' = 0$ and F and f' are close, so it's not too far off — you can show $\left|\mathbb{E}F\right| \leq 3\varepsilon/4 + \cdots$. So you can shift F around a bit and renormalize, and then apply Borel; we're not going to do the details.

So now you first pick δ to be small enough so that the original error is at most $\varepsilon/10$, then pick d large enough such that $(1-\delta)^d$ is at most $\varepsilon/10$, and then you're done.

> **Remark 14.21.** In terms of ideas, there's really 1.5 ideas — 0.5 is how to write things correctly so that you can actually get invariance into play. And the more nuanced idea is that you have to move to these truncated things to make sure that nothing is close to ∞; and the core is that when you move to Gaussians, even though you're no longer bounded, you're still 'morally bounded' because you can apply invariance back (and for Booleans you really are bounded). But of course making all of this into a proof is quite messy.

Next time we'll do a brief introduction to complexity theory — we'll discuss some optimization problems and say that some things are easy and some are hard, and then we're going to turn to approximation algorithms and hardness of approximation.

§15 April 4, 2024

Today we'll give a quick crash course on complexity theory, with the goal of motivating hardness of approximation. (There won't be any Boolean functions today.)

§15.1 Computational problems

We've hopefully all seen some computational problems in our life; we'll sketch some important ones.

Definition 15.1 (Reachability)

INPUT: a graph $G = (V, E)$, and two vertices $s, t \in V$.

TASK: is there a path from s to t in G?

This is probably the first problem you'll see in any algorithms class. You can solve it by BFS (breadth-first search) or DFS (depth-first search). This is not an algorithms class, but we'll describe these anyways. We start with s, and look at all the edges adjacent to it — suppose we reach v_1, v_2, $\dots$. We'll maintain a set of all vertices we've reached so far; at each step we look at some vertex we haven't explored yet (in the set), look at all its neighbors, and add them to our set.

So in other words, we maintain a list of vertices reachable from s; as long as there's some vertex we haven't exhausted, we keep on doing this until we're done. And in the end, we check whether we've reached t.

In an algorithms class, you squint a bit and look at the runtime (which is $O(|V| + |E|)$ or something). We won't care about this; we just care it's polynomial time.

Another problem, which is maybe more important for our purposes:

Definition 15.2 (Matrix multiplication)

INPUT: two matrices $A, B \in \mathbb{F}_q^{n\times n}$.

TASK: compute $A \cdot B$.

We can do this simply by writing down the formula $(A \cdot B)_{ij} = \sum_{k=1}^{n} A_{ik}B_{kj}$. (This takes $O(n^3)$ time — there are n^2 entries, and each takes time $O(n)$ to compute.)

If you ask a random undergrad, they'd probably say this is the best you can do. But it's not — it is suspected that $O(n^{2+o(1)})$ is possible. This is wide open, but there are certainly algorithms better than $O(n^3)$.

Matrix multiplication is one of the most basic tasks in linear algebra, but once you know how to do this, you can do basically anything else with a related cost — matrix inversion, solving systems of linear equations, and so on.

Definition 15.3 (Bipartiteness)

Input: a graph $G = (V, E)$.

Task: determine whether G is bipartite — is there a partition $V = A \cup B$ such that there are no edges inside A or B?

You can solve this by doing BFS, coloring even levels red and odd levels blue, and checking if this is a 2-coloring.

When you first see this, you might think there's many partitions, and checking each would take a long time. But we don't have to do this — we can do this again by exploration, where we start at v and put it in A, then put its neighbors in B, then their neighbors in A, and so on.

So far, all these problems are easy. Now we'll get to a different set of problems. Let's start with a problem that looks related to bipartiteness.

Definition 15.4 (Max-Cut)

Input: a graph $G = (V, E)$.

Task: find a partition $V = A \cup B$ maximizing the number of edges between A and B.

In bipartiteness we're trying to capture *all* the edges by our partition; here we're only trying to get as many as we can.

This is a problem we believe is not polynomial time solvable; it's NP-hard (we'll soon explain what that means, but for all intents and purposes it means there's probably no polynomial time algorithm for it).

The next problem is sort of the mother of all NP-hardness results.

Definition 15.5 (3SAT)

Input: a Boolean 3CNF formula

$$\varphi(x_1, \ldots, x_n) = C_1 \wedge C_2 \wedge \cdots \wedge C_m,$$

where each clause C_i is of the form e.g., $C_i = (x_1 \vee \overline{x_{17}} \vee x_5)$ (an OR of three literals, each of which is either a variable or its negation).

Task: is there a Boolean assignment to the variables $A\colon \{x_1, \ldots, x_n\} \to \{0, 1\}$ satisfying φ (i.e., satisfying every clause C_i)?

This is also a very difficult problem as far as we know; it's also NP-hard.

The last problem we'll mention is the vertex cover problem.

Definition 15.6. Given a graph $G = (V, E)$, a *vertex cover* is a subset $C \subseteq V$ that touches every edge — i.e., such that for every edge $uv \in E$, at least one of u and v is in C.

Clearly every graph has a trivial vertex cover (just take all the vertices). So naturally, what you want to do is, given a graph, find the *smallest* possible vertex cover.

Definition 15.7 (Vertex-Cover)

Input: a graph $G = (V, E)$.

Task: find the smallest vertex cover in G.

This is also NP-hard.

So we see a pretty sharp contrast — the first set of problems are all quite easy (we can solve them in polynomial time), but for the second set, people have thought about them for 50 or more years and no one has come close to solving them.

§15.2 NP-hardness

We've repeatedly mentioned NP-hardness. We're not going to fully define it (that'd require us to talk about Turing machines and NP and so on). But we'll talk about what it sort of means.

In computational complexity, people really like to define classes. There are many classes; most are bizarre, but some are important; as far as Dor is concerned, the two most important ones are P and NP. Here P stands for 'polynomial' — it is the class of problems that can be solved in polynomial time. In particular, all the problems we saw in the first set (e.g., reachability) belong to P. Note that P isn't a boring class by any means — it actually contains lots of nontrivial problems.

Example 15.8

Given an integer $n \in \mathbb{N}$, is it a prime?

This is a fascinating problem. If you've taken a course on randomized algorithms, you've probably seen a randomized algorithm for this problem based on group theory. That was all we had for a long time, but twenty years ago three people came up with a proper polynomial time algorithm — which is nontrivial and quite ingenious. There are many other problems similar to this .

Then there's NP, which does *not* stand for 'not polynomial.' It stands for 'nondeterministic polynomial.' We won't get into nondeterministic complexity; but there are many problems here that we really would *like* to solve in polynomial time.

The class NP is sort of the class of problems where if I give you some *witness*, then you could solve it in polynomial time. For example, 3SAT is in NP because if I told you what the assignment is, you could of course solve it (you could just check that it satisfies all the clauses).

And now we get to the word NP-hardness.

Computational complexity is a great field, but we really don't know how to show that stuff are not solvable. (We're good at coming up with algorithms, but showing there's *no* algorithm for something is something we don't really know how to do.) Still, we want evidence that some things are hard. So we do the next-best thing — we assume that *one* problem is hard, and based on this we prove that a whole bunch of other problems are also hard.

Here's an example statement along these lines:

Theorem 15.9

If there exists an efficient (i.e., polynomial-time) algorithm for MaxCut, then there exists an efficient algorithm for the entire class NP (and in particular, for 3SAT).

What this says is that if we're able to solve this one problem MaxCut in polynomial time, then you're actually able to solve this entire huge class of problems. So in a sense, this means MaxCut is the 'hardest' problem in NP. This is what we mean by NP-hardness — informally, 'NP-hard' means 'as hard as any problem in NP.' This is the gold standard of hardness in complexity theory (and theoretical computer science in general) — today if you show a problem is NP-hard, it's as good as there not being a polynomial time algorithm.

This is an example theorem; and actually this theorem holds if you replace MaxCut with any of the three problems in our second set (if there's an efficient algorithm for Vertex-Cover or 3SAT, then you can also solve NP).

§15.3 Coping with NP-hardness

There are lots of NP-hard problems in the world, and we know as theoreticians that they're hard to solve; but there are practitioners in the world that still want to solve them. So you have to settle for something; there's several ways you can go about it.

For one thing, NP-hardness refers to *worst*-case analysis. What this means is that when we say Max-Cut is hard, we mean that if we look at Max-Cut over *all* possible graphs, it'll be NP-hard. But maybe the graphs we care about have some special properties (e.g., maybe they're planar). So you can try to solve sub-classes of instances that have additional features. For example, maybe for graph problems you care about planar graphs or bounded-degree graphs or random graphs; there's a lot of study about these things, but we won't discuss it.

Another way to deal with NP-hardness is *approximation*; and this is what we'll discuss.

We'll talk about approximation with respect to the problems discussed so far. When you talk about approximation, you need to talk about *promise* problems instead of decision problems. With a decision problem, I give you an instance, and you need to figure out e.g. whether there's a satisfying assignment or not. With a promise problem, I give you an input and *promise* you that there is a satisfying assignment, and your goal is to find an assignment that satisfies *a lot* of the clauses.

Definition 15.10 (Approximation 3SAT)

Input: a 3CNF formula φ *promised* to be satisfiable.

Goal: find an assignment satisfying as many of the clauses as possible.

In the decision problem we want to find an assignment satisfying *all* the clauses; but here we relax that, and we just want to satisfy as many as we can.

The way to think about this is that if I give you an assignment satisfying 99% of the clauses, that's pretty good; maybe we can ignore the rest and call it a day. SO that's what it means to have an approximation algorithm for 3SAT.

Claim 15.11 — There exists an efficient algorithm that, given a 3CNF formula φ with m clauses, finds an assignment satisfying at least $\frac{7}{8}m$ clauses.

In other words, there is a 7/8-approximation algorithm for 3Sat.

Proof. Let's consider a clause, e.g. $C_1 = x_1 \vee \overline{x_7} \vee x_{15}$. If we chose the assignment of x_i's at random, then what is $\mathbb{P}[C_1 = 1]$? For C_1 to evaluate to 0, we need $x_1 = 0$, $x_7 = 1$, and $x_{15} = 0$; each of these is independent and happens with probability $\frac{1}{2}$. So if we choose our assignment randomly, $\mathbb{P}_x[C_i = 0] = 1/8$; this means $\mathbb{E}_x[\#\text{satisfied } C_i] = 7m/8$. So if we choose an assignment randomly, in expectation we get what we want.

Once we have this, we can do standard things to get a *randomized* algorithm. In fact, if you work a bit harder, you can get a deterministic algorithm (if you've heard teh notion of 'pairwise independent hash functions' — or in this case three-wise independent — then you can do something better than selecting everything independently at random). □

So we have 3SAT, and we know satisfying all the clauses is NP-hard, and we have a claim giving us a 7/8-approximation. So then the question is, is there a better approximation algorithm?

We'll put this aside and consider a different problem, approximation for vertex-cover.

Claim 15.12 — There exists an efficient algorithm that, given a graph $G = (V, E)$ with $\mathsf{VC}(G) = k$ (where VC denotes the size of the minimum vertex cover), finds a vertex cover of size at most $2k$.

Proof. We'll use a greedy algorithm — start with $C = \emptyset$. And then at each point, we pick some edge $e = uv \in E$, add both vertices u and v to C, and remove from G every edge adjacent to either u or v.

It's easy to see this runs in polynomial time. To see that it works, suppose that throughout the algorithm we hit edges $e_1, \ldots, e_\ell$. Then first we claim that $k \geq \ell$ — this is because these edges are vertex-disjoint, and a vertex cover has to pick at least one vertex from each edge.

But the total number of vertices that our algorithm took is 2ℓ, so we have $|C| = 2\ell \leq 2k$, and we're done. □

Next we'll look at approximating max-cut.

Claim 15.13 — There exists an efficient algorithm that, given a graph G with $\mathsf{MC}(G) = k$, finds a cut of size at least $\frac{1}{2}k$.

Proof. You can choose a partition $V = A \cup B$ randomly — by including each vertex in A with probability 1/2. Then for each edge uv, the probability that u and v end up on different sides is 1/2; so by linearity of expectation, the expected number of edges crossing the cut is

$$\mathbb{E}_{A,B}\mathsf{cut}(A, B) = \frac{1}{2}|E| \geq \frac{1}{2}k.$$ □

So each of these NP-hard problems has an approximation algorithm. This was the state of affairs in the 1980s, and it's a kind of unclear situation — we have a trivial algorithm that achieves some approximation ratio. It's not clear how to do better, but these algorithms are super simple (e.g., this algorithm doesn't even look at the graph, and the one fot 3SAT doesn't even look at the formula). So you might expect we can do better.

§15.4 The PCP theorem

But now the plot thickens. In the 1990s, the PCP theorem was proved, which is an amazing result in computational complexity. It has many formulations; we'll see one that's tailored to our purposes.

Theorem 15.14 (3SAT formulation of PCP)

There exists a constant $\varepsilon > 0$ such that given a 3CNF formula promised to be satisfiable, it is NP-hard to find an assignment satisfying at least a $(1 - \varepsilon)$ fraction of the clauses.

What this theorem says is that not only is it hard to find an actual satisfying assignment, but in fact it's hard to find even something that satisfies 99% of clauses (which is a significantly easier task, but still turns out to be NP-hard).

So now there's something interesting — we know that 7/8-approximation is possible, but there is some bizarre number $1 - \varepsilon$ where approximation is not possible.

Question 15.15. Where is the threshold for when 3SAT approximation becomes NP-hard?

We'll introduce some convenient notation.

Definition 15.16 (gap-3SAT$[1, s]$)

INPUT: a 3CNF formula φ promised either to be fully satisfiable, or at most s-satisfiable (i.e., such that we can't satisfy more than a s-fraction of clauses).

GOAL: which one is it?

Then the above theorem can equivalently be restated as there existing $\varepsilon > 0$ such that gap-3SAT$[1, 1-\varepsilon]$ is NP-hard.

§15.5 Implications

This is what happened in the beginning of the 1990s, and people tried pushing this forwards — what's the best constant you can get? Can you get results for the other problem?

This is much of where Fourier analysis entered the picture; we'll discuss this in later lectures. A lot of machinery has been developed, and it gave a very nice set of results.

Theorem 15.17 (Hastad)

For all $\varepsilon > 0$, gap-3SAT$[1, \frac{7}{8} + \varepsilon]$ is NP-hard.

This is amazing — it tells you that the trivial algorithm is the best you can do (the best approximation algorithm doesn't even look at the formula, which is kind of bizarre).

Theorem 15.18 (2000)

MaxCut is hard to approximate within a factor of $\frac{16}{17}$.

With 3SAT, the theorem matches the algorithm. But here it doesn't, so there's the question — is the right constant $\frac{1}{2}$ or $\frac{16}{17}$ (or in between)?

For vertex-cover the situation is even more peculiar.

Theorem 15.19 (Dinur–Safra 2002)

Vertex-Cover is NP-hard to approximate within a factor of ≈ 1.36.

(The actual number is a funny number with square roots.)

(Prior to them, there was another result with $7/6$, but we're not discussing all the history.)

Now the situation is even more interesting — for 3SAT we have everything we want, for MaxCut we have some bizarre looking number and it's not clear to do, and with VertexCover we again have a bizarre looking number.

§15.6 The unique games conjecture

What happened for Max-Cut and Vertex-Cover is — this has not been resolved up to today. If you only care about NP-hardness, the theorem for max-cut is the best we know today, and for vertex-cover the best we know (from 2018) is improving 1.36 to $\sqrt{2} - o(1)$.

Even proving NP-hardness results is hard, if you only rely on NP-hardness. One person said, maybe we can strengthen our assumption — we started with the assumption $\mathsf{P} \neq \mathsf{NP}$, but maybe we can strengthen it to let us conquer all the approximation problems.

The unique games conjecture is by Khot 2002 — it's a very interesting conjecture that seemed promising in order to tighten these gaps.

To state this conjecture, we need to define a computational problem, which is a bit of a mouthful.

Definition 15.20 (Unique Games)

Input: a bipartite graph $G = (L \cup R, E)$, an alphabet Σ, and a collection of constraints, with one for each edge — i.e., we have a constraint map $\varphi_e \colon \Sigma \to \Sigma$ for each $e \in E$, such that each is a permutation.

Goal: find assignments $A \colon L \to \Sigma$ and $B \colon R \to \Sigma$ satisfying as many of the constraint maps as possible — we want to maximize

$$\#\{e = (u, v) \mid A(u) = \varphi_e(B(v))\}.$$

So we have a bipartite graph and some edges, and for each edge we attach a constraint $\varphi_{e_1}, \varphi_{e_2}, \ldots$. These constraint maps are simply maps from the alphabet to itself; the word 'unique' refers to the fact that they are one-to-one (i.e., permutations).

This is a bit of a mouthful and a bit abstract. We'll now give a natural example (which actually is quite general).

Example 15.21 (2LIN)

Suppose we have two sets of variables $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_{n'}\}$ and a finite field $\mathbb{F}_q$, and a collection of equations each of the form $x_i - y_j = b_{ij}$ (where all arithmetic is done over $\mathbb{F}_q$).

We claim this problem is a specific example of a unique game. You can set X to be one side of the partition and Y the other, with one edge representing each equation — so $L = X$, $R = Y$, $\Sigma = \mathbb{F}_q$, and the edges in E are all (i, j) participating in an equation; for each, we define $\varphi_e(y) = y + b_{ij}$ (because moving y to the other side, what we're requiring is that $x_i = y_j + b_{ij}$).

We're encouraged to think about 2LIN, which is a lot more concrete and easy to get our hands on but is actually quite general; but for technical reasons we work with the more general unique games problem.

Conjecture 15.22 (Unique games conjecture, Khot 2002) — For all $\varepsilon, \delta > 0$, there exists $k \in \mathbb{N}$ such that given an instance $\psi = (G, \Sigma, \Phi)$ (where Φ is the collection of constraints) of Unique Games with $|\Sigma| \leq k$, it is NP-hard to distinguish between:

- The Yes case — it is possible to satisfy at least an $1 - \varepsilon$ fraction of the constraints.
- The No case — it is not possible to satisfy even a δ fraction of the constraints.

So this is the unique games conjecture. It is still open; if you solve it you will become famous and all that. But this was made in 2002, and if you solved it in 2002 nothing would happen (no one would care).

The reason peopel became interested in it is it turns out that if you assume this conjecture, you can actually nail down the best approximation algorithm for MaxCut, VertexCover, and lots of other problems.

In the next two lectures, we'll see the case of MaxCut — it turns out that neither the 16/17 nor 1/2 number is true. You'll need a smarter algorithm (the Goemans–Williamson algorithm, which is a very beautiful algorithm — we've seen some of the ideas already), and then a NP-hardness reduction showing that's the best you can do (except a reduction from this conjecture instead).

§16 April 9, 2024

Today the lecture will revolve around an optimization problem discussed last time called Max-Cut — we'll quickly recall what it is, and then see an algorithm for it and a matching hardness result (which will rely on the unique games business).

Definition 16.1 (Max-Cut)

INPUT: a graph $G = (V, E)$.

GOAL: find a bipartition of V into two sets $V_1 \cup V_2$ which maximizes the number of edges crossing the cut, i.e.,

$$\#\{e = (u, v) \in E \mid u \in V_1,\, v \in V_2 \text{ or } u \in V_2, v \in V_1\}.$$

We already know that this is NP-hard to solve exactly, so we can ask what we can do with an approximation algorithm. Last time we saw a very easy $\frac{1}{2}$-approximation — just randomly choose the partition, and we'll expect to get half of the *edges*, so of course we'll get half the maximum cut.

This was the best algorithm known up until the mid-1990s. And then a surprise came — it turned out this algorithm is not the best you can do.

§16.1 Goemans–Williamson algorithm

§16.1.1 Max-Cut as an integer program

We'll start by formulating Max-Cut as an integer programming problem.

For every vertex $v \in V$, introduce a variable $x_v \in \{+1, -1\}$. The intention is that all the variables that get 1 are on one side, and all the variables that get -1 are on the other side. With this in mind, we're trying to find

$$\max \frac{1}{2} \sum_{e=(u,v)} (1 - x_u x_v)$$

(since $1 - x_u x_v$ gives us a 0 if u and v are on the same side of the cut, and 2 if they're on different sizes), subject to the constraints that $x_v \in \{-1, 1\}$.

This problem is completely equivalent to Max-Cut, so we haven't made any progress — it's still NP-hard. But the reason we did this is that there's a very useful idea — once you write an integer program that's NP-hard, you can try to *relax* it to something that's not NP-hard, and won't give you the exact solution but should give something close. One example is *linear programming* — instead of insisting on Booleans, we can insist on having values in $[-1, 1]$.

The relaxation we'll do here is different, because here we have some quadratic-looking thing.

§16.1.2 The semi-definite program relaxation

Amusingly, you can think of $\{-1, 1\}$ as the unit ball of dimension 1 (or 0). The semi-definite programming relaxation is that we instead use a unit ball of *higher* dimension.

So now we're choosing $x_u \in \mathbb{R}^k$ and we're trying to find

$$\max \frac{1}{2} \sum_{e=(u,v)} 1 - \langle x_u, x_v \rangle,$$

subject to $\|x_u\|_2^2 = 1$ for all $u \in V$.

We'll call the original program $\mathcal{P}_1$, and the new one $\mathcal{P}_2$.

There's several things we need to observe. First, this is a *relaxation* of $\mathcal{P}_1$ — in particular $\text{Val}(\mathcal{P}_2) \geq \text{Val}(\mathcal{P}_1)$, since every actual ± 1 solution can be made into a vector (by just padding with 0's). So any solution to $\mathcal{P}_1$ can be converted into a solution to $\mathcal{P}_2$.

So at least we're not *decreasing* the value of our problem.

The first time you see semidefinite programs, it's not clear why this is solvable and the original is not; but it turns out that this is the case. If this were an algorithms course, we could spend two lectures on why this is true; but really the reason this is solvable is that here optimization is really occurring over the Gram matrix of our vectors, i.e., the matrix $A = (\langle x_u, x_v \rangle)_{u,v\in V}$. But the set of these matrices has a name — these are *positive semidefinite matrices.* And one feature of the collection of PSD matrices is that it's convex; so this problem falls into the realm of convex optimization, which makes it solvable.

§16.1.3 Generating a cut

So we've started with an integer program and relaxed it into a semidefinite program; and we've said that this is something we can solve. As notation, let $\text{MC}(G) = \rho\,|E|$. Then when we solve $\mathcal{P}_2$, we get a collection of unit vectors $\{x_v\}_{v\in V} \subseteq \mathbb{R}^k$ such that the objective function $\frac{1}{2}\sum_{(u,v)\in E}(1 - \langle x_u, x_v\rangle) \geq \rho\,|E|$.

> **Remark 16.2.** The point is that you'll get some Gram matrix, and then you need to run some algorithm to turn it into a collection of vectors; the k you'll end up with should be polynomial in n. (We don't choose k; you can guarantee it'll be $\mathsf{poly}(n)$, but it's not up to us.) (If you insist and you pay a little bit, you can actually ensure that $k = 3$ in some cases, using some dimension-reduction argument. But this doesn't really help.)

Now we want to analyze this. For intuition, think of ρ as close to 1 (e.g., $\rho = 1 - \varepsilon$ where ε is small). If we divide this equation by $|E|$, we get that

$$\frac{1}{|E|}\sum_{(u,v)\in E}\frac{1}{2}(1 - \langle x_u, x_v\rangle) \geq \rho.$$

So on average $\frac{1}{2}(1 - \langle x_u, x_v\rangle) \geq \rho$; and this number is always at most 1. So this means for a 'typical' edge we have

$$\frac{1}{2}(1 - \langle x_u, x_v\rangle) \geq \rho = 1 - \varepsilon,$$

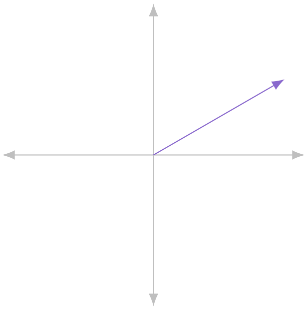
which rearranges to

$$\langle x_u, x_v\rangle \leq -1 + 2\varepsilon.$$

So x_u and x_v are unit vectors, and the inner product of these two vectors is nearly as small as it can be (it's nearly -1). So geometrically, this means that they're nearly opposite each other. And that's the geometric interpretation.

Now the question is, how do you use these vectors — how can we generate a cut using them? This is a nontrivial question.

Pictorially, we have a collection of vectors where for a typical edge, these vectors point in nearly completely different directions. We want to assign ± 1-values to vertices so that edges are typically assigned -1; so let's pick a random hyperplane and assign signs based on which side the corresponding vector is on.



So we sample a unit vector $h \in \mathbb{R}^k$ uniformly, and define $L = \{v \mid \langle x_v, h\rangle < 0\}$ and $R = \{v \mid \langle x_v, h\rangle \geq 0\}$.

As a calculation, fix $(u, v) \in E$; what is the probability that $e = (u, v)$ crosses the cut? We have

$$\mathbb{P}[e = (u, v) \text{ crosses } (L, R)] = \frac{\theta(x_u, x_v)}{\pi}$$

(if x_u and x_v are opposite then the probability is 1). Angles don't appear in our program, but inner products do; and we know

$$\cos\theta(x_u, x_v) \left\lVert x_u\right\rVert_2 \left\lVert x_v\right\rVert_2 = \langle x_u, x_v\rangle.$$

So we get that

$$\mathbb{P}[(u, v) \text{ crosses } (L, R)] = \frac{\arccos(\langle x_u, x_v\rangle)}{\pi}.$$

Now we've found the probability you cross one edge, and we can calculate the expected number of edges cut by (L, R) — by linearity of expectation this is

$$\sum_{e=(u,v)} \frac{\arccos\langle x_u, x_v\rangle}{\pi}.$$

We have this annoying arccos thing, but we know something about the average of $1 - \langle x_u, x_v\rangle$; so what we do now is calculus. We define the function

$$g(z) = \frac{\arccos(z)/\pi}{(1 - z)/2}$$

(where we're taking this function, dividing by the original, and replacing $\langle x_u, x_v\rangle$ with z). We find that $\min_{z\in[-1,1]} g(z) = \alpha_{gw} \approx 0.878$ (we're not going to do this computation, but it is something that can be done). Then we can plug in this fact to replace arccos with $\frac{1}{2}(1 - z)$, to get that the expected number of edges is at least

$$\alpha_{gw} \sum_{e=(u,v)} \frac{1 - \langle x_u, x_v\rangle}{2} \geq \alpha_{gw} \cdot \rho \left|E\right|.$$

And we're done.

> **Remark 16.3.** How do we actually get a cut? It's possible to actually prove that the probability the number of edges is at least $\alpha_{gw} \cdot \rho\left|E\right| - \frac{1}{n}$ is at least $\mathsf{poly}(1/n)$ (this is straightforward, because the number of edges is an integer), so then you can just repeat several times until we get a good cut; but you can do other stuff as well.

This is the algorithm; it seems very weird, and when you see it the first time, there's no reason to believe it's optimal. Another thing which is kind of curious is that when we computed the stability of majority, there was a similar picture (with the two vectors). There's actually a good reason for this.

One remark is that this analysis is pretty general; you can compute some constant z at which this minimum is attained. And you can wonder, what happens when ρ is very close to 1 — does the analysis improve? It turns out the answer is yes.

Theorem 16.4

If $\rho = 1 - \varepsilon$, then this algorithm finds a cut of size at least $(1 - \frac{2}{\pi}\sqrt{\varepsilon} - O(\varepsilon^{1.5}))\,|E|$.

So the point is that as $\varepsilon \to 0$, this also approaches 0; so we find a very good cut. (If we're curious, the proof is the same as ours, but instead of using this loose bound on $\min g(z)$, you replace arccos with a nicer function and use Jensen's inequality — you basically stare at arccos around 1 and do some Taylor expansions, and this is basically what's happening).

Remark 16.5. The worst-case scenario is when all $\langle x_u, x_v\rangle$ are equal to the minimizer of $g(z)$; it turns out that this can actually happen, but if it doesn't (e.g., the above theorem, or if they have some variance) then you can get a better bound.

This was from around 1995 and was a shocker in algorithms; then people tried semidefinite programming and rounding for several discrete optimization problems, and it works (in theory, at least). So it's a very powerful technique, and it's still studied today (though it gets more hairy).

§16.2 Hardness of approximation for Max-Cut

Now we fastforward to around 2005.

Theorem 16.6 (KKMO 2005)

Assuming the unique games conjecture, for all $\rho \in (0, 1)$ and $\varepsilon > 0$, given a graph $G = (V, E)$, it is NP-hard to distinguish between the following cases:

(1) YES case — G has a cut of fractional size at least $\frac{1}{2} + \frac{1}{2}\rho - \varepsilon$.

(2) NO case — G has no cut of size more than $1 - \frac{1}{\pi}\arccos\rho + \varepsilon$.

(The *fractional size* of a cut is its number of edges divided by the total number of edges in the graph.)

This is not fully compatible with the way we presented the Goemans–Williamson algorithm; you need to use some properties of arccos to get one bound from the other. But this number is what you get from the analysis in the end (with $\min g(z)$), and so this tells you that the rounding you got there is the best you can do (it's hard to distinguish between something that has the value of the program, and something that doesn't have the value you're rounding to). So in other words, this means Goemans–Williamson is tight.

This was once again a shocker, since it means the weird-looking analysis and program is for some reason the right thing to do (going to high-dimensional space and finding good vectors and rounding them down).

For the rest of htis lecture, we'll prove this theorem. Of course, we're going to use our analytical stuff.

§16.3 Unique games

First, let's recall what the unique games conjecture is.

Definition 16.7. An instance of UG (*unique games*) consist of a bipartite graph $G = (U \cup V, E)$, an alphabet Σ, and constraints — a collection of permutations $\{\varphi_e \colon \Sigma \to \Sigma\}_{e\in E}$ (with one constraint for each edge).

GOAL: find assignments $A\colon U \to \Sigma$ and $B\colon V \to \Sigma$ maximizing the number of satisfied constraints, i.e.,

$$\#\{(u, v) \in E \mid B(v) = \varphi_{(u,v)}(A(u))\}.$$

Last time we saw a more concrete example that's easier to understand, linear equations in two variables; but this is more easy to work with for reductions.

> **Conjecture 16.8** (Khot 2002) — For all $\varepsilon, \delta > 0$, there exists $k \in \mathbb{N}$ such that given a UG instance $\psi = (G, \Sigma, \Phi)$ with $|\Sigma| \leq k$, it is NP-hard to distinguish the following two cases:
>
> (1) YES case — there exists an assignment (A, B) satisfying at least $(1 - \varepsilon)$ of the constraints.
>
> (2) NO case — every assignment (A, B) satisfies at most δ of the constraints.

In human languages, it's hard to say whether an instance of unique games is *almost fully satisfiable* or whether we can satisfy barely anything. (The point of k is that we want to say the alphabet size is constant.)

When we try to study hardness of approximation, the first reduction is hard to wrap your head around.

> **Remark 16.9.** You should think of ε and δ as small (e.g., $1/100$) — the smaller they are, the easier it is to distinguish between these cases, because they're farther and farther apart.

The way we'll prove our result on Max-Cut is using a reduction — this reduction will take as input an instance of Unique Games, which we'll call ψ. And it'll output a graph H. This reduction should have two features that any reduction should have:

- *Completeness* — if we started with the YES case of the unique games conjecture, then we should land in the YES case of the Max-Cut problem. Explicitly, if $\text{Val}(\psi) \geq 1 - \varepsilon$, then we shoudl have $\text{MC}(H) \geq \frac{1}{2} + \frac{1}{2}\rho - \varepsilon'$. (So if we have an instance of UG where we can do very well, then the graph has a big cut.)
- *Soundness* — if $\text{Val}(\psi) \leq \delta$, then $\text{MC}(H) \leq 1 - \frac{1}{\pi}\arccos(\rho) + \varepsilon'$.

(This is the extension of NP-completeness reductions when we tal kabout approximation algorithms.)

Hopefully it's clear that if we see such a polynomial time reduction, then this theorem is proved — because if we could distinguish between the two cases of Max-Cut, then we could also distinguish between the two cases of Unique Games.

§16.4 Dictatorship vs. non-influential variable paradigm

This is the goal, but achieving this goal outright is too hard. So instead we'll try to do something easier. We're going to try to construct a graph — or rather, a family of graphs — which is going to be an instance of Max-Cut. There's going to be some very good solutions, but they'll be special solutions — achieved by dictatorships.

But the other property we'll want is that if I disallow you to use dictatorships (in a rather strong sense, using influences), then you cannot actually get a cut of that size.

This is on its own a very interesting task, and it turns out that it's important and implies what we want in a black-box way.

So our goal is to design a graph H on the hypercube $\{-1, 1\}^n$ with two properties. On the hypercube, any Boolean-valued function $f\colon \{-1, 1\}^n \to \{-1, 1\}$ represents a cut. So we want the following two properties:

(1) Any dictatorship, namely a cut of the form

$$L = \{x \mid x_i = -1\},\ R = \{x \mid x_i = 1\},$$

has a large size.

(2) Any cut which is 'far' from dictatorships — specifically, a cut defined by a function $f\colon \{-1, 1\}^n \to \{-1, 1\}$ with $\max I_i[f] \leq \tau$, has small size. More specifically, we actually want $\max I_i^{\leq d}[f] \leq \tau$.

(By 'large' and 'small' we mean the quantities from earlier in the completeness and soundness properties.)

We can start by trying to get a graph obeying the first property. (Think of ρ as close to 1 — so we want a dictatorship to get almost all the edges.)

We can imagine connecting x and $-x$ — then dictatorships give you a perfect cut (it crosses all the edges, because $x_i = -x_i$ for all i). That gets the first property, but in too strong of a sense — then it must be the case that the second property fails. We claim that there are good cuts in this graph that are not dictatorships — for example, majority works, as does $\{x \mid \prod_{i \in S} x_i = 1\} \cup \{x \mid \prod_{i \in S} x_i = -1\}$ where S is any set of odd size. So the second property fails miserably.

Now our goal is to modify this simple graph to destroy this type of cut (think of S as large). We're going to penalize all the cuts — there's no way around that. But we'd like to penalize this type of cut much more than dictatorships.

And the way we do this is by using the noise operator. Think of ρ as close to 1 (i.e., $\rho = 1 - \varepsilon$). We'll take the *noisy hypercube* with correlation $-\rho$, with edge weights according to the following process — first we choose $x \in \{-1, 1\}^n$ uniformly at random, and set $\widetilde{x} = -x$. And then we take a noisy copy of $\widetilde{x}$ — we define $y \sim T_\rho \widetilde{x}$, and we output (x, y). And these are the edges of our graph.

Formally speaking, this is actually a weighted graph — the vertices are $\{-1, 1\}^n$. The edges are everything, but that doesn't mean anything; we define the weight of each edge as the probability it gets outputted by this process. (This will produce a weighted instance of Max-Cut, but there's actually a way to go from weighted to unweighted instances.)

(You can observe that the inner products here are going to almost always be $-\rho$, so this corresponds in some sense to our SDP.)

Here are two facts:

Fact 16.10 — For every i, the size of the cut $f(x) = x_i$ is $\frac{1}{2} + \frac{1}{2}\rho$.

Proof. If we look at $(x, \widetilde{x})$, it crosses the cut. And the probability we don't flip the ith coordinate when going from $\widetilde{x}$ to i is

$$\mathbb{P}[y_i = \widetilde{x}_i] = \rho + (1 - \rho) \cdot \frac{1}{2} = \frac{1}{2} + \frac{1}{2}\rho. \qquad \square$$

Fact 16.11 — If $f\colon \{-1, 1\}^n \to \{-1, 1\}$ satisfies $\mathbb{E} f = 0$ and $\max_i I_i^{\leq d}[f] \leq \tau$, then

$$\operatorname{cut}(f) \leq 1 - \frac{1}{\pi} \arccos(\rho) + o(1).$$

Proof. The point is that we can directly relate the size of the cut to the stability (which comes from the inner product of f and a noisy version) — we get

$$\mathbb{P}_{(x,y)}[f(x) \neq f(y)] = \frac{1}{2}(1 - \operatorname{Stab}_{-\rho}(f)) \leq \frac{1}{2}(1 + \operatorname{Stab}_\rho(f))$$

(if you play around with the Fourier-analytic formula, you can pull the $-$ outside and this only increases the value). And then the majority is stablest theorem gives us an upper bound on the right-hand side — it says

$$\operatorname{Stab}_\rho(f) \leq 1 - \frac{2}{\pi} \arccos \rho + o(1),$$

which gives exactly what we want. □

(The $o(1)$ is as $d \to \infty$, and then as $\tau \to 0$.)

§16.5 Reduction from Unique Games to Max-Cut

So we've solved a mini-problem that looks like what we wanted — we designed a graph that has very good structured solutions, and anything that is not structured is small. Now we're going to try to use this to do our reduction.

Imagine we have $\psi = ((U \cup V, E), \Sigma, \Phi)$. What we want to do is use this gadget we constructed here to do the reduction.

The idea is that if we look at the unique games problem, a solution to it is a labelling assignment; and if we look at the max-cut problem, a solution to it is a cut. So we want to try to encode an assignment of labels via cuts in some sense.

The way to do this is to use the *long code* on the hypercube. So we consider $\{-1,1\}^\Sigma$ (where Σ is the alphabet). If we choose a label for a vertex, this is like choosing a coordinate in this cube, which is like choosing a cut. So the idea is to replace each one of the vertices in this graph with a copy of the hypercube, looking at the correspondence between cuts and assignments, and trying to argue about it.

First we'll do this the wrong way, and then we'll fix it. We wish to 'encode' an assignment σ to a vertex v using the corresponding dictatorship cut $f_v(x) = x_\sigma$. (If we have an assignment σ, then we can naturally look at the dictatorship it defines; this is going to be the correspondence that we keep in mind.)

So we replace the vertex v with $\{v\} \times \{-1,1\}^\Sigma$ (a copy of the hypercube $\{-1,1\}^\Sigma$). Our goal is for the edges we draw to ensure that a good cut here corresponds to encoding a valid assignment (since not every cut defines a dictatorship).

This is where the earlier construction comes into play — suppose we put the edges from that noisy hypercube into the graph. Then if we get a good cut here, it must be that you're following a dictatorship-style strategy, so there is some label getting encoded here. (This is a bit tricky, but the idea is there.)

In other words, we want to put edges on this graph to ensure that a good cut actually corresponds to a labelling of v. And for that, we can put the edges of the noisy hypercube from before.

So now we had our graph $(U \cup V, E)$. And we replace it with a graph where we still have two sides U and V, but each vertex has been replaed with a copy $\{v\} \times \{-1,1\}^\Sigma$ of the hypercube (with some edges inside it).

That's the first transformation. Now at least we know that whenever we have a good cut here, it kind of intuitively corresponds to an assignment for unique games.

But we lost all the information about the instance of unique games — we've checked the labels are valid, but we haven't checked the constraints φ_e. This is a big issue — which means this will not work.

(We will continue this on Thursday, and maybe start a similar hardness result for vertex cover.)

§17 April 11, 2024

§17.1 Hardness of approximating max-cut

Recall that we're trying to prove the following result,

Theorem 17.1

Assuming the unique games conjecture, for all $\rho \in [0,1]$ and $\varepsilon > 0$, given a graph G, it is NP-hard to distinguish between:

- YES: $\mathrm{MC}(H) \geq \frac{1}{2} + \frac{1}{2}\rho - \varepsilon$.
- NO: $\mathrm{MC}(H) \leq 1 - \frac{1}{\pi}\arccos\rho + \varepsilon$.

Last time we started talking about the reduction. We'll call our graph H and the unique games instance $\psi = (G, \Sigma, \Phi = \{\varphi_e\})$.

§17.1.1 The long code and noisy hypercube

So far, we've discussed one important idea in the reduction — the *long code*. In unique games, we're supposed to assign labels to the vertices satisfying as many constraints as possible. But in max-cut we just have a cut, not assignments; so how can we encode labels to vertices using cuts?

The idea is that to encode a label $\sigma \in \Sigma$, we think about the dictatorship function $f_\sigma \colon \{-1,1\}^\Sigma \to \{-1,1\}$ defined according to σ, i.e., $f_\sigma(x) = x_\sigma$. So far, we haven't done anything — it's just a mental exercise, where we shift from σ to a Boolean function.

To enforce this, we want to design a graph on the hypercube such that such functions correspond to good cuts; and if your function doesn't resemble a dictatorship whatsoever, it corresponds to a bad cut. We did this last time, and we arrived at the *noisy hypercube* with correlation $-\rho$.

As a recap, the edges are weighted as follows: we sample $x \in \{-1,1\}^\Sigma$ uniformly at random, and take $\widetilde{x} = -x$; and then we take $y \sim T_\rho \widetilde{x}$, and output (x, y). (The weight of each edge is its probability of being outputted under this process.) This has the following properties:

- The cut defined by f_σ is large — $\operatorname{cut}(f_\sigma) \geq \frac{1}{2} + \frac{1}{2}\rho$.
- For every f which doesn't resemble a dictatorship whatsoever — that $\mathbb{E} f = 0$ and $\max_i I_i^{\leq d}[f] \leq \tau$ — we have
$$\operatorname{cut}(f) \leq 1 - \frac{1}{\pi}\arccos(\rho) + o(1).$$

§17.1.2 A first attempt

So far, we've talked about how to think about labels and associate local instances of Max-Cut with checking that our labelling works.

But now we want an actual reduction. Our first attempt was the following — given ψ, we inflate it to an instance where we replace each vertex with a copy of the hypercube. So if our two vertex sets are $U \cup V$, then for each vertex $u \in U$ we'll create a cloud $\{u\} \times \{-1,1\}^\Sigma$ (a copy of the hypercube labelled u) (and we do this for all vertices in both sides). We plant the noisy hypercube in each of these things; and now these local graphs check that indeed we're supposed to be assigning a labelling.

But this has a big issue — where are the constraints? If in unique games we just wanted to assign labels to the vertices and nothing else, it'd be very easy. So we need to somehow incorporate the constraints.

§17.1.3 Incorporating the constraints

Suppose that we have a permutation $\varphi \colon \Sigma \to \Sigma$. Abusing notation, we claim that we can view φ as an isomorphism $\widetilde{\varphi} \colon \{-1,1\}^\Sigma \to \{-1,1\}^\Sigma$ between two copies of the hypercube — whenever we have a permutation, we can get such an isomorphism using it. We do this just by permuting the coordinates — we define $\widetilde{\varphi}(x)$ to be the vector y such that $y_i = x_{\varphi(i)}$.

Now let's try to wrap our heads around this. We're using this word *isomorphism*, which doesn't really have any meaning yet. But if you think about the noisy hypercube on the left and apply this isomorphism, you get exactly the same noisy hypercube — so this permutation preserves the edges weights (because we're just relabelling the coordinates). So this is a way to translate the language of one hypercube to another; and it'll be the key to making our construction less ridiculous.

So here's our second attempt. (The first attempt was to just put edges inside and not worry about anything.)

Suppose that there used to be an edge between $u_1 \in U$ and $v_1 \in V$. On this edge, we had a constraint $\varphi_{u_1v_1}$, corresponding to a permutation. So what we can do is if we wanted to put an edge xy on the left, we instead shift y to the right, and consider the edge between x on the left and $\widetilde{\varphi}_{u_1v_1}(y)$ on the right. So instead of doing a noisy hypercube inside, we're doing it across — this is kind of a bipartite version of the noisy hypercube, except with some relabelling.

(So we've replaced the old edge xy in u_1's hypercube with the new edge $x\widetilde{\varphi}_{u_1v_1}(y)$ going across.)

The idea is that if labels correspond according to this matching, then you can compute that the number of edges you cut across is exactly $\text{cut}(f_\sigma)$ (since the question of whether this edge crosses our cut is the same as for the original).

There are still issues with this (we're going to write something concrete once we have no issues). But the idea morally is that instead of doing edges inside we're doing edges across, and we go across using the isomorphisms coming from the constraints of unique games. (The constraint tells you that your labellings $A(u_1)$ and $B(v_1)$ should satisfy $A(u_1) = \varphi_{u_1v_1}(B(v_1))$; and we're kind of mimicking this on the hypercube.)

The problem is that our graph now is a bipartite graph, so there is a very good Max-Cut (getting all the edges).

§17.1.4 The actual construction

Now we'll fix this final issue. We take $U \times \{-1,1\}^\Sigma$ on the left and $V \times \{-1,1\}^\Sigma$ on the right (as before). Let's focus on one specific vertex on the left, which we call u — so it corresponds to $\{u\} \times \{-1,1\}^\Sigma$. (We'll do this for each vertex u on the left.)

We take our vertex on the left; the left is actually going to be imaginary, and only the right will exist in our graph. So we *imagine* taking u on the left, and sampling an edge (x, y) in its hypercube. Now we sample two neighbors v_1 and v_2 of u randomly; and we drive one of x and y to v_1, and the other to v_2. So we look at $\widetilde{\varphi}_{uv_1}(x)$ and $\widetilde{\varphi}_{uv_2}(y)$, and we draw the edge between them. (So U is imaginary, and V is what's actually happening.) And that's how the construction works.

For simplicity of notation, let's assume that G is bi-regular, so all vertices in U have the same degree and all vertices in V have the same degree. Then we construct H as follows: first, the vertices are $V \times \{-1,1\}^\Sigma$ (only the right side of our picture). For the edges, we'll obtain the edge weights according to the following process (we're just repeating the same thing, more formally):

- Sample $u \in U$.
- Sample an edge (x, y) inside the noisy hypercube of u.
- Sample two neighbors $v_1, v_2 \sim \mathcal{N}(u)$ independently.
- Output the edge between the vertices $(v_1, \varphi_{uv_1}(x))$ (where x is mapped to in the v_1-hypercube) and $(v_2, \varphi_{uv_2}(y))$.

This completes the description of the reduction.

Our goal is to show that:

(1) If the unique games instance is highly satisfiable — $\text{val}(\psi) \geq 1 - \eta$ — then the graph H has a large cut — $\text{MC}(H) \geq \frac{1}{2} + \frac{1}{2}\rho - \varepsilon$.

(2) If $\text{val}(\psi) \leq \eta$, then $\text{MC}(H) \leq 1 - \frac{1}{\pi}\arccos(\rho) + \varepsilon$.

(The quantifiers are 'for every ε, there exists $\eta \ldots$.')

The first property is called completeness, and the second soundness.

§17.1.5 Completeness

Here we want to show that if ψ is almost completely satisfiable, we get something with a large cut. So let $A: U \to \Sigma$ and $B: V \to \Sigma$ be assignments to ψ satisfying at least $1-\eta$ of the constraints.

Claim 17.2 — For u, v_1, and v_2 sampled as in our reduction, we have

$$\mathbb{P}_{u,v_1,v_2}[(u, v_1) \text{ and } (u, v_2) \text{ satisfied}] \geq 1 - 2\eta.$$

Proof. The marginal distribution of (u, v_1) is a uniform edge in the graph, and so is (u, v_2). So by the union bound, the probability that one is unsatisfied is at most the sum of probabilities, which is at most 2η. □

Call such (u, v_1, v_2) *nice* (i.e., if both (u, v_1) and (u, v_2) are satisfied).

Now we need to define a cut in the graph — we define the cut $f: V \times \{-1, 1\}^\Sigma \to \{-1, 1\}$ as $f(v, x) = x_{B(v)}$ (the vertices in our graph are $V \times \{-1, 1\}^\Sigma$, and to label them, we look at the coordinate of x corresponding to the label of v).

Claim 17.3 — If (u, v_1, v_2) is nice, then for x and y chosen as in the reduction, the probability that the edge they generate crosses the cut is

$$\mathbb{P}_{(x,y)}[f(v_1, \varphi_{uv_1}(x)) \neq f(v_2, \varphi_{uv_2}(y))] = \frac{1}{2} + \frac{1}{2}\rho.$$

Proof. Let's see what these functions evaluate to — $f(v_1, \varphi_{uv_1}(x))$ is the $B(v_1)$th coordinate of $\varphi_{uv_1}(x)$. And φ_{uv_1} is satisfied, so this is the same as the $\varphi_{uv_1}(A(u))$th coordinate of φ_{uv_1}, i.e.,

$$\varphi_{uv_1}(x)_{\varphi_{uv_1}(A(u))} = x_{A(u)}.$$

This is a lot of notation, but really what we're saying is that whatever the dictatorship said on u, if the constraint is satisfied, the corresponding dictatorship on v_1 says the same thing. And running the same logic, we have

$$f(v_2, \varphi_{uv_2}(y)) = y_{A(u)}.$$

These two facts together give that the probability we want is

$$\mathbb{P}_{x,y}[x_{A(u)} \neq y_{A(u)}],$$

which is exactly what we computed when we just analyzed a single noisy cube — it's $\frac{1}{2} + \frac{1}{2}\rho$. □

So almost all (u, v_1, v_2) are nice, and whenever they're nice they contribute this much to the cut. If they're not nice, we don't know what to say, but they certainly can't take away edges from the cut (they have nonnegative contribution). So we get

$$\text{cut}(f) \geq (1 - 2\eta)\left(\frac{1}{2} + \frac{1}{2}\rho\right) \geq \frac{1}{2} + \frac{1}{2}\rho - 2\eta.$$

So we've proven completeness.

Remark 17.4. There's no probability in the actual reduction (though probability is the easiest way to phrase things) — you can compute all the relevant probabilities to get explicit edge weights. (There's as much probability here as in the probabilistic method of combinatorics, which is a fancy way of saying 'counting' but is much easier to phrase in terms of probability; the same thing is happening here.)

§17.1.6 Soundness

In most NP-hardness reductions soundness is the harder part, and this problem is no exception. We need to show that if the original unique games instance has very bad satisfiability, then any cut is weak. The way to prove this is the other way around — we show that if H has a good cut, then we can salvage from it, in some bizarre way, a good assignment to ψ. This is what we're going to do.

Suppose that $f\colon V \times \{-1,1\}^\Sigma \to \{-1,1\}$ is a cut of fractional size at least $1 - \frac{1}{\pi}\arccos(\rho) + \varepsilon$. Now we want to use this cut to find an assignment to ψ. And it's completely unclear how you do this.

The first step is just notation — define $f_v\colon \{-1,1\}^\Sigma \to \{-1,1\}$ by $f_v(x) = f(v,x)$. We haven't done anything here; but eventually the labels we'll give are based on the influences of this function — when we want to choose a label of v, we'll look at this function and come up with an assignment based on its influences.

But we also need a label for u. And u doesn't have a corresponding function — it doesn't even exist in the reduction. So the point is that for each $u \in U$, we define $f_u(x)$ by 'asking one of our friends' — we sample a neighbor v, and definen

$$f_u(x) = \mathbb{E}_{v\sim\mathcal{N}(u)} f_v(\widetilde{\varphi}_{uv}(x)).$$

So u doesn't have a function, which means we get a function by sampling a neighbor v, seeing what x got assigned there according to the isomorphism, and then taking an average over v. This isn't a Boolean function, but it is bounded — so $f_u\colon \{-1,1\}^\Sigma \to [-1,1]$.

Now we want to express $\operatorname{cut}(f)$ in terms of these functions — we have

$$\operatorname{cut}(f) = \mathbb{P}_{u,x,y,v_1,v_2}[f_{v_1}(\varphi_{uv_1}(x)) \neq f_{v_2}(\varphi_{uv_2}(y))].$$

Probabilities are annoying, so let's try to write this as an expectation — as usual when we have two bits and want to measure when they're different, we look at their product (which is -1 if they're different and 1 if they're the same), so that

$$\operatorname{cut}(f) = \mathbb{E}_{u,x,y}\left[\mathbb{E}_{v_1,v_2\sim\mathcal{N}(u)} \frac{1 - f_{v_1}(\varphi_{uv_1}(x)) f_{v_2}(\varphi_{uv_2}(y))}{2}\right].$$

(Lots of the φ's should be $\widetilde{\varphi}$'s, in the notation we're using; but you can think about them as the same.) First we'd like to write the inner expectation in terms of our function f_u — conditioned on u, x, and y, the values of v_1 and v_2 are independent, and the expectations of each term is a value of f_u, so this is

$$\mathbb{E}_{u,x,y}\left[\frac{1}{2} - \frac{1}{2} f_u(x) f_u(y)\right] = \frac{1}{2} - \frac{1}{2}\mathbb{E}_u \mathbb{E}_{x,y} f_u(x) f_u(y).$$

And this is exactly

$$\operatorname{cut}(f) = \frac{1}{2} - \frac{1}{2}\mathbb{E}_u \operatorname{Stab}_{-\rho}(f_u)$$

(since the distribution over (x,y) is a $-\rho$-correlated pair).

So this is one big line computing $\operatorname{cut}(f)$ in terms of f_u. But we know the cut size is large. So now what we're going to do is rearrange this to say that this stability thing is at most something — since $\operatorname{cut}(f) \geq 1 - \frac{1}{\pi}\arccos(\rho) + \varepsilon$, we get that

$$\mathbb{E}_u \operatorname{Stab}_{-\rho}(f_u) \leq \frac{2}{\pi}\arccos(\rho) - 1 - 2\varepsilon.$$

(We just rearranged the two above equations.) So the expected stability of f_u is a bit smaller than this number $\frac{2}{\pi}\arccos(\rho) - 1$, which means by an averaging argument that

$$\mathbb{P}_u\left[\operatorname{Stab}_{-\rho}(f_u) \leq \frac{2}{\pi}\arccos(\rho) - 1 - \varepsilon\right] \geq \varepsilon.$$

So there's a noticeable fraction of u's for which the stability is less than $\frac{2}{\pi}\arccos(\rho)-1-\varepsilon$; we say u is *good* if

$$\operatorname{Stab}-\rho(f_u)\leq\frac{2}{\pi}\arccos(\rho)-1-\varepsilon.$$

The unfortunate thing about this reduction is that although this number looks like what's in majority is stablest, it looks a bit messed up. And the reason for that is that here we're looking at negative correlations, whereas majority is stablest looks at positive correlations. So we'll now state majority is stablest adapted for negative noise rates.

Theorem 17.5

For every $\rho\in(0,1)$ and $\varepsilon>0$, there exists $d\in\mathbb{N}$ and $\tau>0$ such that if $g\colon\{-1,1\}^n\to[-1,1]$ is such that $\max_i I_i^{\leq d}[g]\leq\tau$, then

$$\operatorname{Stab}_{-\rho}(g)\geq\frac{2}{\pi}\arccos(\rho)-1-\varepsilon.$$

Proof sketch. The idea is that we define the odd part of g as

$$g_{\mathrm{odd}}(x)=\sum_{|S|\text{ odd}}\widehat{g}(S)\chi_S(x)=\frac{g(x)-g(-x)}{2}$$

(the first expression is more convenient, and the second shows that this is still between -1 and 1). And for $\operatorname{Stab}_{-\rho}(g)$, the odd things give us negative terms and the even things give us positive terms, so

$$\operatorname{Stab}_{-\rho}(g)\geq\operatorname{Stab}_{-\rho}(g_{\mathrm{odd}})=-\operatorname{Stab}_{\rho}(g_{\mathrm{odd}}).$$

And $\operatorname{Stab}_\rho(g_{\mathrm{odd}})$ is upper-bounded by the thing in majority is stablest, which because of the negative sign gives us an upper bound on $\operatorname{Stab}_{-\rho}(g)$. □

So then our functions f_u violate the statement of majority is stablest, and the only explanation for that is they have influential variables. So we take d and τ from the theorem, and we define the list of u as

$$\mathcal{L}(u)=\{\sigma\in\Sigma\mid I_\sigma^{\leq d}[f_u]\geq\tau\}.$$

Claim 17.6 — If u is good, then $\mathcal{L}(u)$ is nonempty.

Proof. If u is good, then f_u has stability too small, which violates the theorem; so the theorem means you have some influential variable. □

The next claim is that these lists are not too big.

Claim 17.7 — For every u, we have $|\mathcal{L}(u)|\leq d/\tau$.

Proof. This was on the problem set — we have $\sum_\sigma I_\sigma^{\leq d}[f_u]=\sum_{|S|\leq d}|S|\,\widehat{f_u}(S)^2\leq d$. So when we sum the low-degree influences we get at most d; this means at most d/τ of them can be at least τ. □

So now the life of u has improved dramatically — at the beginning of the analysis it didn't even have a function, and now it not only has a function but a list of possible labels we can choose from, and there's not too many. But now v is in bad shape — we need a list of labels for it too.

So we can define a similar list $\mathcal{L}(v)$, but we need different parameters here — for each $v\in V$, we define

$$\mathcal{L}(v)=\left\{\sigma\in\Sigma\mid I_\sigma^{\leq d}[f_v]\geq\tau/2\right\}.$$

(We take the same type of thing, but with $\tau/2$ instead of τ.)

Claim 17.8 — $|\mathcal{L}(v)| \leq 2d/\tau$ for all v.

So these lists are also not large, but they might be empty. So this is what we need to eliminate. And this brings us to the final claim, which is very nice.

Claim 17.9 — Let $u \in U$ be good, and suppose that $I_\sigma^{\leq d}[f_u] \geq \tau$. Then if we choose v to be a random neighbor of u and look at the influence of where σ is mapped to, we have

$$\mathbb{E}_{v\sim\mathcal{N}(u)} I_{\varphi_{uv}(\sigma)}^{\leq d}[f_v] \geq I_\sigma^{\leq d}[f_u] \geq \tau.$$

We need to do two things — prove this claim and show that given it, we're done. We'll prove this claim first.

Proof. For the proof, we just spell out what the influence is — we have

$$I_\sigma^{\leq d}[f_u] = \sum_{|S|\leq d, \sigma\in S} \widehat{f_u}(S)^2.$$

And f_u is an average over f_v's, so we can write

$$\widehat{f_u}(S) = \mathbb{E}_v \widehat{f_v}(\varphi_{uv}(S)),$$

which turns this into

$$\sum_{|S|\leq d, \sigma\in S} \left(\mathbb{E}_v \widehat{f_v}(\varphi_{uv}(S))\right)^2.$$

And now we have a square of an expectation, so we're going to push the square inside using Jensen; this gives

$$I_\sigma^{\leq d}[f_u] \leq \sum_{|S|\leq d, \sigma\in S} \mathbb{E}_v \widehat{f_v}(\varphi_{uv}(S))^2.$$

And we can reparametrize the inside sum, letting $T = \varphi_{uv}(S)$ — since φ_{uv} is a permutation it preserves sizes, so we get

$$\mathbb{E}_v \sum_{|T|\leq d, \varphi_{uv}(\sigma)\in T} \widehat{f_v}(T)^2 = \mathbb{E}_v I_{\varphi_{uv}(\sigma)}^{\leq d}[f_v].$$ □

Now we're almost done. For u we're completely fine — we know its list is nonempty for many good u's, and it's never too large. For v we've only said that it's never too large. But now we can also say it's nonempty the same number of times.

Claim 17.10 — Suppose that u is good, and let $\sigma \in \mathcal{L}(u)$. Then

$$\mathbb{P}_{v\sim\mathcal{N}(u)}[\varphi_{uv}(\sigma) \in \mathcal{L}(v)] \geq \frac{\tau}{2}.$$

Proof. This is an averaging argument — $I_{\varphi_{uv}(\sigma)}^{\leq d}[f_v]$ is a random variable whose average is at least τ, and it's always between 0 and 1, so it must be at least $\tau/2$ with probability at least $\tau/2$. □

And with all of this, we're finally ready to give the assignment for unique games.

(1) For each $u \in U$, choose $A(u)$ uniformly from $\mathcal{L}(u)$. (If the list is empty, we can either not give an assignment or give something random.)

(2) Do the same for each $v \in V$.

So we've got a randomized assignment; and we're going to lower-bound the expected number of constraints of ψ that it satisfies, i.e.,

$$\mathbb{E}_{A,B}[\text{fraction of constraints } uv \text{ satisfied}].$$

First, we'll only focus on good u's — we said the fraction of good u's is at least ε, so we'll write down an ε. Now u is good, and we chose some label of it in the first step; let's call this σ. Now we choose a random neighbor v of u; and we know from the above claim that the matching label to σ is in $\mathcal{L}(v)$ with probability $\tau/2$. So there's a $\tau/2$ probability for the eligible symbol for v to even exist in its list. But v chooses something random from its list; so the probability it hits the right one is 1 over the list size. But we know the list size isn't too large, so this is at least $1/(2d/\tau)$. And so we get

$$\mathbb{E} \geq \varepsilon \cdot \frac{\tau}{2} \cdot \frac{1}{2d/\tau} = \frac{\tau^2\varepsilon}{4d} > \eta.$$

The only important feature of this number is that it only depends on the ε we started with (and not alphabet size); so if you take the alphabet of the unique games instance to be large enough, this will increase the alphabet size, but this number will be unaffected. And so for sufficiently good soundness of unique games, this is less than what we get — in other words, $\text{val}(\psi) > \eta$.

So this is how proofs in hardness of approximation look like.

Remark 17.11. This is actually one of the easier reductions. It's hard to appreciate because this is the first proof we're seeing, but this whole trick where you keep one side and do the other side as the average is only possible because of unique games. If you don't have unique games, you're in trouble and things are much worse.

Eventually, you reduce to a combinatorial problem — the stability of some function being smaller than something (this isn't about two different functions or something like that), and so all this analysis depended on the noisy hypercube being the right thing and all the rest of the steps are automatic in some sense.

But this proof didn't come out of thin air. Before that people did NP-hardness, where the reductions were more difficult and ad-hoc, but did use Fourier analysis. But the previous proofs didn't use analytical theorems like majority is stablest; instead they used third powers of Fourier coefficients-type analysis (as we saw in linearity testing). So this paper did introduce many new ideas. But in terms of the simplicity of reductions, this is as simple as it gets.

(They sort of suggested a general way of doing reductions, but didn't do it generally — as mentioned earlier, they didn't prove the majority is stablest theorem and instead conjectured it, and it was proven two years later.)

(This paper was by Khot–Kindler–Mossel–O'Donnell.)

Next time, we're going to see an earlier result about vertex cover. Dor hopes to give us a general view of how these UGC reductions work. Right now we've seen one huge reduction which looks very complicated; but we'll then see one more reduction and hopefully see the similarities and differences.

§18 April 16, 2024

Today we'll discuss another hardness of approximation result. Last time we saw a proof that assuming UGC, the algorithm we saw for Max-Cut is optimal. Today we'll see another important reduction by Khot and Regev showing that under the same assumption, the trivial approximation algorithm for Vertex-Cover is also the best you can do.

§18.1 Vertex covers and independent sets

Definition 18.1. Given a graph $G = (V, E)$, a set $C \subseteq V$ is called a *vertex cover* if for every edge $e = (u, v) \in E$, either u or v is in C — equivalently, $\{u, v\} \cap C \neq \emptyset$.

Given a graph, the problem we care about is finding the *smallest* possible vertex cover inside it.

Definition 18.2. We define $\mathsf{VC}(G) = \min |C|$ over all $C \subseteq V$ that are vertex covers.

So that's the vertex cover problem. But for technical reasons, it'll be easier for us to work with a related problem, the independent set problem.

Definition 18.3. Given a graph $G = (V, E)$, a set $I \subseteq V$ is called an *independent set* if it contains no edges — i.e., for all $e = (u, v) \in E$, we have $\{u, v\} \not\subseteq I$.

For independent sets, we want to find the *largest* possible independent set in the graph.

Definition 18.4. We define $\mathsf{IS}(G) = \max |I|$ over all $I \subseteq V$ that are independent sets.

These are the two problems that will concern us today. The relation between these two problems is that if you take a vertex cover, you have at least one vertex from each set — so if you throw those vertices away, you can't have any edges left. In other words, if C is a vertex cover, then $V \setminus C$ is an independent set. This also goes in the other way — if I is an independent set, then $V \setminus I$ is a vertex cover. So there's a tight connection between VC and IS — we have

$$\mathsf{VC}(G) = n - \mathsf{IS}(G).$$

So computing them is exactly equivalent.

Our main theorem for today is the following.

Theorem 18.5

Assuming the unique games conjecture, for every $\varepsilon > 0$, given a graph $G = (V, E)$, it is NP-hard to distinguish between the following two cases:

- The YES case — $\mathsf{IS}(G) \geq (\frac{1}{2} - \varepsilon) |V|$.
- The NO case — $\mathsf{IS}(G) \leq \varepsilon |V|$.

So we're trying to distinguish between the cases where V contains a huge independent set — nearly half the vertices — and the cases where it doesn't even have one containing an ε-fraction of the vertices. (This is the best you can do — if you increase the constant above $\frac{1}{2}$, there are actually algorithms.)

Remark 18.6. For approximation algorithms for independent sets, the best approximation algorithm we have is $n/\log^2 n$, and the best hardness result is something close to this — something like $n \cdot 2^{-\sqrt{\log n}}$. But this uses different methods.

We promised a result about vertex cover, and we can get one by using the above connection:

Corollary 18.7

Assuming the unique games conjecture, for every $\varepsilon > 0$, given a graph G, it is NP-hard to distinguish:

- The YES case, where $\mathsf{VC}(G) \le (\frac{1}{2} + \varepsilon)\,|V|$.
- The NO case, where $\mathsf{VC}(G) \ge (1 - \varepsilon)\,|V|$.

This means it's NP-hard to approximate vertex cover to a number which is the ratio of these two quantities, i.e., $2 - \Theta(\varepsilon)$.

This is quite remarkable — it gives you a tight hardness result for Vertex-Cover. This is from 2003; historically, this is the first tight hardness result using UGC. Before that people didn't really care about it; and after this people started caring (and they started caring even more after Max-Cut).

§18.2 Proof overview

We're now going to prove this theorem. We'll use the same framework we used before. When we talked about Max-Cut, we first designed a specific graph that did have good max cuts, but they were very structured — there were good cuts corresponding to dictatorships, and if you weren't a dictatorship, your cuts would be pretty bad. We'll do something similar here — we'll design a graph on the hypercube such that dictatorships give large independent sets, and things far from dictatorships give you much smaller independent sets. Then we'll do something called *composition*. (We'll see that we've somewhat lied, but we'll see how to fix that.)

§18.3 The p-biased Kneser graph

Instead of the noisy hypercube, we'll use a construction called the *p-biased Kneser graph*.

Throughout we'll have a parameter p, the bias of our hypercube — we'll set $p = \frac{1}{2} - \varepsilon$.

We usually think of the cube as $\{0,1\}^n$, but in this context it's a bit more natural to think of subsets of $[n]$ — so we'll think of the cube as $\mathcal{P}([n])$. And the p-biased measure, as always, is defined as the measure $\mu_p \colon \mathcal{P}([n]) \to [0,1]$ where

$$\mu_p(A) = p^{|A|}(1-p)^{n-|A|}.$$

In other words, if we want to sample a set A according to μ_p, we look at each coordinate $i \in [n]$, and we include it in A with probability p (and exclude it with probability $1 - p$).

Example 18.8

Consider a dictatorship $\mathcal{F}_i = \{A \subseteq [n] \mid i \in A\}$. Then $\mu_p(\mathcal{F}_i) = p$.

(This is because we're asking what the probability is, if we sample A according to μ_p, that it contains i; and from the above process this is exactly p.)

These will be the vertices of our graph. As usual, we'll take a weighted graph (and μ_p will correspond to the vertex weights); but now we need to put edges.

The goal is to keep dictatorships to be independent sets — we never want to put edges inside a set $\mathcal{F}_i$. But we want to punish everything that is not a dictatorship. So how do we do this?

If we have two subsets $A, B \subseteq [n]$ that are disjoint, then they cannot be in the same dictatorship (because if they were in the same dictatorship, then i would be in both of them). So these are the edges we're going to put.

Definition 18.9. The *p-biased Kneser graph* has vertex set $\mathcal{P}([n])$, μ_p as vertex weights, and edges

$$E = \{(A, B) \mid A \cap B = \emptyset\}.$$

Hopefully this gives intuition why we need $p < \frac{1}{2}$ — if $p > \frac{1}{2}$, then since whenever A and B have sizes greater than $\frac{1}{2}n$ they definitely intersect, the definition would be meaningless. But for p a bit below $\frac{1}{2}$ this really is meaningful, and something will happen.

Now let's see some properties of this graph.

Claim 18.10 — Each dictatorship $\mathcal{F}_i$ is an independent set of weight p.

This corresponds to the gadget in the max cut problem having a large cut along dictatorships.

The next claim (which we'll phrase in a weird way, but actually it's something we've already discussed):

Claim 18.11 — If $\mathcal{I} \subseteq \mathcal{P}([n])$ is an independent set of weight $\mu_p(\mathcal{I}) \geq \delta$, then $\mathcal{I}$ must 'weakly' resemble a dictatorship-like family.

We're writing a bunch of vague words because this is hard to describe precisely. But we claim we've actually been discussing this object (independent sets in the Kneser graph with nontrivial measure) at some earlier point under a different name — intersecting families. What does it mean that $\mathcal{I}$ is an independent set? It means whenever we take two elements inside it, there's no edge between them, which means they have nontrivial intersection. So $\mathcal{I} \subseteq \mathcal{P}([n])$ being an independent set is equivalent to $\mathcal{I}$ being an intersecting family.

And earlier in the course, we proved that if $\mathcal{I}$ is a nontrivial intersecting family, then it must be almost contained in a junta. So for the purposes of this lecture, you should think of independent sets as being juntas. This is a big lie; when we do the proof we'll make it a smaller lie (but still a lie).

So this is the dichotomy; dictatorships give you good solutions, and any even slightly good solution is kind of like a dictatorship.

Remark 18.12. Many times with these reductions you need this kind of structure result, where any kind-of good solution must be something like a dictatorship. (You have a graph structure where you want dictatorships to be good solutions, and anything kind of good to be slightly like a dictatorship.)

This finishes the first part — defining the combinatorial object.

§18.4 Strongish Unique Games

We actually need a slightly different form of unique games for our reduction.

Definition 18.13 (Strongish Unique Games)
Fix parameters $t \in \mathbb{N}$ and $\eta > 0$ (which is small), and suppose we are given a non-bipartite instance of unique games — i.e., $\psi = ((X, E), \Sigma, \{\varphi_e\}_{e \in E})$. The goal is to distinguish between the following two cases:

- The YES case — there exists $X' \subseteq X$ with $|X'| \geq (1 - \eta)|X|$ inside which you can satisfy all the constraints, meaning that there is an assignment $A \colon X' \to \Sigma$ satisfying all constraints inside X'.
- The NO case — for every $X' \subseteq X$ with $|X'| \geq \eta |X|$, and for every *multi-assignment* $A \colon X' \to \binom{\Sigma}{t}$, there is an edge $e = (u, v)$ in X' such that no pair of assignments in $A(u)$ and $A(v)$ satisfy φ_e.

In unique games, the YES case is that you can satisfy $(1-\varepsilon)$ of the constraints. This is slightly different — here you want to satisfy all the constraints in a reasonably big subset. In the grand scheme, you're still satisfying roughly $1-\eta$ of the constraints (assuming your graph is not too crazy), but this is stronger.

And for the NO case, we have a nontrivial set X', and a multi-assignment where for every vertex in X', we give it t possible assignments.

The statement here is a bit confusing, so let's digest what it means. We have our graph (X, E) and a bunch of vertices and edges. More informally, the goal of this problem is to find X' which is quite a sizeable set, and to satisfy all the constraints inside it. In the YES case, we're guaranteed that there is a very large X' such that we can give one assignment to each vertex and satisfy everything. And in the NO case, we have something stronger — even if you allow me to give t assignments to each vertex, I'm *still* going to fail. (If you allow me to choose lots of assignments for each vertex — e.g., $t = \sigma$ — then this is trivial (the NO case never happens). But if t is small, then this starts being hard. (You should think of t as fixed, e.g., 10.))

> **Remark 18.14.** Formally, the condition in the NO case says that $A(u) \cap \varphi_{uv}(A(v)) = \emptyset$.

> **Remark 18.15.** There are several things you could write in the NO case — for example, instead of the t business you could say there are lots of edges inside X'.

Theorem 18.16

For every $t \in \mathbb{N}$ and $\eta > 0$, there exists $k \in \mathbb{N}$ such that the unique games conjecture implies $\textsf{gap-StrongishUG}_t[1-\eta, \eta]$ is $\textsf{NP}$-hard with $|\Sigma| = k$.

(Here $\textsf{gap-StrongishUG}_t[1-\eta, \eta]$ is the problem we defined above, with parameters t and η.)

So if you take the original version of UGC, you can prove that this is also hard (where the alphabet size depends on t and η).

> **Remark 18.17.** Always when you work with unique games, you want the alphabet size to be constant.

The proof of this is not very hard and not very interesting, so we will skip it.

§18.5 The reduction

Fix $\varepsilon > 0$ and choose η sufficiently small and t sufficiently large. (We're not actually going to spell out what you need to choose, but if you look at the proof it's not hard to see.)

We're going to do a reduction from strongish unique games, i.e., $\textsf{gap-StrongishUG}_t[1-\eta, \eta]$. This means we're given an instance $\psi = ((X, E), \Sigma, \{\varphi_e\})$. The reduction should output a weighted graph $H = (V, E', w)$ satisfying two properties:

- If there exists some large $X' \subseteq X$ and $A\colon X' \to \Sigma$ as in the YES case, then there is a large independent set in H, i.e., $\textsf{IS}(H) \geq (\frac{1}{2} - \varepsilon)\, |V(H)|$.
- If ψ is as in the NO case, then $\textsf{IS}(H) \leq \varepsilon\, |V(H)|$.

So these are the two features we want of our reduction (of course, it should also be polynomial-time).

Let's imagine drawing out the graph (X, E) corresponding to ψ. Then we want to produce a graph H. For the sake of simplicity, we'll write down what happens to a specific edge uv, but of course you do the same for each edge.

We're going to replace each vertex with a cloud — specifically, we're going to replace each vertex with a copy of the Kneser graph. So this means u and v get replaced with $\{u\} \times \mathcal{P}(\Sigma)$ and $\{v\} \times \mathcal{P}(\Sigma)$. So in the cloud corresponding to u our vertices are now named (u, A), (u, B), and so on.

In Max-Cut we had a bunch of issues that we don't have here — so this will actually be conceptually simpler. An independent set in the overall graph should in particular be an independent set in the inside graph; so we want to make sure whatever's happening on the inside corresponds to a dictatorship. So we put a Kneser graph inside the cloud for u — we draw an edge between (u, A) and (u, B) if $A \cap B = \emptyset$.

So the graph H has vertex set $X \times P(\Sigma)$ and weights $w(x, A) = \mu_p(A)/|X|$. There's two types of edges. One is the inside edges — we draw an edge $((u, A), (u, B))$ whenever $A \cap B = \emptyset$ (this is just the vanilla Kneser graph on the inside). But this of course can't work, since we haven't addressed the constraints — we've made sure that inside each cloud you look like a dictatorship, but we need to enforce the constraints, and there's no interactions between them.

So we want to add edges between the u-cloud and v-cloud that represent the constraints. Our constraint φ_e is a permutation, so it corresponds to an isomorphism of the two graphs corresponding to u and v; and so we can put edges across. So we also draw edges $((u, A), (v, B))$ — u has to be assigned some label which is supposed to be in A, and similarly v is supposed to be assigned a label in B. So whenever these *don't* intersect, we draw an edge. This means

$$E' = \{(u, A), (u, B) \mid u \in X, A \cap B = \emptyset\} \cup \{((u, A), (v, B)) \mid (u, v) \in E, \varphi_{uv}(A) \cap B = \emptyset\}.$$

How we should read this is we have A in u, and if we look at the isomorphism between our two hypercubes, $\varphi_{uv}(A)$ is the set in the hypercube of v that corresponds to it. So this edge is the one between the hypercubes corresponding to the inside edge.

The point is that if the dictators of u and v match according to φ, then there won't be any edges between their corresponding sets; otherwise there'll be a lot of edges.

§18.6 Proof of completeness

Now we're going to prove that the reduction is correct. We'll first prove completeness (that in the YES case this graph has a large independent set) — suppose $X' \subseteq X$ and $H \colon X' \to \Sigma$ are as in the YES case, meaning that all constraints inside X' are satisfied. Now we need to take this thing and turn it into an independent set. To do this, for each vertex in X', we take its assignment and do the corresponding dictatorship — so we define

$$\mathcal{I} = \{(u, A) \mid u \in X',\, H(u) \in A\}$$

(we're renaming the assignment map from A to H so that we can use A for sets).

We have to argue that this is indeed an independent set, and that this is large. To see it's large, we have

$$w(\mathcal{I}) = \sum_{u \in X'} \sum_{A \in \mathcal{P}([n])} \frac{\mu_p(A)}{|X|} 1_{u \in X'} 1_{H(u) \in A}.$$

We can calculate that this is

$$\sum_{u \in X'} \frac{1}{|X|} \sum_{A} \mu_p(A) 1_{H(u) \in A}.$$

The inner sum is just the weight of a dictatorship in the Kneser graph (we're taking all sets A containing our dictator $H(u)$), so this is just p. And so we get

$$p \cdot \frac{|X'|}{|X|} \geq p(1 - \eta) \geq \frac{1}{2} - 2\varepsilon.$$

So this is a very sizeable set.

Claim 18.18 — $\mathcal{I}$ is an independent set.

Proof. Let's look at the edges. First, there are no edges of the first type (i.e., $((u, A), (u, B))$) — if we have two vertices (u, A) and (u, B) inside $\mathcal{I}$, then by the definition of $\mathcal{I}$, $H(u)$ is supposed to be in both A and B. So they are not disjoint, and there is no edge.

For the second type of edges, for (u, A) and (v, B), we know that $u, v \in X'$ and $H(u) \in A$ and $H(v) \in B$ (by the definition of $\mathcal{I}$). Now let's look at the edges — edges happen when after we apply the permutation on A, there's no intersection. But we know $H(u)$ is in A, so $\varphi_{uv}(H(u))$ is inside $\varphi_{uv}(A)$. But because the constraint φ_{uv} is satisfied (both u and v are in X'), this is the same as $H(v)$. So then $H(v)$ is in both $\varphi_{uv}(A)$ and B, which means they aren't disjoint and there is no edge. □

So we're done with completeness — we proved that if you start with a good strongish unique games instance, you get a graph with a very large independent set.

§18.7 Proof of soundness

Now we'll move to soundness — so we want to prove that if ψ has no good assignment (as in the NO case), then H doesn't have even a slightly large independent set. As we did last time (and is often the case), we'll do this in reverse — we'll show that if H does contain a good independent set, then we can find a good assignment to ψ.

So suppose that $\mathcal{I} \subseteq H$ is an independent set of weight $w(\mathcal{I}) \geq \varepsilon$.

Remark 18.19. When Dor was thinking about how to present this proof, he was aiming for a cool-points factor in that we'd just use the thing we saw for intersecting families and be done with it. This can be done, but it's more complicated; so we'll not do this.

Let's begin with one observation — using upper shadows (as before).

Definition 18.20. We define $\mathcal{I}\uparrow = \{(u, B) \mid \text{exists } A \subseteq B \text{ with } (u, A) \in \mathcal{I}\}$.

Then $\mathcal{I}\uparrow$ is still an independent set (for the same reason as what we saw when talking about intersecting families). And now $\mathcal{I}\uparrow$ is closed upwards. To reduce notation, we'll assume $\mathcal{I}$ originally was closed upwards, i.e., $\mathcal{I} = \mathcal{I}\uparrow$ (otherwise we'd have to write $\mathcal{I}\uparrow$ instead of $\mathcal{I}$ all over the place, which would be annoying).

We want to go from independent sets to labels. So first of all, we have to identify the part of $\mathcal{I}$ that comes from each u. This is just notation — for each $u \in X$, we define

$$\mathcal{I}_u = \{A \subseteq \Sigma \mid (u, A) \in \mathcal{I}\}.$$

Claim 18.21 — If we sample $u \in X$ uniformly, then $\mathbb{E}_{u \in X} \mu_p(\mathcal{I}_u) \geq \varepsilon$.

Proof. If we spell out this expectation, we get

$$\mathbb{E}_{u \in X} \mu_p(\mathcal{I}_u) = \sum_{u \in X} \frac{1}{|X|} \sum_{A \subseteq \Sigma} \mu_p(A) 1_{(u,A) \in \mathcal{I}},$$

and once we collect everything together this is exactly

$$\sum_{(u,A) \in \mathcal{I}} \frac{\mu_p(A)}{|X|} = w(\mathcal{I}).$$ □

So this claim says that the average weight of $\mathcal{I}_u$ over u is at least ε; by an averaging argument, this means it's at least $\varepsilon/2$ for a sizeable number of u's. Let

$$X'' = \{u \mid \mu_p(\mathcal{I}_u) \geq \varepsilon/2\}$$

be the set of u's for which $\mathcal{I}_u$ is somewhat sizeable. Then by the claim and an averaging argument, we know that

$$\left|X''\right| \geq \frac{\varepsilon}{2}\left|X\right|.$$

Now we've identified a large set of u's where we have a large independent set inside u, and we're aiming for a junta-type thing. So we'll do another argument we saw with intersecting families — we define the function $f\colon [p, p + \frac{\varepsilon}{2}] \to [0, 1]$ which just measures the expected measure of I_u when we slightly change p — so

$$f(q) = \sum_{u \in X} \mu_q(\mathcal{I}_u).$$

Then by Lagrange, there exists q for which

$$f'(q) = \frac{f(p + \varepsilon/2) - f(p)}{(p + \varepsilon/2) - p} \leq \frac{1 - 0}{(p + \varepsilon/2) - p} = \frac{2}{\varepsilon}.$$

But if you calculate this derivative, if we just had one family then it'd be the derivative of $\mu_p(\mathcal{I}_u)$; and by the linearity of derivatives, this is the same as

$$f'(q) = \mathbb{E}_{u \in X} \frac{d\mu_q}{dq}(\mathcal{I}_u).$$

So this is telling us that the average value of the derivative of these $\mathcal{I}_u$'s is at most $2/\varepsilon$, which means by Markov that for almost all of them, it's not much more than $2/\varepsilon$. Let

$$X_2 = \left\{u \in X \mid \frac{d\mu_q}{dq}(\mathcal{I}_u) \leq \frac{8}{\varepsilon^2}\right\}.$$

Then by Markov's inequality

$$\left|X_2\right| \geq \left(1 - \frac{\varepsilon}{4}\right)\left|X\right|.$$

(We've secretly already used the upper shadow thing — because we did the upper closure, each $\mathcal{I}_u$ is monotone, which means its derivatve is nonnegative — we need this for Markov.)

And now we take $X' = X'' \cap X_2$ to consist of everything which is sizeable and has small derivative; these two results give

$$\left|X'\right| \geq \frac{\varepsilon}{2}\left|X\right| - \frac{\varepsilon}{4}\left|X\right| = \frac{\varepsilon}{4}\left|X\right|.$$

Now for each vertex $x \in X'$, we know two things — that

$$\mu_q(\mathcal{I}_u) \geq \mu_p(\mathcal{I}_u) \geq \frac{\varepsilon}{2},$$

and that

$$\frac{d\mu_q}{dq}(\mathcal{I}_u) \leq \frac{8}{\varepsilon^2}.$$

Recall that for monotone families, this derivative thing has a name — it's the influence $I(1_{\mathcal{I}_u})$. And so we get that for each u, the total influence is pretty small.

And what we know about families with small total influence is that they are close to juntas — so $\mathcal{I}_u$ must be close to a $2^{O(1/\varepsilon^2)}$ junta (this is Friedgut's theorem).

So far everything is correct, but here's a lie — this closeness is really annoying and to bypass it you need to do some hacks. So we'll lie and say that $\mathcal{I}_u$ really *is* a t-junta (where $t = 2^{O(1/\varepsilon^2)}$), meaning that there exists a set of labels $J_u \subseteq \Sigma$ and possible assignments $\mathcal{J}_u \subseteq \mathcal{P}(\mathcal{J}_u)$ such that $\mathcal{I}_u = \{A \subseteq \Sigma \mid A \cap J_u \in \mathcal{J}_u\}$.

Now we have X', which is all these nice vertices; and for each one, we have a junta. This is actually the t-assignment that we're going to give. So we define $H\colon X' \to \binom{\Sigma}{t}$ by $H(u) = J_u$. Next time we'll show that this asignment actually satisfies all the constraints in X'. (The proof is two lines, and that's really where things end.)

§19 April 18, 2024

§19.1 Soundness of vertex cover reduction

Last time, we saw a polynomial time reduction that takes an instance ψ of (strongish) unique games and produces a graph H, and we proved that for $p = \frac{1}{2} - \varepsilon$, if $\operatorname{val}(\psi) \geq 1 - \eta$ then H has a large independent set — $\mathsf{IS}(H) \geq \frac{1}{2} - 2\varepsilon$. Now we want to prove that if ψ has no (η, t)-assignment (as in the strongish unique games setup), then H *doesn't* have an independent set of even size ε.

We started doing this last time — we actually went nearly all the way. We did this contrapositively. Recall that if $\psi = ((X, E), \Sigma, \{\varphi_e\})$, then the vertices of H were $V(H) = X \times \mathcal{P}(\Sigma)$.

So far, we've showed that if we have an independent set $\mathcal{I} \subseteq V(H)$ of density at least ε, then we can find a subset $X' \subseteq X$ which is sizeable — $\lvert X' \rvert \geq \frac{\varepsilon}{4}\lvert X \rvert$ — such that for every vertex $u \in X'$, looking at the subset $\mathcal{I}_u$ of the hypercube defined as

$$\mathcal{I}_u = \{A \subseteq \Sigma \mid (u, A) \in \mathcal{I}\},$$

this set satisfies two properties:

- $\mu_q(\mathcal{I}_u) \geq \varepsilon/2$, and
- $\mathcal{I}_u$ is a J_u-junta for some $\lvert J_u \rvert \leq t$.

These are the things we proved; where $q \in (p, p + \varepsilon/2)$.

Now we're going to finish up the proof and show that actually the assignment that gives each u this junta is an (η, t)-assignment.

So we define the assignment $H\colon X' \to \binom{\Sigma}{t}$ by $H(u) = J_u$.

> **Claim 19.1** — If we have an edge $(u, v) \in E$ such that $u, v \in X'$, then $\varphi_{uv}(H(u)) \cap H(v) \neq \emptyset$.

In human language, if we look at the labels we gave to u and to v, then there is a pair satisfying the constraint on our edge. If we show this, then we're done (because then H is an (η, t)-satisfying assignment for $\eta = \varepsilon/4$).

Proof. Assume otherwise. For simplicity of notation (and without loss of generality) we can assume φ_{uv} is the identity (otherwise we could just relabel one of the cubes). So our assumption says that $H(u)$ is disjoint from $H(v)$.

But $\mathcal{I}_u$ is nonempty and is a J_u-junta, so we can choose some subset $B_u \subseteq J_u$ such that $A \in \mathcal{I}_u$ for every $A \cap J_u = B_u$. (This is because $\mathcal{I}_u$ is nonempty, so we can find something in it; and because it's a J_u-junta, any set giving us the same intersection with J_u will still be in $\mathcal{I}_u$.) We can do the same for v, so we can find $B_v \subseteq J_v$ such that $A \in \mathcal{I}_v$ for every $A \cap J_v = B_v$.

Now let's think about this for a moment. We have our variables Σ, and we have J_u and J_v, which have no overlap by our assumption (that $H(u)$ and $H(v)$ are disjoint).

But all of this implies that there exist $A \in \mathcal{I}_u$ and $A' \in \mathcal{I}_v$ that are disjoint — in fact, we can take $A = B_u$ and $A' = B_v$.

Remark 19.2. Last lecture, we said we are lying — you're not actually a J_u-junta but actually close to it. If you are just close, then this part of the argument gets messier; but we won't go into this.

So we've found A and A' that are disjoint, but both are inside our independent set; and this means (u, A) and (v, A') are both in $\mathcal{I}$, but $\varphi_{uv}(A) \cap A' = \emptyset$, so $\mathcal{I}$ is not an independent set (because this condition means there is an edge between them). So this is a contradiction. □

§19.2 Boolean functions beyond the hypercube

Now we'll move on to the main topic for today, Boolean functions beyond just the cube.

§19.2.1 The ternary cube

Let's start with a concrete example — the *ternary cube* $\{0,1,2\}^n$. So we'll consider functions $f\colon \{0,1,2\}^n \to \{0,1\}$ (or $\{0,1,2\}$).

Depending on who you are, you may either say this is a very nice domain and let's study it; or if you're more of a computer scientist, you may say, why should I care about this? Dor is more of a computer scientist, so he will try to answer that question.

Here's one example where the ternary cube naturally comes up, related to what we've done in the last two lectures. It turns out if you want to prove hardness results for 3-coloring, this is what you want to use.

In the last two or three lectures, we saw a paradigm of proving hardness of approximation; this involved setting up a concrete graph, typically on the hypercube (here on the ternary cube), and then designing a graph from it such that dictators are good solutions, while anything that's not a dictator at all is a very bad solution.

If we look at a dictator $f(x) = x_i$, this is a 3-coloring of *any* graph $G = (\{0,1,2\}^n, E)$. We want to design a graph such that these dictatorships are valid colorings, but anything that doesn't look like a dictatorship is *really* not a coloring.

We'll connect x and y whenever they disagree on all coordinates — so

$$E = \{(x,y) \mid x_i \neq y_i \text{ for all } i\}.$$

Then virtually by design, each dictatorship is a valid 3-coloring. And it turns out the following theorem is true.

If we can color the vertices using 10 colors, it means we've partitioned the vertices into ten independent sets, so one has size at least $\frac{1}{10}$. So a coloring with a few colors gives a large independent set. And we'll say that even coming up with a large independent set in this graph requires an influential variable.

Theorem 19.3

For every $\delta > 0$, there exists $d, \tau > 0$ such that if $S \subseteq \{0,1,2\}^n$ with $|S| \geq \delta \cdot 3^n$ is an independent set, then there exists $i \in [n]$ such that $I_i^{\leq d}[1_S] \geq \tau$.

This is a nice theorem, even though strictly speaking it's not well-defined, in the sense that we never defined what are influences over this domain (let alone low-degree influences). But we will do this soon, or at least say how it's possible.

Still, it's not clear how to prove something like this.

§19.2.2 Product spaces

It actually turns out that the moment you step outside the Boolean cube, it doesn't really make sense to tolerate just this one domain; there's a more general family of domains called *product spaces*, in which you can get lots of the theory we've been doing so far. And the ternary cube is just one example.

Definition 19.4. Given spaces $(\Omega_1, \mu_1), \ldots, (\Omega_n, \mu_n)$, we define their *product space* as $(\Omega, \mu) = (\Omega_1 \times \cdots \times \Omega_n, \mu_1 \times \cdots \times \mu_n)$.

So we take the product of sets and the product measure.

§19.3 Efron–Stein decomposition

Whenever we're in this setting, there's something kind of analogous to the Fourier decomposition that you can do — it's less explicit, but often as useful.

Definition 19.5. For a set $S \subseteq [n]$, we say that a function $f\colon (\Omega, \mu) \to \mathbb{R}$ is an *S-junta* if there exists a function $g\colon \prod_{i \in S} \Omega_i \to \mathbb{R}$ such that $f(x) = g(x_S)$ for all $x \in \Omega$.

(In other words, f is a S-junta if it only depends on the coordinates in S.)

Definition 19.6. We define the space $V^{\subseteq S} \subseteq L_2(\Omega, \mu)$ as the span of all S-juntas.

Now that any function that can be expressed as a combination of S-juntas is still a junta, but we're writing it this way to make it clear the space is linear.

Now we're going to define an inner product, the obvious one.

Definition 19.7. For two functions $f, g\colon (\Omega, \mu) \to \mathbb{R}$, we define $\langle f, g \rangle = \mathbb{E}_{x \sim \mu} f(x) g(x)$.

Remark 19.8. Typically we'll think of the Ω_i's as small (and finite) sets.

Definition 19.9. We define $V^{=s} = V^{\subseteq S} \cap \bigcap_{T \subsetneq S} (V^{\subseteq T})^{\perp}$.

In other words, we're looking at everything that's a S-junta, but is orthogonal to everything that's a smaller junta.

Remark 19.10. This is a bunch of definitions; to make sense of them, we're encouraged to think about where each (Ω_i, μ_i) is $\{0, 1\}$ with the uniform measure. Then we're going to get that $V^{=S} = \operatorname{Span}\{\chi_S\}$. (We know χ_S is orthogonal to any character other than itself; and it's also a S-junta.) So this is a good sign.

And once we have all these spaces, we can write

$$L_2(\Omega, \mu) = \bigoplus_{S \subseteq [n]} V^{=S}.$$

This in particular tells you that for every $f\colon (\Omega, \mu) \to \mathbb{R}$, there is a unique representation of f in the form

$$f(x) = \sum_{S \subseteq [n]} f^{=S}(x)$$

where $f^{=S} \in V^{=S}$. In the hypercube case, $f^{=S}(x) = \widehat{f}(S)\chi_S(x)$.

You can see this more general decomposition coincides with the Fourier decomposition in the Boolean case; in more general product spaces, it's called the *Efron–Stein decomposition*.

§19.4 More definitions

And then we can generalize a bunch of things we did in the course.

Definition 19.11. We define $\deg(f) = \max_{f^{=S} \neq 0} |S|$.

Definition 19.12. We define $I_i[f] = \mathbb{E}_{x,y\sim\mu}[|f(x) - f(y)|^2 \mid x_{-i} = y_{-i}]$.

So we're sampling x and y according to μ, conditioned on them being equal everywhere except on the ith coordinate. If you play around with this definition, you get

$$I_i[f] = \sum_{S \ni i} \left\| f^{=S} \right\|_2^2,$$

which matches the influences in the hypercube. So this answers what are influences and low-degree influences on the ternary cube (the low-degree influences are where you take the low-degree part of f).

§19.5 Extending the theory

You have an orthogonal decomposition, so you still get Parseval and Plancherel and all that. Next, you can wonder what happens to hypercontractivity and KKL and all that stuff.

Here, not all product spaces are made the same; some are different than others. Roughly speaking, as long as your product spaces are not too 'wide' or 'unbalanced', nearly all the theory extends.

Theorem 19.13 (Hypercontractivity)

If for every $i \in [n]$ and every a such that $\mu_i(a) \neq 0$ it holds that $\mu_i(a) \geq \alpha$ (where α is some constant), then

$$\|f\|_q \leq \sqrt{q/\alpha}^{\,d} \, \|f\|_2$$

for all f of degree at most d.

Usually in words we say the probability of each atom in μ_i is at least a constant. And as long as this happens, you have hypercontractivity and everything goes through.

In the Boolean cube $\alpha = \frac{1}{2}$, and here you get $2q$ (you can improve this if you want). And then you get all of KKL and Friedgut and the invariance principle and so on.

In the ternary cube it's the same except $\alpha = \frac{1}{3}$, and nothing changes.

So for constant α, all of the theory extends (KKL, Friedgut, the invariance principle, and so on). Some of these things are kind of direct (KKL and Friedgut); some require more thought. The invariance principle does require some thought or notation, but once you get this the proof is the same.

And the proof of the theorem we stated for the ternary cube is via the invariance principle. You imagine the edge set as a sort of operator (like the noise operator in Majority is Stablest) and try to move to the Gaussian setting, and there are no independent sets in that setting.

(This lecture is going to be very high-level; we're giving an overview of what exists beyond the Boolean cube. Next lectures we'll focus on some concrete things and prove stuff.)

Remark 19.14. In the $\{0,1,2\}$-case, what is $\dim V^{=S}$? It should be $2^{|S|}$. For the ternary cube, you can get an explicit basis — for ω a third root of unity, you get a basis $\omega^{\langle v,x\rangle}$ for $v \in \{0,1,2\}^n$. (Then $V^{=S}$ is spanned by v with support S; you have $2^{|S|}$ such things. And if you sum all the dimensions with the appropriate binomial coefficients you get 3^n, which is a good sign.)

§19.6 Non-balanced product spaces

This is a very important set of spaces — people use them a lot (whenever you see UGC and some reduction from it, some space like this is used). But not all spaces have constant α. Here's a concrete example of the simplest space that is a product space, but doesn't fall under this umbrella.

Question 19.15. What if μ_i has atoms with small probability?

More concretely, consider the space $(\Omega_i, \mu_i) = (\{0,1\}, \mu_p)$ for e.g. $p = 1/\sqrt{n}$. We can still plug in $\alpha = 1/\sqrt{n}$ and get some sort of hypercontractivity result; but when your hypercontractive inequality depends on the dimension, bad things can and will happen, and the proofs of all the theorems collapse. And this is for a good reason — the theorems are just false.

For example, Friedgut is false. Let's see an example of a function with small total influence which is clearly not a junta — define

$$f(x) = \bigvee_{i=1}^{n/2} (x_{2i} \wedge x_{2i-1}).$$

This is a very simple function — a DNF formula of width 2. You can check that $\mathrm{Var}[f] = \Theta(1)$ (the probability a given clause is 1 is $p^2 = 1/n$, and we have linearly many of them, so the probability that one of them is 1 is constant). If you compute the total influence, you see that $I[f]$ is also constant (at most 2). (It's a general fact that the total influence of a DNF formula is at most its width, but here you can also compute it exactly if you want.) But thius is clearly not a junta — if you take constantly many variables, you're not going to see anything.

So this fails. When you think about it, the real reason is that the hypercontractivity inequality is just wrong without the α. The most convincing way to see this is that if you look at the noisy version of this graph, it is not a small-set expander.

Consider the p-biased noisy hypercube with parameter $\rho \in (0,1)$ (this doesn't really matter; you can think about it as $1/2$ or close to 1). For each $x \in \{0,1\}^n$, we sample $y \sim T_\rho x$ as follows: independently for each coordinate $i \in [n]$, we set $y_i = x_i$ with probability ρ, and otherwise we resample it from μ_p. This is really the straightforward generalization of the noise operator we saw for the cube; this also defines a graph (or Markov chain) on the cube.

We claim that this graph is not a small-set expander — there are sets of very small measure that have expansion bounded away from 1. It's actually quite easy to construct such sets — if you look at

$$A = \{x \mid x_1 = 1\},$$

then $\mu(A) = p = 1/\sqrt{n}$, which goes to 0. But if we look at its expansion, we get

$$\mathbb{P}_{x \in A, y \sim T_\rho x}[y \notin A] \leq 1 - \rho.$$

(This is because whenever we pick $y_1 = x_1$ we stay inside the set.)

So the point of all this is to say that when we move from α constant to subconstant, it's not just that we don't know how to prove things, but that things are completely different (Friedgut is false, and you don't have small-set expansion), so it's not clear what to do.

This case was less studied until recently — at least, from the TCS point of view — because it's mainly related to sharp thresholds in random graph theory and statistical physics, but there were no applications in TCS. People did work on this (e.g., Friedgut and Bourgain and a bunch of other people — we may mention some results next time), but it's really hard to work here.

§19.7 Non-product spaces

In the grand scheme of things, any product space either has constant or subconstant probability for atoms, so one of the two sorts of theories work. But there are other spaces — spaces that are not product spaces, that are also interesting.

Here there's many examples; we'll mention two related ones. One is the *Johnson scheme* $\binom{[n]}{k}$; and the other is the *symmetric group* S_n.

§19.7.1 The Johnson scheme

The Johnson scheme is the set $\binom{[n]}{k}$ — which we can think of as the set of strings in the hypercube with Hamming weight k, i.e.,

$$\binom{[n]}{k} = \{x \in \{0,1\}^n \mid |x| = k\}.$$

Then we want to study $f\colon \binom{[n]}{k} \to \mathbb{R}$. This makes sense for lots of k's — if we look at $k = n/2$ this roughly looks like the hypercube we've been studying so far (because a typical point in the hypercube has roughly $n/2$ 1's). So how different can they be?

It turns out they are different — here's a problem that's not hard on the hypercube, but is on the Johnson scheme. Let's imagine we sample three points from the Johnson scheme, condition on their sum also being in the Johnson scheme, and suppose that $f\colon \binom{[n]}{n/2} \to \{0,1\}$ satisfies that whenever we have four such points, we have

$$f(x+y+z) = f(x) + f(y) + f(z).$$

This is the type of thing we've seen at the beginning of the course (it even appeared in a problem set). More specifically, what can we say about such a function if

$$\mathbb{P}_{x,y,z\in\binom{[n]}{n/2}}[f(x+y+z) = f(x) + f(y) + f(z)] \geq \frac{1}{2}?$$

In the hypercube, this is just a Fourier analytic computation where you do some stuff, get fourth powers, and are done. Here that's not the case; we do know how to solve it, but it requires more ideas.

§19.8 Connection to representation theory

How do you analyze such functions and get decompositions? You can do ad hoc things and that gets you some mileage, but it doesn't feel like the right thing to do. It turns out the right thing to do is representation theory.

For this, we're going to switch for a moment to S_n. Suppose we look at the functions $f\colon S_n \to \mathbb{R}$; we'd like to define a nice basis like the Fourier basis or Efron–Stein decomposition.

In the Fourier decomposition, what you do is you find homomorphisms $\chi\colon S_n \to \mathbb{C}$. (In our case, it was just ±1, but in general it's $\mathbb{C}$.) By a *homomorphism* we mean $\chi(a \cdot b) = \chi(a)\chi(b)$. So you can try to find homomorphisms, and if you have enough of them then you can actually use them as a basis. This is really what we did with the Boolean cube.

So we have S_n, and we can try to find homomorphisms in this case too; let's try to define some. We have one homomorphism for free — $\chi_{\text{triv}}\colon S_n \to \mathbb{R}$ sending $\chi(\pi) = 1$ for all $\pi \in S_n$. This is not a very interesting homomorphism, but it is one nonetheless. And there is another one — $\chi_{\text{sgn}}\colon S_n \to \{-1, 1\}$ defined as $\chi_{\text{sgn}}(\pi) = \text{sgn}(\pi)$ (you can check that this is also multiplicative).

So let's see — the dimension fo the span of functions $S_n \to \mathbb{R}$ is $n!$, and we've found two functions; so this is not a basis. And it's going to be hard to find another function like this, because there are none. It's a fact of life that when you work with domains that are non-abelian groups, you're not going to find a basis composed of homomorphisms like this. (With abelian groups you can, but nonabelian groups don't work.)

Instead, you have to find *representations*. A representation is in a sense another homomorphism, but not into $\mathbb{C}$.

> **Definition 19.16.** A *representation* ρ of S_n is a homomorphism $\rho\colon S_n \to \text{GL}(V)$ for some linear space V. We call $\dim V$ the *dimension* of the representation ρ.

So we're sending each permutation to a linear operation on some space V; a homomorphism just means that $\rho(\pi)\rho(\tau) = \rho(\pi\tau)$. This is a generalization of one-dimensional homomorphisms.

It turns out that once you enlarge your view to representations, you can find a bunch more representations for S_n. But there is a notion of *reducibility* vs. *irreducibility*.

§19.8.1 Reducibility

Suppose we have a representation $\rho\colon S_n \to \text{GL}(V)$. If we can decompose $V = U \oplus W$ where U and W are invariant spaces of each one of the operations $\rho(\pi)$, then this means we can decompose each $\rho(\pi)$ into an operation on U and one on W — so we can get smaller-dimensional representations $\rho_U\colon S_n \to \text{GL}(U)$ and $\rho_W\colon S_n \to \text{GL}(W)$. In that case, we say ρ is *reducible* — because we can salvage smaller things from it. Otherwise, we say that ρ is *irreducible*.

This is not a course on representation theory, so we are not going to say much more about this. But essentially, irreducible representations of S_n are well-known, and if you look at all of them and take their span, you get a basis for the complex-valued functions on S_n. And they're pretty nice — not as nice as the Fourier characters, but almost as nice. And you can extend a lot of the theory we know to this, but it gets more complicated.

So the moral is that irreducible representations are the basic building blocks of all the orthogonal decompositions; and this is how you build theory for S_n.

§19.8.2 Back to the Johnson scheme

We started looking at the Johnson scheme and then moved to representations and functions over S_n, but how does this apply to the Johnson scheme? It turns out that if you have a domain which has a group action (S_n acts on vectors of length n), then you automatically get something. The point is as follows — suppose we have a function $f\colon \binom{[n]}{k} \to \mathbb{R}$. Then we can define $V = L_2(\binom{[n]}{k})$ (this is going to be our V in $\text{GL}(V)$), and we can define $\rho\colon S_n \to \text{GL}(V)$ in the following way — for every $\pi \in S_n$, we define $\rho(\pi)\colon V \to V$ by $\rho(\pi)f(x) = f(\pi x)$. (This is somewhat bad notation, but what we're saying is for each π, you can consider the action on functions where you permute the coordinates of x according to π. This gives you some map $V \to V$, and in fact this is a linear map, and you can check that it is also multiplicative — $\rho(\pi)\rho(\tau) = \rho(\pi\tau)$, since if we first act with τ and then π, then overall we're acting with $\tau\pi$.)

> **Remark 19.17.** Note that $\rho_\pi f$ is a function $\binom{[n]}{k} \to \mathbb{R}$, and the way we define this function is that $\rho(\pi)f(x) = f(\pi x)$.

So for each permutation π, we can give it an action on functions; and this action is actually a representation. So we've got a representation of S_n (not the Johnson scheme or anything else). Once we have that, we can apply all the nice theory and decompose ρ into irreducible representations. And when you do that, you'll be decomposing V into a direct sum of some nice-looking spaces, which is actually how the orthogonal decomposition works for the Johnson scheme.

To summarize, the moral of all this is that whenever you have a non-product space, sometimes it looks like a product space and you can do stuff by ad-hoc means, but otherwise if you have some algebraic structure (with a group acting on you), then you can try to appeal to algebra. This is a very powerful approach, and it's fairly recent.

§19.9 The Grassman graph

Now we'll define one more space (or two spaces), which is another space we often want to do analysis on.

Definition 19.18. Given parameters ℓ and n with $0 < \ell \leq n-1$, the *Grassman graph* $\mathrm{Grr}_{\mathbb{F}_2}(n, \ell)$ is defined as the graph whose vertices are all ℓ-dimensional subspaces of $\mathbb{F}_2^n$, i.e.,

$$\{L \subseteq \mathbb{F}_2^n \mid \dim(L) = \ell\}$$

(we denote this as $\begin{bmatrix}\mathbb{F}_2^n \\ \ell\end{bmatrix}$) and whose edges are

$$\{(L, L') \mid \dim(L \cap L') = \ell - 1\}.$$

If we make an analogy to the Johnson scheme, in the Johnson scheme we took all subsets of $[n]$ of a fixed size. This is similar — we're taking subsets of size 2^ℓ — but now they also have to be subspaces. There's an analogy between them, and lots of features actually carry over, but this is more complicated.

You can ask many questions about this graph.

Question 19.19. Can you get an orthogonal decomposition of functions $f\colon \begin{bmatrix}\mathbb{F}_2^n \\ \ell\end{bmatrix} \to \mathbb{R}$?

The answer is yes, and there's two ways to do it (a brute-force way and a group-theoretic way).

Question 19.20. What happens to KKL or hypercontractivity or small-set expansion?

This is where the mess starts — basically none of these things work (like the p-biased cube, or the Johnson scheme for some k). As long as $0 \ll \ell \ll n$, these things fail. So that's annoying; in the next lecture we'll see some examples of sets that are non-expanding.

Why do people care about this? Historically, harder things were studied before easier things were — much of the development for product spaces was directly motivated by the unique games reduction we saw, so that theory expanded itself. But then people didn't really care about these things — if you researched them they'd be nice, but people in TCS wouldn't understand why you're doing that. This was true until this graph or domain played a significant role in some PCP stuff — Dor will tell us the statement of what was needed next time. That basically opened a new type of analysis for these domains, which turned out to be important in PCPs. And later on people realized interesting things happen here and extended them to the other spaces we've discussed; and these things have lots of applications (some of which we will mention).

In the next two lectures, Dor will tell us the statement but not prove it; instead we'll prove a baby version of that statement for something easier, which will hopefully give some intuition.

§20 April 23, 2024

Last time we discussed a bit about the extensions of analysis of Boolean functions to other domains. We sort of divided domains into two types — those where everything works, and those where nothing works. Today we're going to speak about the latter type.

§20.1 The p-biased cube and graph properties

The p-biased cube with $p = o(1)$ has historically been of interest in random graph theory — suppose we sample a graph $G \sim \mathcal{G}(n, p)$, and ask a question such as 'is G connected?' This is completely equivalent to looking at the domain $\{0,1\}^{\binom{n}{2}}$ and considering the Boolean function $f\colon \{0,1\}^{\binom{n}{2}} \to \{0,1\}$ where $f(x) = 1$ if and only if the graph described by x is connected.

When you look at these types of properties, they're monotone; so you can graph the probability that $f(x) = 1$ as a function of p. It's a well-known theorem that you have a sharp threshold — it goes from 0 to 1 very quickly — and this happens at $p \sim \frac{\log n}{n}$, where n is the number of vertices.

So here the interesting thing happens at p which is small. This is simple enough that you can do it with your bare hands, and don't need fancy machinery. But there are other properties that are also kind of like graph properties (or often *hypergraph properties*) that are much less trivial.

Definition 20.1. A *k-CNF formula* is a formula of the form $\varphi(x_1, \ldots, x_n) = C_1 \wedge \cdots \wedge C_m$ where each clause C_i is an OR of k variables or their negations.

(This is the same thing as 3SAT with 3 replaced by k.)

Example 20.2

There are $N = 2^k\binom{n}{k}$ possible clauses on k variables (the factor of 2^k comes from negating some of the variables).

Definition 20.3. A *random k-CNF* with density p is one where each possible clause appears with probability p.

The question we're interested in about CNF formulas is whether they're satisfiable.

Definition 20.4. We say φ is satisfiable if there exists an assignment $A\colon \{x_1, \ldots, x_n\} \to \{0,1\}$ where each clause evaluates to 1.

If we increase p, the probability that the formula is satisfiable will decrease. So now we can look at the satisfiability problem, which we can again phrase in terms of Boolean functions — we have a Boolean function $f\colon \{0,1\}^N \to \{0,1\}$ that takes a description of a Boolean formula and outputs 1 if it is satisfiable.

If $p = 0$ then of course the formula is satisfiable, while if p is large then it's not. But what happens in between?

It turns out that there's a sharp threshold. It's not so hard to show that it occurs at $p_k \approx c_k/n$. But the remarkable thing here is that the first paper that made a dent on this problem proved that there *exists* a sharp threshold, but they couldn't say what c_k is. A later paper figured out the constant c_k; this is very difficult.

How do you prove something like this? There's a paper by Friedgut from 1999 (it has an appendix by Bourgain). Friedgut (from the junta theorem) was interested in being able to tell when a graph property has a sharp threshold, even when p is small. He proved the following result:

Theorem 20.5 (Friedgut 1999)

If $f\colon \{0,1\}^n \to \{0,1\}$ is a monotone graph property with $p \cdot I[f] \leq k$, then for all $\varepsilon > 0$, f is ε-close to a DNF formula of width $O_{k,\varepsilon}(1)$.

Today we'll define influences slightly differently for the p-biased measure. (The condition $p \cdot I[f] \leq k$ intuitively states that f has a coarse threshold.)

Definition 20.6. A *DNF formula* is a formula of the form $D_1 \vee \cdots \vee D_m$ where each clause D_i is an AND of a bunch of variables; the *width* is the number of variables in each clause.

If you think about it, if f is a graph property that looks like a DNF, then it's something like containing a copy of a subgraph.

Example 20.7

If you write down the property of containing a clique of size 4, that's a DNF formula of size $\binom{4}{2}$ — we do an OR over all cliques of an AND of the edges inside that clique. More generally, containing a clique of size t is a width $\binom{t}{2}$ DNF formula.

Then Friedgut managed to prove that the k-SAT problem doesn't look like a DNF.

This is a very nice result, and you can use it to prove a bunch of sharp threshold results even when p is small.

In this theorem, we have two conditions. Monotonicity is necessary (otherwise there's a bunch of examples that we don't know how to characterize). But there's also the 'graph property' condition.

Conjecture 20.8 — The same theorem holds without the graph property assumption.

There was some progress towards this by Hatami, but we won't state the result (it's kind of weird).

If you're close to a DNF formula, then in particular you have all these clauses of constant size; so if we take one of them and set all the variables to make e.g. D_1 evaluate to 1, then f also should evaluate to 1. It turns out that a corollary of this, almost as good as the theorem in applications:

Corollary 20.9

If f is a monotone graph property with $pI[f] \leq k$, then there exists $S \subseteq [n]$ with $|S| \leq O_k(1)$ such that $\mu(f_{S \to 1}) \geq 0.99$.

So if f is a monotone graph property with small total influence, then there's a small set of variables such that if we fix all of them to 1, then the average of f jumps to 0.99.

Remark 20.10. The result of Hatami establishes this corollary without the graph property assumption.

So people in the sharp threshold community for graph properties are very happy with this; there's many results proved using this. But the proof of this doesn't use hypercontractivity, partly because hypercontractivity doesn't hold; it uses lots of nice work, but it's ad hoc-ish and uses symmetry very heavily, so it's not clear how to extend it to other domains or more general settings.

§20.2 2-to-1 games

When we talked about unique games a few lectures ago, there are some close siblings of this; now we'll talk about one of those close siblings, namely 2*-to-*1 *games*.

Definition 20.11. We say $\psi = (G = (L \cup R, E), \Sigma_L, \Sigma_R, \{\varphi_e\}_{e \in E})$ is an instance of 2-to-1 Games if $|\Sigma_L| = 2\,|\Sigma_R|$ and for each $e \in E$, the map $\varphi_e \colon \Sigma_L \to \Sigma_R$ is a 2-to-1 map (i.e., the pre-image of everything in Σ_R has size 2).

In unique games, we only have one alphabet; here we have two alphabets, where one is slightly larger than the other, and instead of requiring the constraints to be bijections, they're 2-to-1.

In the same paper as the unique games conjecture, the following conjecture was made:

Conjecture 20.12 (Khot) — For all $\varepsilon > 0$, there exists k such that given an instance ψ of 2-to-1 games with alphabet sizes at most k, it is NP-hard to distinguish:

- The YES case: $\text{Val}(\psi) = 1$ (i.e., we can satisfy *all* the constraints of ψ).
- The NO case: $\text{Val}(\psi) \le \varepsilon$ (i.e., we can't even satisfy an ε-fraction of the constraints).

There's a lot of history related to this, which we won't go into. In 2016 there was some attempt to prove this conjecture. It turned out that one of the components you need has to do with this lack of hypercontractivity, but on a different domain instead of the p-biased cube.

§20.2.1 The Grassman graph

Definition 20.13. The *Grassman graph* $\mathbb{G}(n, \ell)$ over $\mathbb{F}_2$ with $0 \ll \ell \ll n$ is the graph whose vertices are

$$V = \{L \mid L \le \mathbb{F}_2^n,\ \dim(L) = \ell\}$$

and whose edges are

$$E = \{(L, L') \mid \dim(L \cap L') = \ell - 1\}.$$

In 2016 we had a reduction that tried to prove the conjecture; it involved the Grassman graph. Dor worked on trying to prove this (with collaborators). They found that if this graph is a small-set expander then things would simplify. But eventually it turns out that it isn't actually a small-set expander, and then everything collapsed.

§20.2.2 Failure of small-set expansion

Now we'll see that the Grassman graph is not a small-set expander. We'll see two examples.

Example 20.14

Fix $a \in \mathbb{F}_2^n \setminus \{0\}$, and let $S_a = \{L \mid a \in L\}$.

This set has two features. First, it's has small measure — we have

$$\mu(S_a) = \frac{2^\ell - 1}{2^n - 1} \approx 2^{\ell - n} = o(1)$$

(S_a has $2^\ell - 1$ nonzero vectors, and each is a with the same probability by symmetry). But on the other hand, if we sample $L \in S_a$ and choose a random edge $(L, L') \in E$, what's the probability we stay in S_a? We have

$$\mathbb{P}_{L \in S_a, (L, L') \in E}[L' \in S_a] = \frac{2^{\ell-1} - 1}{2^\ell - 1} \approx \frac{1}{2}$$

(since L and L' have $2^{\ell-1}-1$ nonzero vectors in common; and each nonzero thing in L' has equal probability of being in there). This is a violation of small-set expansion (because this probability is bounded away from 0).

So this is very sad. The second example is even sadder.

Example 20.15

Let $W \subseteq \mathbb{F}_2^n$ be a hyperplane, and let $S_W = \{L \mid L \subseteq W, \dim(L) = \ell\}$.

We can again do similar computations, and we find that $\mu(S_W) \approx 2^{-\ell}$ (because choosing L of dimension ℓ is sort of like choosing ℓ vectors, and the probability each is in W is roughly $\frac{1}{2}$), and $\Phi(S_W) \approx \frac{1}{2}$. So this is another violation.

So this was very unfortunate, and after that they kind of gave up for a few months. But then there's hope that once you look at these two examples, they're actually very structured. (If you had a random set that violated things, then you'd be in trouble. But these are very structured.) So there's the hope that maybe these are the 'only' ones in some sense, and maybe you can make the analysis of the reduction go through even if you had that weaker type of property.

First they showed you can modify the analysis in some nontrivial way to make it go through; but they didn't know how to prove that these are the only things. But eventually they figured it out.

We'll now write a concrete theorem, but the point is that when you don't have small-set expansion, you can try to say small-set expansion holds for 'most' sets, and we can characterize the few exceptions where it doesn't hold. Sometimes that's good enough, and that's the case here.

Theorem 20.16

For every $\varepsilon > 0$, there exists $r \in \mathbb{N}$ and $\delta > 0$ such that if $S \subseteq \mathbb{G}(n, \ell)$ is a set of vertices in the Grassman graph violating small-set expansion — i.e., with $\Phi(S) \leq 1 - \varepsilon$ — then there exist subspaces $A \subseteq W \subseteq \mathbb{F}_2^n$ with $\dim(A) + \operatorname{codim}(W) \leq r$ such that

$$\frac{\mu(S \cap S_A \cap S_W)}{\mu(S_A \cap S_W)} \geq \delta.$$

(You can imagine using 2-dimensional subspaces instead of 1-dimensional ones for our S_a example, and combining the two examples.)

Think of $\delta = 1$ for a moment to understand what this is saying — then it says that S essentially contains a copy of one of these sets. You can't say something that strong — this is too much. But instead you're saying S contains a *dense* part of one of these examples.

Here $S_A = \{L \mid A \subseteq L\}$ and $S_W = \{L \mid L \subseteq W\}$ (this is a slight abuse of notation).

This was nice, and then they kind of managed to prove the conjecture using it. But it turns out that with this phenomenon you can actually state a hypercontractive inequality that sort of captures this notion. And this is actually a much more general phenomenon than just with this graph. We won't prove this theorem (it's too hard), but we'll prove something for the p-biased cube that captures the intuition going on here.

§20.3 Global hypercontractivity

From now, we'll consider the p-biased cube — so we look at functions $f\colon (\{0,1\}^n, \mu_p) \to \mathbb{R}$, and we think of p as $o(1)$. (Much of what we'll say holds as long as $p \leq 1/2$, but it's only really interesting if p is small — otherwise you can prove stronger things.)

Definition 20.17. A function f is called (r, ε)-*global* if for all subsets $S \subseteq [n]$ with $|S| \leq r$ and for all partial assignments $x \in \{0,1\}^S$, we have $\|f_{S\to x}\|_2^2 \leq \varepsilon$.

Example 20.18

If f is a DNF formula with width r, then f is certainly not global — if you take S to be all the variables in the first clause D_1 and x to be all-1's, then $f_{S\to x}$ is just 1.

You can think more generally about the p-biased cube and the noisy hypercube on it; this graph is again not a small-set expander, and you can again cook up examples like S_a above. But if you try to cook up other ones you won't succeed — it turns out these are the only ones, and you can formulate a similar theorem.

What we'll do instead is state a hypercontractive inequality that works really well when your function is global; that'll be the main goal for this lecture, and probably the next.

Before we formulate the inequality, let's observe another thing about globalness — if $p = \frac{1}{2}$ then you have globalness of f for free, with quite good parameters. Specifically, if $p = \frac{1}{2}$ then f is $(r, 2^r \|f\|_2^2)$-global for every r — this is because

$$\|f\|_2^2 \geq \mathbb{P}_{y\sim\{0,1\}^S}[y = x] \cdot \|f_{S\to x}\|_2^2,$$

and the first factor is at least p^r (where $p = 1/2$).

But when p is sub-constant, you can clearly see globalness — not everything is global.

Theorem 20.19

Suppose $f\colon (\{0,1\}^n, \mu_p) \to \mathbb{R}$ has $\deg f \leq d$ and is (d, ε)-global. Then

$$\|f\|_4^4 \leq C^d \varepsilon \|f\|_2^2$$

(for some constant C).

If $p = \frac{1}{2}$ then we could replace ε by $2^r \|f\|_2^2$; then we'd have an exponential in d times $\|f\|_2^4$, which is exactly the same as the hypercontractive inequality (up to a different constant). So you can see this as a generalization of standard hypercontractivity. It turns out this is much more powerful with good globalness parameters.

The way this is going to work is first we'll prove this theorem (which will take a bit). And then we'll see a few applications, to convey the point that using this you can do stuff that you couldn't do using bare hypercontractivity.

§20.4 Fourier analysis over μ_p

The first thing we need is that we discussed how to do Fourier analysis for different domains, but for the p-biased cube you can do something more specific, which we'll need to introduce.

It turns out that for this domain, you can come up with a nice basis and just write it down. For univariate expressions — i.e., $L_2(\{0,1\}, \mu_p)$ — we can set up the orthonormal basis consisting of the two functions 1 and χ, where

$$\chi(x) = \frac{x - p}{\sqrt{p(1-p)}}.$$

(This really means you take x, and then you try to make sure that the average of χ is 0 by subtracting p, and then you try to make sure that $\mathbb{E}[\chi^2] = 1$ by dividing by the 2-norm.)

When you tensorize this, i.e., when you talk about $L_2(\{0,1\}^n, \mu_p)$, you get an orthonormal basis $\{\chi_S\}_{S\subseteq[n]}$ where

$$\chi_S(x) = \prod_{i\in S} \frac{x_i - p}{\sqrt{p(1-p)}}.$$

So these are the p-biased Fourier characters. In fact, you can try to run the standard proof of hypercontractivity using this decomposition instead of our usual Fourier-analytic decomposition. And most of it will go through, with one exception. When we did the standard proof, we wrote $f = g + (-1)^{x_i}\partial_i f$, and then we expanded the fourth norm and did our best. One feature of that was that if you take the expectation of $(-1)^{x_i}$ to the third power, it vanishes, and when you take it to the fourth power you still get 1. But when you look at these characters, things become more complicated. It is still true that $\mathbb{E}\chi(x) = 0$ and $\mathbb{E}\chi(x)^2 = 1$, but $\mathbb{E}\chi(x)^3 = 1/\sqrt{p}$ is already very large, and $\mathbb{E}\chi(x)^4 = 1/p$. This is the main difference between p-biased and non-biased, and this is why the whole proof of hypercontractivity would collapse.

Definition 20.20. For $f\colon \{0,1\}^n \to \mathbb{R}$ and $i \in [n]$, we define the derivative $\partial_i f\colon \{0,1\}^{n-1} \to \mathbb{R}$ by

$$\partial_i f(z) = f(x_i = 1, x_{-i} = z) - f(x_i = 0, x_{-i} = z).$$

(This is slightly different from the definition we used last time, but it's good for the p-biased cube.)

Definition 20.21. More generally, for $I = \{i_1, \ldots, i_t\} \subseteq [n]$, we define $\partial_I f = \partial_{i_1}\cdots\partial_{i_t} f$.

(Strictly speaking you need to check that the order of i_1, …, i_t doesn't matter, but this is true.)

Remark 20.22. Our goal is to prove this theorem. There's more than one known proof. The proof we'll present is elegant, but for it, it's more convenient to work with a notion that's equivalent to globalness but is somehow more convenient. We'll define that notion soon, and for that we need derivatives.

Definition 20.23. For $f\colon \{0,1\}^n \to \mathbb{R}$ and $S \subseteq [n]$, the *generalized influence* of f with respect to S is

$$I_S[f] = \|\partial_S f\|_2^2.$$

Fact 20.24 — Suppose that f is (r, ε)-global. Then for each $S \subseteq [n]$ with $|S| \le r$, we have $I_S[f] \le 2^{2r}\varepsilon$.

The way to read this is that if you're global, then you have small generalized influences. The next fact is the converse of this.

Fact 20.25 — If $\max_{|S|\le r} I_S[f] \le \varepsilon$, then f is $(r, 2^{2r}\varepsilon)$-global.

The way to read these facts is that essentially being global and having small generalized influences are the same property (you need to pay some exponential factors in r, but this is often not too bad).

We probably won't prove this, but it should remind us of something we saw on the problem set; it's essentially the same thing except that now p isn't $1/2$, and the only thing we really need is that there's a Fourier-analytic formula for these discrete derivatives:

Fact 20.26 — For all $i \in [n]$, we have

$$\partial_i f(z) = \frac{1}{\sqrt{p(1-p)}} \sum_{T \ni i} \widehat{f}(T) \chi_{T \setminus \{i\}}(z).$$

More generally, for $S \subseteq [n]$ we have

$$\partial_S f(z) = \frac{1}{\sqrt{p(1-p)}^{|S|}} \sum_{T \subseteq S} \widehat{f}(T) \chi_{T \setminus S}(z).$$

(The extra factor is the value of $\chi(1) - \chi(0)$.)

The first direction shouldn't require anything (you can prove it just by definitions); and the second direction can be proven by playing around with this, though we're not going to do it.

Remark 20.27. We can also get formulas for influences and total influences — we have

$$I_i[f] = \|\partial_i f\|_2^2 = \frac{1}{p(1-p)} \sum_{T \ni i} \widehat{f}(T)^2.$$

This means we have

$$I[f] = \frac{1}{p(1-p)} \sum_{T \subseteq [n]} |T| \, \widehat{f}(T)^2.$$

In the statement of Friedgut we multiplied the total influence by p, and this makes sense because of the first factor — this is off by a factor of $1/p$ from the average degree of your function.

Definition 20.28. For every $x \in \{0,1\}^n$ and $\rho \in (0,1)$, a *ρ-correlated input* $y \sim \mathrm{T}_\rho x$ is sampled as follows: for each $i \in [n]$ independently, we take $y_i = x_i$ with probability ρ, and otherwise sample $y_i \sim \mu_p$.

Once we have this, we can define the corresponding averaging operator in the obvious way.

Definition 20.29. We define $\mathrm{T}_\rho \colon L_2(\{0,1\}^n, \mu_p) \to L_2(\{0,1\}^n, \mu_p)$ as $\mathrm{T}_\rho f(x) = \mathbb{E}_{y \sim \mathrm{T}_\rho x} f(y)$.

§20.5 A different hypercontractive inequality

Now once we have this setup, we can state the theorem that will imply the global hypercontractive inequality stated earlier.

Theorem 20.30
For $f \colon (\{0,1\}^n, \mu_p) \to \mathbb{R}$ and $0 < \rho \le 1/4\sqrt{3}$, we have

$$\|\mathrm{T}_\rho f\|_4^4 \le \sum_{S \subseteq [n]} \left(\frac{12\rho^5}{\sqrt{p}}\right)^{|S|} \cdot I_S[f]^2.$$

This theorem actually seems quite bad (which is why we started by presenting the first, more sensible-looking, theorem). But the fortunate thing is that it's easier to prove.

(We'll start the proof, but probably won't finish it.)

First, we can write f in terms of its Fourier decomposition as

$$f(x) = \sum_{S \subseteq [n]} \widehat{f}(S)\chi_S^p(x)$$

(when we write χ_S^p we mean that we're referring to the p-biased character). We can syntactically define a multilinear polynomial $P(z_1, \ldots, z_n) = \sum_S \widehat{f}(S) \prod_{i \in S} z_i$; then we get

$$f(x) = P(\chi^p(x_1), \ldots, \chi^p(x_n)).$$

The next thing we're going to do is define an operator S_k. When we use T_ρ, we're using the same noise on each coordinate, and also the same p. But now we're going to do something a bit strange — in S_k, for the first k coordinates we apply correlation rate 4ρ, and for the next ones we apply ρ. So

$$\mathrm{S}_k = \mathrm{T}_{4\rho}^{\otimes k} \otimes \mathrm{T}_\rho^{\otimes(n-k)}.$$

(Right now this is very mysterious.)

We started out caring about $\mathrm{T}_\rho^{\otimes} f$. What we're going to do amusingly is change these characters from p-biased to unbiased (i.e., half-biased). This is going to be costly; it's going to cost us having to change the noise amount. So for each k, we define the strange-looking function $f_k\colon (\{0,1\}^k, \mu_{1/2}) \otimes (\{0,1\}^{n-k}, \mu_p) \to \mathbb{R}$ (where for the first k coordinates we're using the unbiased measure, and for the rest we're using the p-biased one) as

$$f_k(y_1, \ldots, y_k, x_{k+1}, \ldots, x_n) = \mathrm{S}_k P(\chi^{1/2}(y_1), \ldots, \chi^{1/2}(y_k), \chi^p(x_{k+1}), \ldots, \chi^p(x_n)).$$

If you look at f_0, this is the function that we care about. And if you look at f_n, that's a function that lives in some unbiased space, so there we can hopefully say something (i.e., we can use hypercontractivity there). This is indeed what we're going to do.

The bulk of this proof is the following lemma.

Lemma 20.31

For every k, we have

$$\lVert f_k \rVert_4^4 \leq \lVert f_{k+1} \rVert_4^4 + \frac{3\rho^4}{p} \lVert \partial_{k+1} f_{k+1} \rVert_4^4.$$

(We're trying to estimate $\lVert f_0 \rVert_4^4$; and the point is that we can upper-bound $\lVert f_k \rVert_4^4$ by the same thing for $k+1$, plus something depending on the derivative.)

Let's absorb this slowly. If we didn't have the second term, then it'd mean that when we move from f_k to f_{k+1}, the 4-norm increases; we could do that repeatedly to get that $\lVert f_0 \rVert_4^4 \leq \lVert f_n \rVert_4^4$. But that's an operator on an unbiased space, so we can look at the operator we get — which is $\mathrm{T}_{4\rho}$ — and so we could apply hypercontractivity. (This is why we have $4\sqrt{3}$ instead of $\sqrt{3}$, so we can apply hypercontractivity with 4ρ.) So we can then go down to the 2-norm; and that's the same whether you plug in p-biased or half-biased characters.

Remark 20.32. The normalization factors in our definition of influence may be a bit off, so what we've written down may be incorrect.

But we have this mess, which is where the generalized influences come in — and when you iterate this, you end up getting the thing in our theorem statement. (Eventually you have to recurse on both terms, because you want to change all the variables to be unbiased.)

The proof of this is not very illuminating; we'll see it next time. (It's some computation where you open things up and use Hölder and Cauchy–Schwarz and AM-GM.)

§21 April 25, 2024

Today we'll continue discussing global hypercontractivity.

§21.1 Proof of global hypercontractivity

(We did have some normalization issues last time; we'll fix them now.)

Theorem 21.1

Suppose that $f\colon (\{0,1\}^n, \mu_p) \to \mathbb{R}$ and $0 \le \mu \le 1/4\sqrt{3}$. Then

$$\|\mathrm{T}_\rho f\|_4^4 \le \sum_{S\subseteq[n]} \Big(12\rho^5 \cdot p(1-p)\Big)^{|S|} I_S[f]^2.$$

Last time, we discussed a more comprehensible version of this statement; we'll see that again, but this statement turns out to be nicer to prove.

The proof is kind of by induction, and the core of it is the following claim.

Throughout, we'll let $y \sim \mu_{1/2}^n$ and $x \sim \mu_p^n$. We'll use both unbiased characters $\chi^{1/2}$ and biased characters χ^p. And let's consider the multilinear polynomial

$$P(z_1, \dots, z_n) = \sum_{S\subseteq[n]} \widehat{f}(S) \prod_{i\in S} z_i,$$

where we take f and plug in the variables z_i instead of the corresponding characters. Then we can define functions f_0, …, f_n as follows: f_k takes k boolean bits and $n-k$ p-biased things, and it plugs into P the corresponding characters. Explicitly,

$$f_k(y_1, \dots, y_k, x_{k+1}, \dots, x_n) = P(\chi^{1/2}(y_1), \dots, \chi^{1/2}(y_k), \chi^p(x_{k+1}), \dots, \chi^p(x_n)).$$

Note that $f = f_0$. So our goal is to bound $\|\mathrm{T}_\rho f_0\|$.

We're going to slightly change the definition of f_k in a moment. We have this noise operator that we'll have to carry around all over the place, so we'll actually change the definition of f_k a bit. We define an operator S_k as follows — on the coordinates where we already have unbiased bits we apply 4ρ noise, and on the p-biased bits we apply ρ noise. Explicitly, $\mathrm{S}_k = \mathrm{T}_{4\rho}^{\otimes k} \otimes \mathrm{T}_\rho^{\otimes(n-k)}$. We will actually define

$$f_k(y_1, \dots, y_k, x_{k+1}, \dots, x_n) = \mathrm{S}_k P(\chi^{1/2}(y_1), \dots, \chi^{1/2}(y_k), \chi^p(x_{k+1}), \dots, \chi^p(x_n)),$$

so our goal is to upper bound $\|f_0\|_4^4$ (we've squeezed the noise operator into the definition).

Remark 21.2. Was this the original proof? The original proofs about global hypercontractivity were actually on Grassman graphs rather than the p-biased cube, and those proofs didn't look like this.

Here's the key claim. We want to upper-bound $\|f_0\|_4$. And if we look at these functions, f_n is a function of unbiased bits, so for that function we *can* bound the 4-norm using hypercontractivity. So our goal is to show that if you can upper-bound $\|f_n\|_4$, then you can upper-bound $\|f_0\|_4$. We'll do this one step at a time.

Claim 21.3 — For all $k \le n-1$, we have

$$\|f_k\|_4^4 \le \|f_{k+1}\|_4^4 + \frac{3\rho^4}{p} \|\partial_{k+1} f_{k+1}\|_4^4.$$

If you ignore the precise terms, it makes sense you'll get f_{k+1} and some derivative-looking thing, because the difference between f_k and f_{k+1} is only in the $(k+1)$st coordinate (so you'd expect the change to depend on the derivative).

Proof. We're just going to expand f_k and f_{k+1}, raise to the fourth power, and try to compare terms. We have

$$f_k(y_1, \dots, y_k, x_{k+1}, \dots, x_n) = g(y', x') + \rho \cdot \chi^\rho(x_{k+1})h(y', x')$$
$$f_{k+1}(y_1, \dots, y_{k+1}, x_{k+2}, \dots, x_n) = g(y', x') + 4\rho\chi^{1/2}(y_{k+1})h(y', x')$$

(we'll define these terms in a moment). The idea is we're partitioning the monomials into those that don't contain the $(k+1)$st term and those that do, and then we're factoring out the corresponding character. Explicitly $h = \mathrm{S}_k\partial_{k+1}P(y', x')$ where $y' = (y_1, \dots, y_k)$ and $x' = (x_{k+2}, \dots, x_n)$. If you ignore the noise operator for a moment, this makes sense because we've pulled out the common things and we're left with a derivative. And g is everything that doesn't contain z_{k+1}.

But then when you add in noise, you have to take into account how it affects these characters. We're applying ρ in f_k and 4ρ in f_{k+1}, so this is the difference between them.

Remark 21.4. Explicitly, we can write $P = P_{-(k+1)} + \partial_{k+1}P \cdot z_{k+1}$. Then we apply the noise operator. The noise operator, when we apply it on a product of two functions depending on different variables, will split. So really what we wrote is $\rho\chi^p(x_{k+1}) = \mathrm{T}_\rho\chi^p(x_{k+1})$.

Now we're really going to take fourth powers and expand. We're not going to write down all 2^4 terms. The first term will be $g(y', x')^4$. The next term will have three factors of $g(y', x')$ and one of the second term; but this is 0, because $\mathbb{E}\chi^\rho(x_{k+1}) = 0$ and y' and x' are independent of x_{k+1}, so its expectation is 0 and we don't have to write it down. The next thing is where we pick two factors of the first term and two from the second, giving

$$\binom{4}{2} \cdot \rho^2\mathbb{E}[g(y', x')^2 h(y', x')^2 \chi^p(x_{k+1})^2].$$

We can again get rid of the character using independence — its expectation is 1, so we just get

$$\mathbb{E}[g(y', x')^2 h(y', x')^2] = \|gh\|_2^2.$$

For the third term, we get

$$\binom{4}{3}\rho^3\mathbb{E}_{x', y'}[g(y', x')h(y', x')^3]\mathbb{E}_{x_{k+1}}[\chi^p(x_{k+1})^3].$$

But this is where the proof now differs from the uniform-measure case — in the uniform case this third power would give 0, but now that's no longer the case. So we'll leave it as it is. And finally, we get the last term of $\rho^4 \|h\|_4^4 \mathbb{E}\chi^p(x_{k+1})^4$.

So we end up with

$$\mathbb{E}_{y,x}f_k(y, x)^4 = \|g\|_4^4 + + \binom{4}{2}\rho^2 \|gh\|_2^2 + \mathbb{E}_{x',y'}[g(y', x')h(y', x')^3]\mathbb{E}_{x_{k+1}}\chi^\rho(x_{k+1}) + \rho^4 \|h\|_4^4 \mathbb{E}_{x_{k+1}}\chi^\rho(x_{k+1})^4.$$

Similarly, if we expand out f_{k+1} we get

$$\mathbb{E}_{y,x}f_{k+1}(y, x)^4 = \|g\|_4^4 + \binom{4}{2}(4\rho)^2 \|gh\|_2^2 + (4\rho)^4 \|h\|_4^4$$

(here for the third term we get an unbiased character instead of the p-biased one, and its expectation is 0; and similarly the fourth power becomes 1).

(The point of the 4's is you need a bit of slack to resolve the extra terms.)

Now let's try to understand the two nasty things in f_k. It's not hard to see $\mathbb{E}\chi^p(x_{k+1})^3$ is at most $1/\sqrt{p}$ in absolute value — we have

$$\left|\mathbb{E}\chi^p(x_{k+1})^3\right| \leq \max|\chi^p(x_{k+1})|\,\mathbb{E}\,|\chi^p(x_{k+1})|^2,$$

and using the explicit expression for the character we can see that the max is at most $1/\sqrt{p}$. And by the same logic, the expectation of the fourth power is at most $1/p$. (If you really want to, you can actually compute them, but these estimates are fairly tight, so there's no reason to.)

Now we're going to stare at these terms and try to hammer at them using Hölder and Cauchy–Schwarz and AM-GM. First let's say we want to bound the third term

$$\frac{\rho^3}{\sqrt{p}}\left|\mathbb{E}gh^3\right|.$$

We think of $gh = gh \cdot h^2$ and then use Cauchy–Schwarz to get that this is at most

$$\frac{\rho^3}{\sqrt{p}}\left\|gh\right\|_2\left\|h\right\|_4^2$$

(by Cauchy–Schwarz). The reason we want this type of term is these are the things we can try to absorb into the f_{k+1} thing — the name of the game is absorbing these by terms that appear in the expectation for f_{k+1}.

Now we'll attach one of the ρ's to the gh and the rest to h, to rewrite this as

$$\rho\left\|gh\right\|_2\cdot\frac{\rho^2}{\sqrt{p}}\left\|h\right\|_4^2.$$

And using the inequality $ab \leq \frac{1}{2}(a^2+b^2)$, this is at most

$$\frac{1}{2}\left(\rho^2\left\|gh\right\|_2^2+\frac{\rho^4}{p}\left\|h\right\|_4^4\right).$$

(The name of the game is that in gh there shouldn't be any dependence of the coefficient on p, because we don't have p in f_{k+1}; but for h it's fine to have p, since this will be swallowed into our derivative term.)

So overall, we get that

$$\mathbb{E}f_k^4 \leq \left\|g\right\|_4^4+(6\rho^2+2\rho^2)\left\|gh\right\|_2^2+\left(\frac{2\rho^4}{p}+\frac{\rho^4}{p}\right)\left\|h\right\|_4^4$$

(collecting everything). Now we're almost done. The first part is certainly upper-bounded by the thing in f_{k+1} (because the second coefficient has $\binom{4}{2}\cdot 4^2\rho^2$, and $6\cdot 4^2 > 8$). And the other thing is where the derivative comes in — h is exactly the same thing as taking the derivative of f_{k+1}, so we get

$$\mathbb{E}f_k^4 \leq \mathbb{E}f_{k+1}^4+\frac{3\rho^4}{p}\mathbb{E}(\rho_{k+1}f_{k+1})^4. \qquad \square$$

Remark 21.5. The nice thing is that this is kind of clean; because it's so clean it seems like magic. But it's not really magic. Once you understand that some bound along the lines of this claim should work — just because the only difference between f_k and f_{k+1} is in the $(k+1)$st coordinate, so you should pay something in the derivative — then the rest involves some trial and error, but you can eventually get to it.

The original proof for the Grassman graph was more direct, and involved writing out fourth powers and doing a bunch of case analysis; here the case analysis is sort of absorbed into the claim.

Now that we have this key claim, we can iterate it.

Corollary 21.6

We have

$$\|f_0\|_4^4 \leq \sum_{S \subseteq [n]} \left(\frac{3\rho^4}{p}\right)^{|S|} \|\partial_S f_n\|_4^4.$$

Proof. We'll prove by induction on t that for every k, we have

$$\|f_k\|_4^4 \leq \sum_{S \subseteq [k+t] \setminus [k]} \left(\frac{3\rho^4}{p}\right)^{|S|} \|\partial_S f_{k+t}\|_4^4$$

(we sum over the variables that have been changed, and we have to take the derivatives and pay some factor). The thing we want is the case $k = 0$ and $t = n$.

The base case is $t = 1$; this is trivial by the key claim. (It's precisely what we wrote there, because taking the derivative with respect to the empty set doesn't do anything.)

For the inductive step, spupose it's true for some $t \geq 1$; we'll prove it for $t + 1$. We have

$$\|f_k\|_4^4 \leq \sum_{S \subseteq [k+t] \setminus [k]} \left(\frac{3\rho^4}{p}\right)^{|S|} \|\partial_S f_{t+k}\|_4^4$$

by the inductive hypothesis. And now we can apply the key claim again — $\partial_S f_{t+k}$ is still a function, so we can apply the claim to it and replace the next variable to rewrite this as

$$\sum_{S \subseteq [t+k] \setminus [k]} \left(\frac{3\rho^4}{p}\right)^{|S|} \left(\|\partial_S f_{t+k+1}\|_4^4 + \frac{3\rho^4}{p^4} \|\partial_{t+k+1} \partial_S f_{t+k+1}\|_4^4\right).$$

And this is exactly a sum in the form we wanted, since when we took another derivative we got another factor, and when we didn't take a derivative we got the same factor. □

So we've managed to prove the corollary, and therefore we've managed to replace all our variables with unbiased ones. And now we can finish the proof.

By the corollary, we have

$$\|\mathrm{T}_\rho f\|_4^4 \leq \sum_S \left(\frac{3\rho^4}{p}\right)^{|S|} \|\partial_S f_n\|_4^4.$$

Now to be precise, we need a bit more notation — let $f'(y_1, \ldots, y_n) = P(\chi^{1/2}(y_1), \ldots, \chi^{1/2}(y_n))$ be where we plug in unbiased characters but don't apply any noise. Then $f_n = \mathrm{T}_{4\rho} f'$ (this is just the definition). But we want to apply hypercontractivity. Here we're first applying noise and then taking a derivative, and we want to flip the two operations. You can do this, but you have to multiply by the right factor — we have

$$\partial_S f_n = (4\rho)^{|S|} \cdot \mathrm{T}_{4\rho} \partial_S f'.$$

(On the left we first apply noise and then take the derivative; but we want to do it the other way around. Taking the derivative drops variables, so the characters get multiplied by less noise; that's why we have this factor to compensate. You can see this immediately by writing out the Fourier analytic formulas for both sides.)

Then we have

$$\|\partial_S f_n\|_4^4 = (4\rho)^{|S|} \|\mathrm{T}_{4\rho} \partial_S f'\|_4^4 .$$

And now we can finally apply hypercontractivity — we have $4\rho \leq 1/\sqrt{3}$, so we can go from 4-norms to 2-norms using hypercontractivity to get that this is at most

$$(4\rho)^{|S|} \|\partial_S f'\|_2^4$$

(using hypercontractivity for $\mu_{1/2}$). That's the whole point of this proof — once you've managed to replace all the bits by unbiased bits, you can apply the good hypercontractivity.

And now that we're in the unbiased world, we know what these derivatives look like — so this is

$$(4\rho)^{|S|} \left(\sum_{T \supseteq S} \widehat{f}(T)^2\right)^2$$

(the thing inside the expression is the 2-norm of the derivative).

And this is the place we made the error last time because of a normalization mistake. This expression looks like a generalized influence — we saw that

$$I_S[f] = \frac{1}{(p(1-p))^{|S|}} \sum_{T \supseteq S} \widehat{f}(T)^2.$$

These sums are almost the same, up to this factor; so you get that this is

$$(4\rho)^{|S|} \cdot (p(1-p))^{2|S|} \cdot I_S[f]^2.$$

Now we plug this into our expression for $\|\mathrm{T}_\rho f\|_4^4$ to get that

$$\|\mathrm{T}_\rho f\|_4^4 \leq \sum_{S \subseteq [n]} \left(\frac{3\rho^4}{p}\right)^{|S|} \cdot (4\rho)^{|S|} \cdot (p(1-p))^{2|S|} \cdot I_S[f]^2.$$

And now when you simplify all this, one of the p's cancel and you get that this is at most

$$\sum_{S \subseteq [n]} (12\rho^5 p(1-p))^{|S|} I_S[f]^2$$

(we're dropping a square on the $1 - p$ or something like this, but it doesn't matter).

§21.2 Some applications

The first application will be a more comprehensible version for low-degree functions, which we started by stating last time.

Theorem 21.7

Suppose $f \colon (\{0,1\}^n, \mu_p) \to \mathbb{R}$ has degree at most d and is (d, ε)-global. Then

$$\|f\|_4^4 \leq C^d \varepsilon \|f\|_2^2$$

(where $C > 0$ is an absolute constant).

If ε were the optimal globalness parameter, namely $\|f\|_2^2$, then you'd get exactly what hypercontractivity gives up to an exponential factor.

Proof. The proof has one neat technical trick, but that's about it. The inequality we proved only holds when we have noise applied to some function f. Here we just have some function f, but with no noise. So we'd like to find some function g which f is the noisy version of. If you think of the noise operator as an averaging operator, this is hard; but if you just look at how it acts on characters, then you can do it. We're going to take $\rho = 1/4\sqrt{3}$, and we want some g with $f = \mathrm{T}_\rho g$. If we spell out the Fourier expansion of g, then we know what $\mathrm{T}_\rho g$ looks like; and we know f and have some equality, so we can cook up the Fourier expansion of g to be what works. So we take

$$g(x) = \sum_{S \subseteq [n]} \rho^{-|S|} \widehat{f}(S) \chi_S(x).$$

This is kind of a nice trick that's sometimes useful to know. (This is also what you use to go between standard hypercontractivity and the noise operator version.)

Now we'll see why generalized influences are nice. If f is global, then we know something about its restrictions. But we want g to be global, because we're going to get generalized influences of g. But how do we say that? This is where it's helpful to know that globalness is kind of equivalent to having small generalized influences — note that for all $S \subseteq [n]$ of size $|S| \leq d$, we have

$$I_S[f] \leq 2^{2d} \varepsilon$$

(we proved last time that if you're global then you have small generalized influences). But we can directly relate the generalized influences of g to those in f — we have

$$I_S[g] = \frac{1}{(p(1-p))^{|S|}} \sum_{T \subseteq S} \rho^{-2|T|} \widehat{f}(T)^2 \leq \rho^{-2d} I_S[f] \leq 2^{O(d)} \cdot \varepsilon.$$

So that's another benefit of generalized influences.

> **Remark 21.8.** This also means g is $(d, 2^{O(d)}\varepsilon)$-global. (We don't actually need this though.)

Returning to what we're trying to prove, we have

$$\|f\|_4^4 = \|T_\rho g\|_4^4 \leq \sum_{S \subseteq [n]} (12\rho^5 p(1-p))^{|S|} I_S[g]^2$$

by the theorem. And we use the fact that the generalized influences are small to replace one of these squares with this bound (and we also absorb the $12\rho^5$ into the exponential in d), so this is at most

$$2^{O(d)} \cdot \varepsilon \sum_{S \subseteq [n]} (p(1-p))^{|S|} I_S[g]$$

(if you have derivative of order more than d, then you just get 0). And now we have this sum, adn this is kind of the Parseval thing — we just spell out the formula and get that this is

$$2^{O(d)} \varepsilon \sum_{S \subseteq [n]} \sum_{T \supseteq S} \rho^{-2|T|} \widehat{f}(T)^2$$

(the two $(p(1-p))^{|S|}$'s cancel). If we rearrange this double sum, each $\widehat{f}(T)^2$ is counted in the number of subsets it has, so we get

$$2^{O(d)} \sum_{T \subseteq [n]} \rho^{-|T|} 2^{|T|} \widehat{f}(T)^2.$$

And this is again $2^{O(d)}$, so we get that this is at most

$$2^{O(d)} \varepsilon \sum \widehat{f}(T)^2 = 2^{O(d)} \varepsilon \, \|f\|_2^2$$

by Parseval. □

Here's another corollary.

Theorem 21.9

Suppose that $A \subseteq \{0,1\}^n$ is such that $f = 1_A$ is $(d, C^{-d}\varepsilon)$-global (with respect to $\mu_p()$), where C is a sufficiently large constant. Then A has edge-expansion close to 1 — precisely, $\Phi_{1/2}(A) \geq 1 - \varepsilon - 2^{-d}$, where expansion is over the $\frac{1}{2}$-correlated cube.

In other words, if we pick some vertex $x \in \mu_p$ conditioned on $x \in A$ and then take a noisy copy $y \sim \mathrm{T}_{1/2}x$, then the probability that the noisy copy will still be in A is small, specifically

$$\mathbb{P}_{x\sim\mu_p, y\sim\mathrm{T}_{1/2}x}[y \in A \mid x \in A] \leq \varepsilon + 2^{-d}.$$

The proof is in the notes; it's the same Hölder-type stuff we saw ten or twelve lectures ago, but using this stuff instead of standard hypercontractivity.

But this is the theorem from earlier saying that if we want to construct sets A with expansion bounded away from 1, the only way is to use things looking like indicators of $x_1 = 1$ (very local things). Because this theorem says that if you're not local — i.e., you're global — then your expansion is close to 1.

The next corollary is about sharp thresholds. All this discussion started with Friedgut's theorem, which said that if you have a monotone graph property with small total influence, then it's close to a DNF formula. There's a conjecture you don't need 'graph property', and some things along these lines are known.

Theorem 21.10 (Sharp thresholds)

Suppose that $f\colon (\{0,1\}^n, \mu_p) \to \{0,1\}$ is monotone and has small total influence $pI[f] \leq k \operatorname{Var}(f)$. Then there exists $S \subseteq [n]$ of size at most $2k$ such that $\mu_p(f_{S\to 1}) \geq 2^{-O(k)}$.

So we can't prove Friedgut's conjecture, but what we can prove is that there's a very weak kind of DNF structure inside f — there's a small set such that if we fix these variables to 1, then our measure becomes large. So S kind of looks like a clause of a DNF.

This is a very modest improvement on what you can get with Friedgut (which would give you k^2 in the exponent). This is the right bound, but we don't know how to improve this structure into a DNF structure. (We won't prove this.)

§21.3 Concluding remarks

The notes have one more corollary, which we won't mention. This will conclude the discussion on global hypercontractivity. But this sort of thing extends to toher domains — for example, in the p-biased cube we had a variable that we could restrict to 1 or 0. In S_n you have permutations $\pi\colon [n] \to [n]$, and the analog of restricting a variable is restricting the value of the permutation on a certain input — i.e. $\pi(i) \to j$. But there is also another type of restriction — fixing $\pi^{-1}(i)$. Then once you have the analogs of restrictions, you can define analogs of globalness and prove everything here (not with the same proof; this is where some representation theory enters).

There are also things called *high-dimensional expanders* which we will not define; and it also extends there. (There was one paper on the project list about this.)

And there's also the Grassman stuff.

Each one of these things has some nice consequences.

Next time we'll talk about something completely different; we'll talk about some progress on the sunflower lemma achieved a few years ago.

§22 April 30, 2024

Today we'll talk about the sunflower lemma and recent developments by Alweiss–Lovett–Wu–Zhang; in the rest of the course we'll talk about Gowers norms and some related things.

§22.1 Sunflowers

Suppose we have some collection of subsets $\mathcal{F} \subseteq \binom{[n]}{2}$.

Definition 22.1. We say a collection of sets $A_1, \ldots, A_r$ is a *sunflower* if their pairwise intersections are all the same — i.e., $A_i \cap A_j = \bigcap_{k=1}^r A_k$ for all $i \neq j$.

The reason this is called a sunflower is that you can imagine drawing the sets pictorially, each as an oval; then the common intersection looks like the center of a sunflower, with the remainder of the sets as the 'petals.'

Theorem 22.2 (Sunflower lemma)

There exists some $f(w, r)$ such that if $|\mathcal{F}| \geq f(w, r)$, then $\mathcal{F}$ must contain an r-sunflower.

We're intentionally writing it this way, which doesn't tell us what the bound on f is; the main point is that there's a dependency on w and r (which is necessary), but no dependency on the universe size n.

But now that we know there's a finite bound, there's the question of what the bound actually looks like.

Lemma 22.3 (Erdős–Rado 1960)

If $|\mathcal{F}| \geq (rw)^w$, then $\mathcal{F}$ contains an r-sunflower.

(This is slightly weaker than what they proved, to avoid messing around with factorials.)

Proof. For each $i \in [n]$, consider the set $\mathcal{F}_i = \{A \setminus \{i\} \mid A \in \mathcal{F},\ A \ni i\}$ (where we take all the sets containing i in $\mathcal{F}$, and then remove i from them). Then there are two cases.

Case 1 (There exists i such that $|\mathcal{F}_i| \geq \frac{1}{rw} |\mathcal{F}|$). In that case, we can look at $\mathcal{F}_i$, which has size at least $(rw)^{w-1}$; its uniformity w has droppped by 1, so we can do induction to say that $\mathcal{F}_i$ contains an r-sunflower $B_1, \ldots, B_r \in \mathcal{F}_i$; and then $B_1 \cup \{i\}, \ldots, B_r \cup \{i\}$ is a sunflower in $\mathcal{F}$.

So this is the case where there's some element which is very popular in the elements of $\mathcal{F}$; the second case is when there's nothing like that.

Case 2 (For all i, we have $|\mathcal{F}_i| < \frac{1}{rw} |\mathcal{F}_i|$). In this case, we'll actually show that you can find r *disjoint* sets inside $\mathcal{F}$. (Disjoint sets are automatically a sunflower, because the pairwise intersections are empty.)

To see this, fix $A_1 \in \mathcal{F}$. Then we can upper-bound the number of sets in $\mathcal{F}$ that intersect A_1 — we have

$$\#\{B \in \mathcal{F} \mid A_1 \cap B \neq \emptyset\} \leq \sum_{i \in A_1} |\mathcal{F}_i| \leq |A_1| \max_i |\mathcal{F}_i| < w \cdot \frac{1}{rw} |\mathcal{F}| = \frac{1}{r} |\mathcal{F}|$$

(because if we want to intersect A_1, then we have to intersect it at some i, which means B has to be in $\mathcal{F}_i$ for some $i \in A_1$). This is less than the total number of sets, so we can find $A_2 \in \mathcal{F}$ which is disjoint from A_1. And then we can iterate this — if we've found pairwise disjoint $A_1, \ldots, A_t$, then now we have

$$\#\{B \in \mathcal{F} \mid A_i \cap B = \emptyset \text{ for some } i\} \leq \frac{t}{r} |\mathcal{F}| < |\mathcal{F}|,$$

so we can find $A_{t+1} \in \mathcal{F}$ such that $A_1, \ldots, A_{t+1}$ are pairwise disjoint. □

Remark 22.4. This proof actually gives $(r-1)^w w!$, but we didn't want to write down all the factorials.

This is the sunflower lemma from 1960, and this bound hasn't moved a lot since then until a few years ago (there may have been some improvements, but not very large ones). But the conjecture was that the bound should be much better.

Conjecture 22.5 (Sunflower conjecture) — For every $r \in \mathbb{N}$, there is some $C > 0$ such that if $\mathcal{F} \subseteq \binom{[n]}{w}$ has $|\mathcal{F}| \geq C^w$, then $\mathcal{F}$ contains an r-sunflower.

And ALWZ proved something in between the previous bound and this conjecture:

Theorem 22.6 (Alweiss–Lovett–Wu–Zhang)

There exists an absolute constant $C > 0$ such that if $|\mathcal{F}| \subseteq \binom{[n]}{w}$ has $|\mathcal{F}| \geq (Cr\log(wr))^w$, then $\mathcal{F}$ contains an r-sunflower.

Our goal today is to give most of the ideas that give a bound along these lines; we may get $\log^2$ instead.

Remark 22.7. Is there an obvious construction to motivate C^r in the conjecture? The answer is yes, but Dor doesn't remember it off the top of his head.

§22.2 Some definitions

Definition 22.8. Given $\mathcal{F} \subseteq \binom{[n]}{w}$ and $A \subseteq [n]$, we define the *link* of A as

$$\mathcal{F}_A = \{S \mid S \in \mathcal{F},\ A \subseteq S\}.$$

This is kind of like the subcollections $\mathcal{F}_i$ that we defined in the earlier proof, but now we're not just looking at a single element.

Definition 22.9. We say that $\mathcal{F} \subseteq \binom{[n]}{r}$ is κ-*spread* if for all $A \subseteq [n]$,

$$|\mathcal{F}_A| \leq \kappa^{-|A|} \cdot |\mathcal{F}|\,.$$

The reason for the word 'spread' is that this condition is telling you in a moral sense that the elements of $\mathcal{F}$ are sort of spread around the universe kind of evenly — there's no A where there are too many elements of $\mathcal{F}$ containing it.

Remark 22.10. In the proof we gave, in Case 2 we used some weak form of spreadness — the condition was that if we restrict to one element we're not very large. So in particular, from that proof we know that if $\mathcal{F}_A$ is rw-spread and $\mathcal{F}$ is sufficiently large, then $\mathcal{F}$ contains r disjoint sets (and therefore a sunflower).

Theorem 22.11

If $\mathcal{F} \subseteq \binom{[n]}{w}$ has $|\mathcal{F}| \geq \kappa^w$ and is κ-spread for $\kappa = O(r\log(rw))$, then $\mathcal{F}$ contains r disjoint sets.

This is where the heart of the matter is — once you get this, you're done. What you do is essentially the previous proof — as long as you're not κ-bounded you find some A where your set is large and pass to it; and then once you're spread you get r disjoint sets, which end up giving a sunflower.

So this is the core. It makes some sense — we already saw that if you're sufficiently spread then you can find r disjoint sets — but it turns out you need much less spread than this trivial bound.

§22.3 Boolean formulas

First, we'll offer some intuition in terms of formulas. Given a family $\mathcal{F}$, we can define a CNF formula $\varphi_{\mathcal{F}}(x) = \bigvee_{S\in\mathcal{F}} \bigwedge_{i\in S} x_i$. This is a nice formula, and evaulating to 1 or not is intimately related to the structure of $\mathcal{F}$ — if $\varphi_{\mathcal{F}}(x) = 1$ and we consider the set $W = \operatorname{supp}(x)$, what can we say about W? This means there is some $S \in \mathcal{F}$ with $S \subseteq W$. In other words,

$$\varphi_{\mathcal{F}}(x) = 1_{\mathcal{F}\uparrow}(x).$$

This is a nice observation, but it's not clear why this is related to anything at all.

We'll write a lemma, and then explain why if you have the lemma you're done; and then we'll handwave a bit about why we should expect the lemma to be true.

Lemma 22.12

Suppose that $\mathcal{F} \subseteq \binom{[n]}{w}$ has size at least κ^w and is κ-spread for $\kappa = O((\log w)^2)$. Then if we sample $x \in \{0,1\}^n$ according to the measure $\mu_{p \log w}$, we have

$$\mathbb{P}_{x\sim\mu_{p\log w}}[\varphi_{\mathcal{F}}(x) = 1] \geq 1 - \log w \cdot \sqrt{p}.$$

What's going on? We said that if we find points x that evaluate to 1, then we get that there's some element of $\mathcal{F}$ fully contained in its support. This lemma says that if you sample x according to this p-biased measure, then you evaluate to 1 with some reasonably big probability — think of

$$p \sim \frac{1}{C\log^2 w}.$$

Then we're sampling $x \sim \mu_{1/C\log w}$, and this is saying e.g., that $\mathbb{P}[\varphi_{\mathcal{F}}(x) = 1] \geq 0.99$.

Once we have this, we claim you can find three disjoint elements inside $\mathcal{F}$. Why? You can partition $[n]$ into $\frac{1}{2}C\log w$ buckets $B_1, \ldots, B_\ell$, where you include each element in one of them with equal probability. Then you can sample x_1 which is 0 outside B_1 but is uniform in B_1; and same for all the other buckets. Then we claim that marginally, each of $x_1, \ldots, x_\ell$ is distributed according to the right measure — because the probability an element is in B_1 is $\frac{2}{C\log w}$, and the probability we set it to 1 in that case is then $\frac{1}{2}$. So marginally each x_i is distributed according to $\mu_{1/C\log w}$. And this means $\mathbb{E}\sum_{k=1}^{\ell} 1_{\varphi_{\mathcal{F}}(x_k)=1} \geq 0.99\ell$. In particular, we can find some sampling where this sum is at least 3 (3 is a very modest number; we can go much higher). But now the supports of the things we'll find are disjoint, just by construction.

Remark 22.13. In other words, what we want to do is sample $x_1, \ldots, x_\ell$ *jointly*, such that each one of them is distributed according to $\mu_{p\log w}$, but their supports are disjoint. The way we do this is by randomly partitioning $[n]$ into ℓ buckets, where place each i in each bucket with equal probability. And then we use $\mu_{1/2}$ in each.

What's $\mathbb{P}[x_1(j) = 1]$? For this to happen, we first need to place $j \in B_1$, and then we need to sample it to be 1; the probability this happens is

$$\mathbb{P}[j \in B_1] \cdot \frac{1}{2} = \frac{1}{\ell} \cdot \frac{1}{2} = \frac{1}{C\log w}.$$

This holds for every j, and they're independent. (This is quite a common trick, and is quite useful.)

§22.4 Some motivation

We'd like to connect this to something we've seen. This looks sort of like a sharp threshold theorem — you have some $\mathcal{F}$ that's not that small and is a little spread. And this says if you go slightly above the range of $\mathcal{F}$ itself, suddenly you're 1 with high probability. But the proof of this is going to be very different from stuff we've seen (it won't involve Friedgut or juntas or anything like that).

First we'll try to motivate this statement. To start with, let's do some sanity checks.

Suppose we sample $x \sim \mu_p$, and we want to count the expected number of clauses that evaluate to 1 — i.e.,

$$\mathbb{E}_{x\sim\mu_p}\#\{S \in \mathcal{F} \mid x_S = \mathbf{1}\}.$$

If we want to have any chance of the above event happening, this expectation had better be larger than 1 (otherwise the probability there are no such S is decent). And this is $|\mathcal{F}| \cdot p^w$.

So here you see why we might expect a bound of C^w — if you had $|\mathcal{F}| \geq (10r)^w$ and we picked $p = \frac{1}{3r}$, then this expectation would be quite a lot — at least $(3r)^w > 1$.

This is just the expectation. If we additionally knew there's 'good concentration' of this number $\#\{S \in \mathcal{F} \mid x_S = \mathbf{1}\}$, then we would expect $\mathbb{P}_{x\sim\mu_p}[\varphi_{\mathcal{F}}(x) = 1]$ to be close to 1 (e.g., $1 - \frac{1}{3r}$), because the expectation is very large and we're just asking for it to be more than 1. Then the above argument will work, and you'll get r disjoint sets.

So the point is it's very easy to compute the expectation, but it may be the case that this is completely wild — most of the time it's 0 but some of the time it's extremely large — and then this would break.

Let's look at $\varphi_{\mathcal{F}}(x)$ and say something simple. One case where we have good concentration is when these clauses are disjoint — because then the event that one clause is 1 is completely independent of another clause being 1. Then we can make this argument go through (though that's kind of silly, because we already have the result we want).

But the point is that the loss here — the lack of concentration — mainly comes from the interaction between clauses (if there's too much interaction, then concentration could break). And the hope in this lemma is that if you're κ-spread, then the interaction between clauses is hopefully not enough to break this, and something like this should hold.

If you look at this carefully and use Chebyshev or Jensen, you can prove that if you're sufficiently κ-spread, then there is good concentration and everything holds; but the κ you need for that argument is kind of like rw, so it's not very useful. So in order to actually get an improvement, you need a different argument.

§22.5 Random restrictions

Now we get to the main idea.

Definition 22.14. We say that a pair of subsets (S, W) is *c-good* if there exists $S' \in \mathcal{F}$ such that:

(1) $S' \setminus W \subseteq S \setminus W$.

(2) $|S' \setminus W| \leq (1 - c)w$.

Otherwise, we say (S, W) is *c-bad.*

Think of W as kind of large (e.g., $\text{supp}(x)$ in the earlier argument) and S as a member of $\mathcal{F}$.

What does this mean? Let's try to motivate it. Let's say we look at $\varphi_{\mathcal{F}}(x) = \bigvee_{S\in\mathcal{F}} \bigwedge_{i\in S} x_i$, and we choose W in some way and restrict ourselves to just x such that $x_W = 1$ (kind of like a random restriction). Then the interpretation of (1) in terms of $\varphi_{\mathcal{F}}|_{x_W=1}$ is that the clause corresponding to S in this formula *implies* the clause corresponding to S' (because when all the variables in S/W are true, so are all the ones in S'/W). So then S is kind of redundant, meaning we can throw it away from our restricted formula (because it never helps us).

And for (2), this is saying that the width of this clause corresponding to S' in the restricted formula is significantly smaller than the original width (which is w).

We're going to take W of size pn, where p is going to be determined, but we can think of it as $p \sim \frac{1}{(\log w)^2}$. The idea is to show that if we sample W randomly among things of this size, then the number of bad pairs is very small.

WHy is this interesting? Let's imagine all the pairs are good. Then we can clean up all the clauses given by S', since they don't help us anyways; and the new formula has significantly smaller width than the original. (We restricted a $1/\mathsf{polylog}$ fraction of the coordinates, but our width dropped from w to $(1-c)w$.) Then the hope is to iterate — we do this again and again, where each time we restrict a few coordinates and drop the width, until the width becomes very small. Then once the width is long enough, you can apply some trivial argument and finish up the proof.

So the crux is that we're going to look at φ, apply some random restriction, and show that it 'simplifies' in the sense that the width kind of drops.

To actually do this, we need a slightly different definition of spread (it's the same one, but has more letters).

Definition 22.15. We say that $\mathcal{F} \subseteq \binom{[n]}{w}$ with weight function $\mathsf{wt}\colon \mathcal{F} \to \mathbb{R}_{>0}$ is $(s_0, s_1, \ldots, s_{w-1})$-*spread* if $\mathsf{wt}(\mathcal{F}) \geq s_0$, and $\mathsf{wt}(\mathcal{F}_A) \leq s_{|A|}$ for all $A \subseteq [n]$.

This makes it easier to iterate, but if you don't care about iteration you can ignore it and just think about the previous definition.

Here's the main lemma. (We're going to ignore the weight function, and assume it's just uniform.)

Lemma 22.16

Suppose that $\mathcal{F} \subseteq \binom{[n]}{w}$ is $(s_0; s_1, \ldots, s_{w-1})$-spread. Then if we sample W of size pn, we have

$$\mathbb{E}_W \#\{S \in \mathcal{F} \mid (S, W) \text{ bad}\} \leq 2\left(\frac{4}{p}\right)^w s_{(1-c)w}.$$

The way to read this is that this is much smaller than the size of $\mathcal{F}$ itself — for each thing we restrict we pay a κ-factor — but then we pay a little bit, because $4/p > 1$. Eventually we'll pick p that makes this factor kind of comparable to the gain we get from $s_{(1-c)w}$. The exact bounds don't really matter; the main point is this is better than $|\mathcal{F}|$.

Remark 22.17. How should we think of s_0, s_1, ...? We should think of s_0 as κ^w, $s_1 = \kappa^{-1}s_0$, ..., $s_{w-1} = \kappa^{-(w-1)}s_0$. It's just that when you iterate it's easier to keep track of things with this notation. (For each element you contain, you pay a factor of κ, as before.)

Then if we rewrite this bound, we get $2(4/p\kappa^{1-c})^w \cdot s_0$, and this thing is much smaller than 1, so this is a win.

Proof. The proof is by magic — we're just going to count the number of bad pairs. We're going to encode bad pairs using just a little bit of information.

First let's fix an ordering on $\mathcal{F}$ — say $\mathcal{F} = \{S_1, \ldots, S_N\}$. (We just need some ordering; we don't care what it is.)

The first piece of information we need is $S \cup W$. Here W has size pn, and S is of size w, so this is of size between pn and $pn + w$. So the number of options for $S \cup W$ is at most

$$\sum_{k=0}^{w} \binom{n}{pn+k} = \binom{n}{pn} \sum_{k=0}^{w} \frac{\binom{n}{pn+k}}{\binom{n}{pn}} \leq \binom{n}{pn} \sum_{k=0}^{w} p^{-k}$$

using some facts about binomial coefficients (specifically, that $\binom{n}{pn+1}/\binom{n}{pn} \leq 1/p$ and so on). And this geometric series is dominated by the top term, so this is at most

$$\binom{n}{pn} \cdot 2 \cdot p^{-w}.$$

Next, we consider the smallest j such that $S_j \subseteq S \cup W$. Once you've told me what is $S \cup W$, I know how to find S_j myself, so I don't need any help — so here we don't have to specify any further information.

The next step is to specify $A = S \cap S_j$. This is some subset of S_j, and S_j has size w, so there are at most 2^w options for A.

And next, let's look at $S_j \setminus W$. We know $S_j \subseteq S \cup W$, and $S \cup W \setminus W = S \setminus W$. But (S, W) is bad; and because $S_j \setminus W \subseteq S \setminus W$, then $S_j \setminus W$ must be large (or else (S, W) wouldn't be bad). So we know $|S_j \setminus W| \geq (1 - c)w$.

Now we claim that $A \setminus W$ is large — note that $A \setminus W = (S_j \setminus W) \cap (S \setminus W) = S_j \setminus W$, so $|A \setminus W| \geq (1 - c)w$. In particular, this means A itself is large — specifically, $|A| \geq (1 - c)w$.

This is really telling us that once you know S_j, you know lots of things about S — you know it contains this big subset A. And now how many options are there for S itself? S has to be a member of $\mathcal{F}_A$, and A is kind of large, so there are not too many options — there's at most $s_{(1-c)w}$ options.

So far, we've managed to recover S; and now we just need to recover W. We have $S \cup W$, so we can also just specify $S \cap W$ and be done — this has at most 2^w options.

So with all this, it should be clear that once we have all this information we now know S and W (we know S, $S \cap W$, and $S \cup W$).

So if we want to count the total number of bad pairs, we can count the number of outcomes of this specification process — we can just multiply these numbers, and we get that this is at most

$$\binom{n}{pn} \cdot 2p^{-w} \cdot 2^w \cdot s_{(1-c)w} \cdot 2^w = \binom{n}{pn} \cdot 2 \cdot \left(\frac{4}{p}\right)^w \cdot s_{(1-c)w}.$$

Now when we take an expectation over W, the number of options for W is $\binom{n}{pn}$, so we get

$$\mathbb{E}_W \#\{S \mid (S, W) \text{ bad}\} = \frac{1}{\binom{n}{pn}} \#\{\text{bad } (S, W)\} \leq 2 \cdot \left(\frac{4}{p}\right)^w \cdot s_{(1-c)w}. \qquad \square$$

(The real thing that's going to be used here is that I told you lots of things about S by telling you $A \subseteq S$ and that it's large; this dwindles the number of options for S considerably.)

§22.6 Conclusion

We won't do the iterative process on the board (it's in the lecture notes), but we'll wave our hands and say something.

If we look at the parameters $s_1, \ldots, s_w$, if $s_k \leq \kappa^{-k} s_0$, then by Markov we get that

$$\mathbb{P}_W[\#\{S \in \mathcal{F} \mid (S, W) \text{ bad}\} \geq 0.1 s_0] \leq 2 \cdot \left(\frac{4}{p}\right)^w \cdot \frac{s_{(1-c)w}}{0.1 s_0} \lesssim \left(\frac{4}{\kappa^{1-c} p}\right)^w$$

is very small. And if you do things carefully, then with high probability (at least $1-q$, where q is the above thing), we can find a family $\mathcal{F}'$ such that $\varphi_{\mathcal{F}}|_{X_w=1} \geq \varphi_{\mathcal{F}'}$ (here $\mathcal{F}'$ is the result of restricting and then cleaning up the things that are bad), and that $\mathcal{F}'$ is almost as spread — specifically, $(\frac{1}{2}s_0; s_1, \ldots, s_{(1-c)w})$-spread and $\mathcal{F}' \subseteq \binom{[n]}{(1-c)w}$. So you get formulas with smaller and smaller width, but whose spreadness is virtually the same. So by iterating, we get such $\mathcal{F}'$ with width $(\log w)^{10}$ (for example). And then you can use a simple Chebyshev argument to finish. (By this we mean you can show just by Chebyshev that if your width is this small and you are spread, then you are 1 with probability close to 1; and because this is a lower bound, the original is also 1 with probability close to 1.) Making the iterations formal is a bit messy, but this is the main idea.

§22.7 Some remarks

This is a beautiful work (it's one of the best papers Dor knows that's very short — we basically saw the entire argument in one lecture), and following this, there were a bunch of developments in both TCS and combinatorics. Probably the most famous one is the Kahn–Kalai conjecture, which was proved by Park and Pham.

We won't say too much about this, but suppose you have some graph property $\mathcal{F}$ of containing a subgraph H. (Think of H as e.g. a cycle or a Hamiltonian cycle or something like that — something quite complicated.) If we want to compute the expected number of copies of H in a graph $G \sim \mathcal{G}(n,p)$, this is very easy — you just use linearity of expectation. But if we want to find the *critical probability* — the probability such that

$$\mathbb{P}_{G\sim\mathcal{G}(n,p)}[G \text{ contains a copy of } H] \geq 0.99,$$

this is often quite hard (much harder than the expectation).

What the conjecture says is that if you can find p where the expectation is at least 1, then the critical probability p might not be the same, but it is up to a logarithmic factor.

Now that we've talked about overlaps in clauses we might see there's some connection, but it took a few years between this proof and people realizing you could prove this conjecture using similar things.

§23 May 2, 2024

§23.1 Avoiding arithmetic progressions

For the next three lectures, we'll talk about the following problem.

Question 23.1. Fix $k \in \mathbb{N}$ with $k \geq 3$, and let p be a prime (sufficiently large with respect to k). How large can a set $A \subseteq \mathbb{F}_p^n$ be if it does not contain an arithmetic progression of length k (i.e., a set of the form $\{x, x+a, x+2a, \ldots, x+(k-1)a\}$ with $a \neq 0$)?

We'll abbreviate *k-term arithmetic progression* as k-AP.

So this is the type of problem we're going to study. We'll first make one comment — we're going to work in the finite-field setting, but it also makes sense to ask this question over $\mathbb{Z}$ (i.e., for sets $A \subseteq \{1, \ldots, n\}$). In fact, this is the more well-known setting. But for the arguments we'll see, there's not too much of a difference, except that things are much messier in $A \subseteq \{1, \ldots, n\}$ (instead of talking about subspaces we'd have to talk about Bohr sets, and we don't want to do that). So we'll work in the finite-field setting.

Theorem 23.2

If $A \subseteq \mathbb{F}_p^n$ has no k-APs, then A must have vanishing measure — i.e., $|A| = o_k(p^n)$.

Here's what the next few lectures will look like. Today we'll handle the case $k = 3$; we'll realize that to some extent, we've already done this twice in the course. Then in the next two lectures, we're going to handle the case $k = 4$, which is significantly more involved. Essentially we're going to see some very famous work of Gowers. If you work harder, you can extend this to higher k; but $k = 4$ is interesting enough.

§23.2 Fourier analysis over $\mathbb{F}_p^n$

We'll start with $k = 3$; this will be done with Fourier analysis. In this class we've done Fourier analysis over the cube and said some words about how it can be extended to other domains. And $\mathbb{F}_p^n$ is a product domain, so we have things like the Effron–Stein decomposition. But it turns out that whenever you have a Cayley group, you can do something more explicit and refined than Effron–Stein, and this is what we're going to do next.

Here's how it'll look like — let $\omega_p = e^{2\pi i/p}$ be a primitive pth root of unity of order p. Our first goal is to find the characters over $\mathbb{F}_p^n$. For that, it's enough to find characters over $\mathbb{F}_p$, and then tensorize them.

What would be a character over $\mathbb{F}_p$? If we just want a one-dimensional character over $\mathbb{F}_p$, for each $\alpha \in \mathbb{F}_p$, we can define the character $\chi_\alpha \colon \mathbb{F}_p \to \mathbb{C}$ by $\chi_\alpha(x) = \omega_p^{\alpha x}$. You can check that this is indeed a character; and if $p = 2$, then this fancy word 'primitive root of unity' just means -1, so this is exactly the same as what we've been doing.

And now if we want characters over $\mathbb{F}_p^n$, we just tensorize; so for each vector $\alpha \in \mathbb{F}_p^n$, we have a corresponding character $\chi_\alpha \colon \mathbb{F}_p^n \to \mathbb{C}$ defined as $\chi_\alpha(x) = \omega_p^{\langle \alpha, x\rangle} = \prod_{i=1}^n \chi_{\alpha_i}(x_i)$. So these are the characters.

Fact 23.3 — The characters χ_α form a basis for the set of complex-valued functions over $\mathbb{F}_p^n$.

The easiest way to do this is to define an inner product and show that this is in fact an orthonormal set (and since it has size p^n, that means it's a basis).

Definition 23.4. For $f, g \colon \mathbb{F}_p^n \to \mathbb{C}$, we define $\langle f, g\rangle = \mathbb{E}_x[f(x)\overline{g(x)}]$.

Unfortunately now we have to work with $\mathbb{C}$, so we'll have conjugation everywhere; but ignoring that, this is the same as what we've done up to now.

Then we can improve our fact to the one we'll actually prove:

Fact 23.5 — The characters χ_α form an orthonormal basis for $L_2(\mathbb{F}_p^n)$.

This will be left to us to check; it's not too hard.

And with this, we can finally say what is the Fourier decomposition — for each $f \colon \mathbb{F}_p^n \to \mathbb{C}$, there is a unique decomposition

$$f(x) = \sum_{\alpha \in \mathbb{F}_p^n} \widehat{f}(\alpha)\chi_\alpha(x),$$

where the Fourier coefficients $\widehat{f}(\alpha)$ are given as

$$\widehat{f}(\alpha) = \langle f, \chi_\alpha\rangle.$$

§23.3 Proof for $k = 3$

With this, we actually have all that we need to solve the case $k = 3$. Suppose that $A \subseteq \mathbb{F}_p^n$ contains no nontrivial 3-AP (the trivial k-AP is the case $a = 0$ in the definition of a k-AP; of course any set contains a trivial k-AP, and we're supposing A doesn't contain any others), and consider $1_A \colon \mathbb{F}_p^n \to \{0,1\}$. Then the condition that A contains no 3-APs, written more analytically, says that

$$1_A(x)1_A(x+a)1_A(x+2a) = 0$$

for all $x \in \mathbb{F}_p^n$ and $a \neq 0$. And we can average this over random uniform x and a to get that

$$\mathbb{E}_{x,a} 1_A(x)1_A(x+a)1_A(x+2a) = p^{-n} \cdot \mu(A)$$

(if $a \neq 0$ then we get no contribution; if $a = 0$ then we get 1 if and only if $x \in A$). So now we've translated this information into something that looks like the type of stuff we've studied in this course. And the goal will be to try to salvage some structure out of A using this information.

Our goal in life is to prove A is small; so if it's small, then we're done. If it's not small, we have something to do. And this lemma says that in that case, you have a large Fourier coefficient.

Lemma 23.6

Suppose that $A \subseteq \mathbb{F}_p^n$ contains no nontrivial 3-APs, and let $\alpha = \mu(A)$. Suppose that $\alpha \geq p^{-n/10}$. Then there exists $t \in \mathbb{F}_p^n$ with $t \neq 0$ and $|\widehat{1_A}(t)| \geq \frac{1}{2}\alpha^2$.

Proof. The proof is to plug in the Fourier transform into the above expression and hope for the best — we know that

$$p^{-n} \cdot \alpha = \mathbb{E}_{x,a}[1_A(x)1_A(x+a)1_A(x+2a)].$$

And now we're going to plug in the Fourier transform for 1_A — this is

$$\mathbb{E}_{x,a} \sum_{\alpha,\beta,\gamma} \widehat{1_A}(\alpha)\widehat{1_\beta}(\beta)\widehat{1_A}(\gamma)\chi_\alpha(x)\chi_\beta(x+a)\chi_\gamma(x+2a)$$

(where α, β, and γ are vectors corresponding to the Fourier coefficient we take from each term). The next step is to interchange the summation and expectation and then make sense of this expression — this is

$$\sum_{\alpha,\beta,\gamma} \widehat{1_A}(\alpha)\widehat{1_A}(\beta)\widehat{1_A}(\gamma)\mathbb{E}_{x,a}[\chi_{\alpha+\beta+\gamma}(x)\chi_{\beta+2\gamma}(a)]$$

(here we're observing that e.g., $\chi_\beta(x+a) = \chi_\beta(x)\chi_\beta(a)$, and $\chi_\alpha(x)\chi_\beta(x)\chi_\gamma(x) = \chi_{\alpha+\beta+\gamma}(x)$). And now we can split this expectation as a product of two expectations

$$\mathbb{E}_x \chi_{\alpha+\beta+\gamma}(x)\mathbb{E}_a \chi_{\beta+2\gamma}(a).$$

And so now we have an expectation of a character, and because the characters are an orthonormal basis, this expectation is either 0 or 1 — $\mathbb{E}_a \chi_{\beta+2\gamma}(a)$ is 1 if and only if $\beta + 2\gamma = 0$, and similarly the first expectation is 1 if and only if $\alpha + \beta + \gamma = 0$. Now we've got two equations in three variables, so we can solve in terms of γ — we get $\alpha = \gamma$ and $\beta = -2\gamma$, which gives

$$p^{-n}\alpha = \sum_\gamma \widehat{1_A}(\gamma)^2\widehat{1_A}(-2\gamma).$$

Next, we're going to separate out the case $\gamma = 0$ from the rest; this gives

$$\widehat{1_A}(0)^3 + \sum_{\gamma \neq 0} \widehat{1_A}(\gamma)^2\widehat{1_A}(-2\gamma).$$

The first term is α^3, because $\widehat{1_A}(0)$ is the constant character, and taking the inner product of 1_A with that gives just the measure.

This is why we need the condition that α isn't super tiny — we want to say that α^3 is much more dominant than $p^{-n}\alpha$. If we rearrange, we find that the sum of the other terms is actually quite negative — we get

$$\sum_{\gamma\neq 0}\widehat{1_A}(\gamma)^2\widehat{1_A}(-2\gamma) = p^{-n}\alpha - \alpha^3 \leq -\frac{1}{2}\alpha^3.$$

So this guy is quite negative, which means if we take absolute values it's quite positive — we get

$$\frac{1}{2}\alpha^3 \leq \left|\sum_{\gamma\neq 0}\widehat{1_A}(\gamma)^2\widehat{1_A}(-2\gamma)\right|.$$

And now we do our favorite thing in life, which is that we have something that looks almost like a sum of third powers of Fourier coefficients; first we do a triangle inequality and then pull out a maximum to get that this is at most

$$\sum_{\gamma\neq 0}|\widehat{1_A}(\gamma)|^2|\widehat{1_A}(-2\gamma)| \leq \max_{\gamma\neq 0}|\widehat{1_A}(\gamma)|\sum_{\gamma\neq 0}|\widehat{1_A}(\gamma)\widehat{1_A}(-2\gamma)|$$

(we could also pull out the -2γ instead and things would be slightly simpler, but we want to use Cauchy–Schwarz to get into the spirit of Gowers norms). Using Cauchy–Schwarz on this, we get that this is at most

$$\max_{\gamma\neq 0}|\widehat{1_A}(\gamma)|\cdot\left(\sum_{\gamma\neq 0}|\widehat{1_A}(\gamma)|^2\sum_{\gamma\neq 0}|\widehat{1_A}(-2\gamma)|^2\right)^{1/2}.$$

And we can use Parseval to say that each of these sums of squares is at most α. So we get that this is at most $\alpha\cdot\max_{\gamma\neq 0}|\widehat{1_A}(\gamma)|$, and since it's supposed to be at least $\frac{1}{2}\alpha^3$, we get the result. □

So this is the lemma; we've kind of seen this computation three times in the course, and this is another. But it turns out that once you've done this, the question for $k = 3$ is more or less finished.

Lemma 23.7

Under the same conditions, we can find a set $B \subseteq \mathbb{F}_p^{n-1}$ that contains no nontrivial 3-APs and such that $\mu(B) \geq \mu(A) + \frac{1}{2p}\alpha^2$.

The main point is that B is significantly *denser* than A.

We have to do two things — first we have to prove this lemma, and then we have to understand why it's good for us. Let's start with the second part. Once we have this lemma, why are we done? What this lemma is telling us is that if you have a set of dimension n which is 3-AP-free, then you can find a set in one less dimension which is denser and still 3-AP-free. So you can iterate — getting an even denser set in dimension $n-2$, and so on. If we iterate for $\frac{n}{2}$ steps, then we find a set $B' \subseteq \mathbb{F}_p^{n/2}$ which is 3-AP-free and has density something ridiculous, i.e.,

$$\mu(B') \geq \mu(A) + \frac{n}{2}\cdot\frac{1}{2p}\alpha^2.$$

And whatever B' is, its density is certainly at most 1; this gives

$$\frac{n}{2}\cdot\frac{\alpha^2}{2p} \leq 1,$$

which means $\alpha \lesssim \sqrt{p/n}$.

These sort of arguments are called *density-increment arguments* — if you have an object with some property, then you can find some different object which is denser and still has that property. This increases the density, and if you iterate then you can often get extremal results like this.

So now we have to prove the lemma.

Proof of lemma. Fix $t \neq 0$ from the previous lemma, so that $|\widehat{1_A}(t)| \geq \alpha^2/2$. This is telling us that $\mathbb{E}_x 1_A(x)\omega_p^{-\langle t,x\rangle}$ is large — so 1_A is correlated with a function only depending on $\langle t, x\rangle$. Now we're going to partition the space depending on what this inner product is, and show that for at least one of the choices, A has to be dense.

So for each $a \in \mathbb{F}_p$, let $p_a = \mathbb{E}_x[1_A(x) \mid \langle x,t\rangle = a]$. Then we know two things. On one hand, we know that $\frac{1}{p}\sum_{a\in\mathbb{F}_p} p_a = \alpha$. And next, what we're going to show is that there's some variance between these numbers — they cannot all literally be α. In other words, we want to show

$$\frac{1}{p}\sum_{a\in\mathbb{F}_p} |p_a - \alpha|$$

is somewhat large. First, by the triangle inequality

$$\frac{1}{p}\sum_{a\in\mathbb{F}_p} |p_a - \alpha| \geq \frac{1}{p}\sum_{a\in\mathbb{F}_p} \left|\omega^{-a}(p_a - \alpha)\right| = \frac{1}{p}\left|\sum_{a\in\mathbb{F}_p}\omega^{-a}p_a - \sum_{a\in\mathbb{F}_p}\omega^{-a}\alpha\right|.$$

The second sum is 0, because the sum of all roots of unity is 0; so then we're left with

$$\left|\frac{1}{p}\sum_{a\in\mathbb{F}_p}\omega^{-a}\mathbb{E}_x[1_A(x) \mid \langle x,t\rangle = a]\right|$$

(plugging in the definition of p_a). Now we want to massage this into looking something like a Fourier coefficient. If you sample random x, the probability that $\langle x,t\rangle = a$ is exactly $\frac{1}{p}$ (this is where we use the fact that $t \neq 0$ — if t were 0 this would always be 0, but otherwise it's uniform). So this is

$$\left|\sum_a \omega^{-a}\mathbb{E}_x 1_A(x)1_{\langle x,t\rangle = a}\right| = \left|\sum_a \mathbb{E}_x 1_A(x)\omega^{-\langle x,t\rangle}1_{\langle x,t\rangle = a}\right|.$$

And finally, we can push the summation over a inside, to get that the sum of all these indicators is 1; so this is just

$$\left|\mathbb{E}_x 1_A(x)\omega^{-\langle x,t\rangle}\right|,$$

which is exactly the Fourier coefficient that we assumed is at least $\alpha^2/2$. Collecting the two points of data that we have, we know that $\frac{1}{p}\sum p_a = \alpha$, and we know that $\frac{1}{p}\sum|p_a - \alpha| \geq \frac{\alpha^2}{2}$ (the average is α, and there's some variation). From this, we claim there exists a such that p_a is at least slightly larger than α — assume for contradiction that $p_a < \alpha + \alpha^2/2p$ for all a. Then using the fact that the average of the p_a's is α, this means $p_a > \alpha - \alpha^2/2$ for all α (if we had something smaller than that, then it dampens the sum $\sum p_a$ by at least $\alpha^2/2$; and the remaining things can't compensate for that, because each contributes at most $\alpha^2/2p$ and there's at most $p-1$ of them).

So if we assume all of them are not too large, then none of them can be too small; this means they're all camping around α, and we get that

$$\frac{1}{p}\sum_a |p_a - \alpha| < \frac{1}{p}\sum_a \frac{\alpha^2}{2} = \frac{\alpha^2}{2},$$

which is a contradiction.

So we've concluded there is some a such that p_a is significantly larger than α; fix a such that $p_a \geq \alpha + \alpha^2/2p$. Then if we look at

$$\mathcal{A} = \{x \in A \mid \langle x,t\rangle = \alpha\},$$

this is a set of one smaller dimension and it has larger density. It's a bit annoying that this lives in some affine space rather than a linear space; but we can shift it around and then we're done. Specifically, taking any $z \in \mathcal{A}$ and looking at

$$B = \{x - z \mid \langle x, t\rangle = a,\, x \in A\},$$

this is now a set in some linear space of one smaller dimension, and $\mu(B) = p_a \geq \alpha + \alpha^2/2p$ (and B also doesn't contain 3-APs). □

Remark 23.8. When we used the lemma to finish the proof we used a somewhat trivial bound on $\mu(B')$; can we improve things by using the fact that as we repeatedly apply the lemma, α gets bigger?

Suppose that $\alpha = 1/\sqrt{n}$; if we apply a small number of steps, α is still of the same order because we only gain $1/n$ each time. So there isn't a whole lot to gain here.

This is not a tight bound, though, and even using these tools you can get something stronger, though you need some ideas. But to make matters worse, there's a theorem from a few years ago, due to Croot–Lev—Pach and Ellenberg–Gijsuijt, that we have $\alpha \leq (1 - \varepsilon(p))^n$ (i.e., $\mu(A)$ is *exponentially* vanishing).

This blows this proof out of the water and uses none of these ideas — it uses the polynomial method. Still, there's lots of appeal in the Fourier-analytic approach, and we'll now talk about how to extend it to longer progressions (where nothing along the lines of the above result is known).

§23.4 A non-extension

This Fourier-analytic proof is quite nice, but it's also quite fragile. First, here's a variation of this theorem that's not known.

Definition 23.9. We say a *restricted* 3-*AP* is a set $\{x, x + a, x + 2a\}$ where $a \in \{0, 1\}^n$.

Question 23.10. What can we say about a set A not containing any restricted 3-AP — is it true that

$$\mu(A) \lesssim \frac{1}{\log\log\log n}?$$

Our proof breaks down — we had a part where we computed $\mathbb{E}\chi_{\beta+2\gamma}(a)$, but now a is only in $\{0, 1\}^n$ and this is no longer 0 or 1, but instead something complicated. So that proof fails.

And in fact, this is an open problem. It is known that $\mu(A)$ has to be vanishing, but the best bound is $1/\log^* n$ (and if you prove e.g. ten logs, then people would be happy).

§23.5 Larger values of k

Now that we've seen this proof, we might think that surely something like this works for k-APs. Suppose $A \subseteq \mathbb{F}_p^n$ contains no k-APs; then we can consider

$$\mathbb{E}_{x,a} 1_A(x) 1_A(x + a) \ldots 1_A(x + (k - 1)a) = p^{-n}\alpha,$$

and we might think you can apply some Fourier-analytic magic, write this in terms of Fourier coefficients, and be done. But that doesn't work. When you plug things in, you'll have k characters, but only 2 equations over them — so you'll have $k - 2$ free things, instead of just one. That's very unfortunate; so that doesn't work.

And indeed, the first proof that showed $\mu(A)$ has to be vanishing didn't use any Fourier analysis; it was done by Szemerédi, and it was completely combinatorial and very complicated. It took a few years, but then Tim Gowers came up with an actual Fourier-analytic approach that picks up where the previous proof left and manages to push it true.

§23.6 U^1 and U^2 Gowers uniformity norms

The story starts with the definition of two norms. The first is kind of silly.

> **Definition 23.11.** For $f\colon \mathbb{F}_p^n \to \mathbb{C}$, we define $\|f\|_{U^1} = (\mathbb{E}_{x,h} f(x)\overline{f(x+h)})^{1/2}$.

This is one of the more complicated ways to write $|\mathbb{E}f(x)|$, because x and $x+h$ are just independent.

This is a seminorm, and is not very interesting. But the U^2 norm is actually much more interesting — now you sample x and *two* h's.

> **Definition 23.12.** We define $\|f\|_{U^2} = (\mathbb{E}_{x,h,h'} f(x)\overline{f(x+h)f(x+h')}f(x+h+h'))^{1/4}$.

Where is this going? Well, U^2 is much more interesting than U^1 — if you plug in the Fourier transform of f and follow your nose, you're going to get that

$$\|f\|_{U^2}^4 = \sum_t |\widehat{f}(t)|^4.$$

So in some sense, what we saw in the proof of $k = 3$ is the combination of the following two facts:

- If A doesn't have a large 3-AP, then it has a large U^2 norm. (We didn't show this exactly, but it can be rephased in this way.)
- If a function has large U^2-norm, then we can get a density bound on some subspace.

Taking this more abstract view, you can hope that maybe this straightforwards Fourier-analytic approach doesn't work, but maybe there's some more convoluted way of showing that if our U^2 norm is small, then we do contain 4-APs.

In other words, we know that containing 3-APs is related to $\|1_A - \alpha\|_{U^2}$ being large/small (i.e., if A doesn't contain 3-APs, then this norm is actually large).

> **Question 23.13.** Does counting 4-APs relate to $\|1_A - \alpha\|_{U^2}$ in the same way — namely, is it the case that $\|1_A - \alpha\|_{U^2} \le \varepsilon$ implies that A contains roughly $\alpha^4 \cdot p^{2n}$ 4-APs?

(This number is just the number of 4-APs if A was completely random of the correct density.)

This is the starting point of Gowers's line of thought. The answer to this is complicated; the way that we've phrased it, the answer is no — there are sets with small U^2 norm where the number of 4-APs is not the right number. But in the example we'll see, it's actually *more* than the right number — it turns out that there are sets A with $\|1_A - \alpha\|_{U^2} = o(1)$ but at least $(\alpha^4 + \Omega(1))p^{2n}$ 4-APs. (An example is in the notes.)

This is why we said it's complicated — it may be the case that any set with small U^2 norm has *at least* the average number of 4-APs. If you construct a set with *fewer* than this number of 4-APs, that would be an interesting result.

But this tells you something fishy is going on — the previous proof showed you that if you had the small U^2 norm, then your count is genuinely close to the right number.

That's kind of a bummer; it means we wrote this complicated U^2 norm and we said it's good for 3-APs, but it's not going to work for 4-APs.

Question 23.14. Is there a different norm that captures 4-APs, and is 'useful'?

We won't formally define what 'useful' is; but we want it to facilitate some density-increment argument. (Of course you can define a weird norm like counting the number of 4-APs, but that won't get us anything.)

§23.7 More Gowers uniformity norms

Definition 23.15. Given $f\colon \mathbb{F}_p^n \to \mathbb{C}$ and a direction $h \in \mathbb{F}_p^n$, we define the *discrete multiplicative derivative* of f in direction h, denoted $\partial_h f\colon \mathbb{F}_p^n \to \mathbb{C}$, by

$$\partial_h f(x) = f(x)\overline{f(x+h)}.$$

Why would we call something like this a derivative? The reason is that if $f(x) = \omega_p^{q(x)}$ for some degree-d polynomial q, then if we pick some h and compute the derivative, we get

$$\partial_h f(x) = \omega_p^{q(x)-q(x+h)}.$$

And if you think of h as fixed, this is some polynomial $q_h(x)$ where q_h has degree at most $d-1$. So this is the reason these things are called discrete derivatives.

Now we can give a little more sense to what the U^1 and U^2 norms are. What U^1 checks is whether you are biased towards some direction — if the U^1 norm is large, then $\mathbb{E}f(x)$ is nonzero, so you're biased towards somewhere.

What does U^2 check? We can rewrite

$$\|f\|_{U^2}^4 = \mathbb{E}_h \|\partial_h f\|_{U^1}^2.$$

So we're taking the discrete derivative of f along some direction h, and checking whether *this* is biased. (We can open up the definitions to check that this is equivalent to the original one.)

The upshot of this business is that now in terms of common sense, we have some natural guess as to what other norms we can define — why just take one derivative and not more?

Definition 23.16. For $s \geq 2$, the U^s norm of $f\colon \mathbb{F}_p^n \to \mathbb{C}$ is defined as

$$\|f\|_{U^s}^{2^s} = \mathbb{E}_{h_1,\ldots,h_{s-1}} \|\partial_{h_1}\ldots\partial_{h_{s-1}} f\|_{U^1}^2.$$

So we take $s-1$ derivatives and check, after we apply all of them, whether we have a biased function. You can check that this is equivalent to taking $s-2$ derivatives and then checking whether you have a large U^2 norm, i.e.,

$$\mathbb{E}_{h_1,\ldots,h_{s-2}} \|\partial_{h_1}\ldots\partial_{h_{s-2}} f\|_{U^2}^4.$$

And if you want to spell out this thing as a product, it takes the somewhat hideous form

$$\mathbb{E}_{x,h_1,\ldots,h_{s-1}} \prod_{\alpha\in\{0,1\}^{s-1}} \mathcal{C}^{|\alpha|} f(x + \sum_{i=1}^{s-1} \alpha_i h_i),$$

where $\mathcal{C}$ is the complex conjugation operation (so $\mathcal{C}^{|\alpha|}$ means you conjugate the thing $|\alpha|$ times).

In the next few lectures, what we're going to see is that if you look at $s = 3$ (the U^3 norm), this is directly related to 4-APs (it captures how many 4-APs are in the set). That's not too hard; but then proving that it's useful is the more challenging task, and this will be very interesting.

§24 May 7, 2024

Last time, we started talking about arithmetic progressions over the finite field model, i.e., $\mathbb{F}_p^n$ for $p \geq 5$. We saw that if a set A is dense then it has an arithmetic progression of length 3, using Fourier coefficients. Fourier coefficients don't really work for arithmetic progressions of length 4, but we can hope that something does; this leads to the Gowers norms.

§24.1 Gowers norms

Definition 24.1. We define $\|f\|_{U^1} = |\mathbb{E}f| = |\mathbb{E}_h \mathbb{E}_x \partial_h f(x)|^{1/2}$.

Definition 24.2. We define $\|f\|_{U^2}^4 = \left|\mathbb{E}_{h,h'} \mathbb{E}_x f(x)\overline{f(x+h)f(x+h')}f(x+h+h')\right| = \mathbb{E}_h \|\partial_h f\|_{U^1}^2$.

The first formula is convenient when you want to expand, but it's not clear how to extend it beyond just Fourier coefficients. But in the second definition, it makes sense how to generalize to larger s.

Definition 24.3. We define $\|f\|_{U^s}^{2^s} = \mathbb{E}_{h_1,\ldots,h_{s-1}} \|\partial_{h_1,\ldots,h_{s-1}} f\|_{U^1}^2 = \mathbb{E}_{h_1,\ldots,h_{s-2}} \|\partial_{h_1,\ldots,h_{s-2}} f\|_{U^2}^4$.

This norm is quite nontrivial to study; we'll focus on the U^3 norm, which is the first norm not captured by Fourier coefficients.

When we take $s = 3$, what we're asking is, after we take a derivative, is f correlated with a linear function?

In general, there's a bunch of properties you can prove about these norms; one is monotonicity.

Fact 24.4 — We have $\|f\|_{U^1} \leq \|f\|_{U^2} \leq \|f\|_{U^3} \leq \cdots$.

This is not terribly difficult, but the easiest way to see it is by first arguing that $\|f\|_{U^1} \leq \|f\|_{U^2}$. The most direct way to see this is that you can actually write down a Fourier-analytic formula for $\|f\|_{U^2}^4$ — it's a sum of fourth powers of Fourier coefficients, so it's in particular at least $\widehat{f}(\{0\})^4$ (which is $\|f\|_{U^1}^4$). Another way to see this is Cauchy–Schwarz — we have

$$\|f\|_{U^1} = |\mathbb{E}_x \mathbb{E}_h \partial_h f(x)|^{1/2} \leq \left(\mathbb{E}_x |\mathbb{E}_h \partial_h f(x)|^2\right)^{1/4} = \left(\mathbb{E}_x \mathbb{E}_{h,h'} \partial_h f(x) \overline{\partial_{h'} f(x)}\right)^{1/4},$$

and if you write things out this becomes $\|f\|_{U^2}$. Then for the general case, we can write

$$\|f\|_{U^s} \geq \left(\mathbb{E}_{h_1,\ldots,h_{s-2}} \|\partial_{h_1,\ldots,h_{s-2}} f\|_{U^1}^2\right)^{2^{1-s}} = \|f\|_{U^{s-1}}.$$

(We use the definition with U^2, replace it with U^1 using the above fact, and then take out one of the squares using Jensen; this gives us the definition for U^{s-1} in terms of U^1.)

§24.2 Two questions

Question 24.5. Is the U^3 norm related to 4-APs?

For U^2, we saw that if $\|1_A - \mu(A)\|_{U^2}$ is small — meaning that your Fourier coefficients are small — then A contains 3-APs, and the count is roughly as it would be for a random set — namely $\mu(A)^3 \left|\mathbb{F}_p^n\right|^2$. (Last time we said things the other way — that if your count is off from the random thing, then you must have a large Fourier coefficient.)

And if we replace U^2 by U^3, we can wonder if this continues to be true for 4-APs. We'll see the answer is positive, and it's not that hard (it's several Cauchy–Schwarz's).

Question 24.6. If the U^3 norm of $1_A - \mu(A)$ is large, can we get a density increment?

With U^2, we saw that if you have a large Fourier coefficient, then you can consider a bunch of affine subspaces, and on one of them you'll be much denser; then you can restrict your problem to that.

The answer to this is also yes, but this will involve many Cauchy–Schwarzes and many other things.

§24.3 U3 and 4-APs

Lemma 24.7

Suppose that $A \subseteq \mathbb{F}_p^n$ has $\mu(A) = \alpha$, and suppose that $f = 1_A - \alpha$ satisfies $\|f\|_{U^3} \leq \varepsilon$. Then the 4-AP count in A is $(\alpha^4 + \Theta_p(\varepsilon^2))|\mathbb{F}_p^n|^2$.

This lemma will answer the first question.

Proof. First we'll write down an analytic expression for the count of 4-APs and see where life takes us — we can write down

$$\mathbb{E}_{x,a} 1_A(x) 1_A(x+a) 1_A(x+2a) 1_A(x+3a).$$

We know something about f, so we can try to write down this expression in terms of f — we replace each 1_A with $\alpha + f$, so this becomes

$$\mathbb{E}_{x,a}(\alpha + f(x))(\alpha + f(x+a))(\alpha + f(x+2a))(\alpha + f(x+3a)).$$

We're not going to expand this out, but you get α^4 plus 15 other terms. We'll show all the other terms are small. We won't actually do this for each of the terms individually; we'll just demonstrate this on

$$(*) = \mathbb{E}_{x,a} f(x) f(x+a) f(x+2a) f(x+3a),$$

which is the hardest term to deal with.

We're going to look at $|(*)|^2$. First, we're going to separate the expectations of x and a — so we get

$$|(*)|^2 = |\mathbb{E}_x f(x) \mathbb{E}_a f(x+a) f(x+2a) f(x+3a)|^2.$$

Now we do Cauchy–Schwarz, so this is at most

$$\mathbb{E}_x |\mathbb{E}_a f(x+a) f(x+2a) f(x+3a)|^2.$$

Now we unravel the square, so this is equal to

$$\mathbb{E}_x \mathbb{E}_{a,a'} f(x+a)\overline{f(x+a')} f(x+2a)\overline{f(x+2a')} f(x+3a)\overline{f(x+3a')}.$$

(Note that $\mathbb{E}|f(x)|^2 \leq 1$; we just drop that term when doing the Cauchy–Schwarz.)

We're going to use more Cauchy–Schwarz, so we'd like a way to write these terms more cleanly (since the number of terms is going to increase by a factor of 2 each time). Note that $f(x+a)\overline{f(x+a')} = \partial_{a'-a} f(x+a)$; and the second term is $\partial_{2(a'-a)} f(x+2a)$ and the third is $\partial_{3(a'-a)} f(x+3a)$. So let's change variables — we'll let $a' - a = h$, so this becomes

$$\mathbb{E}_{x,a,h} \partial_h f(x+a) \partial_{2h} f(x+2a) \partial_{3h} f(x+3a).$$

Now you can sort of see the pattern in this madness, which is very important to identify (or else you can't find your arms and legs). We started with things with four terms, and we ended up with pretty much the same-looking thing, except now we have derivatives and 3 terms. So the number of terms has decreased for us, but we paid for that with a derivative. Now we're going to do this again to end up with two terms and another derivative.

The first thing we'll do is we change variables to replace $x+a$ with x, so we get

$$\mathbb{E}_{x,a,h}\partial_h f(x)\partial_{2h}f(x+a)\partial_{3h}f(x+2a).$$

Now we have one term that doesn't depend on a, so we can do the same thing — this is

$$\mathbb{E}_{x,h}\partial_h f(x)\mathbb{E}_a\partial_{2h}f(x+a)\partial_{3h}f(x+2a).$$

And we can do Cauchy–Schwarz, so this is at most

$$\mathbb{E}_{x,h}\left|\mathbb{E}_a\partial_{2h}f(x+a)\partial_{3h}f(x+2a)\right|^2.$$

And now we can do the same thing again — we expand it out to get a and a', and collect again into derivatives, to get that this is

$$\left(\mathbb{E}_{x,h,h',a}\partial_{h'}\partial_{2h}f(x)\partial_{2h'}\partial_{3h}f(x+a)\right)^{1/2}.$$

Now x and $x+a$ are just independent, so we can write this as

$$\left(\mathbb{E}_{h,h'}\left(\mathbb{E}\partial_{h'}\partial_{2h}f\right)\left(\mathbb{E}\partial_{2h'}\partial_{3h}f\right)\right)^{1/2}.$$

Now we apply Cauchy–Schwarz again to get that this is at most

$$\left(\mathbb{E}_{h,h'}\left|\mathbb{E}\partial_{h'}\partial_{2h}f\right|^2\cdot\mathbb{E}_{h,h'}\left|\mathbb{E}\partial_{2h'}\partial_{3h}f\right|^2\right)^{1/4}.$$

And since $p\geq 5$ (this is what we need $p\geq 5$ for), the distribution of $2h$ is the same as that of h (and same with all the other terms), so we can write this as

$$\left(\mathbb{E}_{h,h'}\left|\mathbb{E}\partial_{h'}\partial_h f\right|^2\right)^{1/2}.$$

But this function has a name — where we take two derivatives and check whether the result is biased — and we get that this is $\|f\|_{U^3}^4$.

So we've bounded $|(*)|^2\leq\|f\|_{U^3}^4\leq\varepsilon^4$, which means $|(*)|\leq\varepsilon^2$. And you can show the same for each of the 15 terms, so you end up with an error term of $\Theta(\varepsilon^2)$. □

So we've answered the first question and showed that indeed the U^3 norm is relevant for 4-APs. And the proof made some sense — it's sensible to write out the count of 4-APs and replace 1_A with $\alpha+f$. What happens after you expand and get 15 error terms is sort of unclear, but you get used to it.

> **Remark 24.8.** By $\Theta(\varepsilon^2)$ we mean something with absolute value at most a constant times ε^2.

If you want to try to discover this for yourself, it makes sense to try to do something like this for U^2 (which should work and be simpler), and then try to generalize it.

§24.4 Functions with large Gowers norm

Now we move to the second question, which is the more interesting one.

Question 24.9. Suppose $p \geq 5$ and $f \colon \mathbb{F}_p^n \to \mathbb{C}$ is 1-bounded, and we know $\|f\|_{U^3} \geq \varepsilon$. What can we say about f?

You should think of f as $1_A - \alpha$ (this is always what it'll be).

First, here are some examples.

Example 24.10

Suppose that $f = \omega_p^{\sum \alpha_i x_i}$. Then after we take one derivative, $\partial_h f$ is a constant function — so f even has large U^2-norm, which in particular means it has large U^3-norm.

Example 24.11

Suppose that $f = \omega_p^{\sum \alpha_i x_i^2}$ (you could take any quadratic function). When we take one derivative, we get $\partial_h f = \omega_p^{2\sum \alpha_i h_i x_i + \sum \alpha_i h_i^2}$ (maybe up to some signs). In particular, this derivative is a linear function — we think of $\sum \alpha_i h_i^2$ as a constant. So $\|\partial_h f\|_{U^2}$ is large (it's $\Omega(1)$). And this is true for every h, so $\|f\|_{U^3} \geq \Omega(1)$ (in fact, it's just 1 or something like this).

More genearlly, any $\omega_p^{q(x)}$ where q is a quadratic polynomial works — you'd get $\partial_h \omega_p^{q(x)} = \omega_p^{2\langle h, \nabla q(x)\rangle + L(h)}$ where $L(h)$ is some function of just h, and the same thing happens.

So we see these two examples — f can be a linear function or a quadratic function.

Our goal is to show these are essentially the only things that can happen — if you have large U^3 norm, then you must be correlated with something like $\omega_p^{q(x)}$. That's most of the content of the second question.

§24.5 Correlation with a quadratic

How do we even get started? We know $\|f\|_{U^3}^8 \geq \varepsilon$ (it's really ε^8 in the previous notation, but we'll replace ε^8 with ε for convenience). And U^3 is very hard, but we can write U^3 as an average of U^2 norms of derivatives, and we do know how to say something about U^2 norms — so we can write

$$\varepsilon \leq \|f\|_{U^3}^8 = \mathbb{E}_h \|\partial_h f\|_{U^2}^4 .$$

And since f is 1-bounded, all of these things are at most 1; so by an averaging argument, for at least a $\frac{1}{2}\varepsilon$-fraction of h's, this U^2-norm is at least $\varepsilon/2$ — i.e.,

$$\|\partial_h f\|_{U^2}^4 \geq \frac{\varepsilon}{2}.$$

So that's good. Now we can return to the previous lecture — there we saw that if you have large U^2-norm, then it means there is a large Fourier coefficient. So there exists some $\varphi(h) \in \mathbb{F}_p^n$ such that $\partial_h f$ is correlated with the corresponding character — i.e.,

$$\left|\mathbb{E}_x \partial_h f(x) \omega_p^{\langle x, \varphi(x)\rangle}\right| \geq \frac{\varepsilon}{2}.$$

This is quite nice, and if we inspect the second example, here we see that $\varphi(h)$ will be $2h$. Right now $\varphi(h)$ is completely arbitrary, but this example suggests maybe φ has further structure.

The hope is that after you take a derivative you know you're correlated with a linear function; we'd like to 'integrate' this correlation and say before the derivative you're correlated with something quadratic. But right now there's no hope because these linear functions could be completely different. So our goal is going to be to argue that φ can't actually be completely arbitrary — it has to have some nice structure that will let us do this 'integration.'

§24.6 Establishing structure for φ

In other words, φ is a map $\mathbb{F}_p^n \to \mathbb{F}_p^n$ that takes the direction h of the derivative and returns the function you're correlated with. So far φ is completely arbitrary. Our goal will be to establish structure for it, in the sense that we want to say it looks like a *linear* function (of h) — because in the example it's $2h$, which is something linear.

Lemma 24.12

We have $\mathbb{P}_{x,h_1,h_2}[\varphi(x) - \varphi(x+h_1) - \varphi(x+h_2) + \varphi(x+h_1+h_2) = 0] \gtrsim \varepsilon^{12}$.

What is going on here? First, what happens if φ is actually linear?

Example 24.13

If $\varphi(h) = 2h$, then we get

$$2x - 2(x+h_1) - 2(x+h_2) + 2(x+h_1+h_2) = 0.$$

So in that case, the probability is 1.

And you can pretty easily show that if this probability is 1, then your function must be linear. So what this lemma is telling us is that at least weakly speaking, φ 'looks' linear. (That doesn't mean it's linear — there's still work to do — but that's why this lemma is a step in the right direction.)

Proof. The proof is by magic — it's going to be another series of Cauchy–Schwarzes. We start out with just spelling out what we know about φ — we know that

$$\mathbb{E}_x \partial_h f(x) \omega_p^{\langle \partial(h), x\rangle} \geq \frac{\varepsilon}{2}$$

for at least an $\varepsilon/2$-fraction of the h's, so averaging over h, we get that

$$\mathbb{E}_h \left|\mathbb{E}_x \partial_h f(x) \omega_p^{\langle \partial(h), x\rangle}\right|^2 \geq \frac{\varepsilon}{2}\left(\frac{\varepsilon}{2}\right)^2 \gtrsim \varepsilon^3.$$

Next, we expand out the inside, and we get that

$$\varepsilon^3 \lesssim \mathbb{E}_h \mathbb{E}_{x,x'} \partial_h f(x) \overline{\partial_h f(x')} \omega_p^{\langle \varphi(h), x-x'\rangle}.$$

We don't want to carry $x - x'$ around, so we'll call it z; then we'll replace x' by $x - z$. So we get that this is

$$\mathbb{E}_h \mathbb{E}_{x,z} \partial_h f(x) \overline{\partial_h f(x-z)} \omega_p^{\langle \varphi(h), z\rangle}.$$

Now we're going to do something unthinkable that doesn't make any sense; we're going to expand out the derivatives using the definition, so this is

$$\mathbb{E}_{h,x,z} f(x) \overline{f(x+h) f(x-z)} f(x+h-z) \omega_p^{\langle \varphi(h), z\rangle} \gtrsim \varepsilon^3.$$

What did we gain from this? We're only looking for information about φ; the rest of things are sort of floating around. So we'll only keep the expectation over h inside, and pull the rest outside; this becomes

$$\mathbb{E}_{x,z} f(x) \overline{f(x-z)} \mathbb{E}_h \overline{f(x+h)} f(x+h-z) \omega_p^{\langle \varphi(h), z\rangle} \gtrsim \varepsilon^3.$$

Now we're going to put absolute values on everything and bound the first two terms by 1 — we get

$$\mathbb{E}_{x,z} \left|\mathbb{E}_h \overline{f(x+h)} f(x+h-z) \omega_p^{\langle \varphi(h), z\rangle}\right| \gtrsim \varepsilon^3.$$

And now we're going to apply Cauchy–Schwarz again — this gives

$$\mathbb{E}_{x,z}\left|\mathbb{E}_h\overline{f(x+h)}f(x+h-z)\omega_p^{\langle\varphi(h),z\rangle}\right|^2 \gtrsim \varepsilon^6.$$

So far, it's not clear how we're progressing; but now it's going to make even less sense, and we'll define stuff that looks completely arbitrary. We define $B_z(h) = \omega_p^{-\langle\varphi(-h),z\rangle}$ (basically the last part, but we play around with it a bit to make it what we want) and $A_z(h) = \overline{f(h)}f(-z+h)$. Then our expression becomes

$$\mathbb{E}_{x,z}\left|\mathbb{E}_hA_z(x+h)\overline{B_z(-h)}\right|^2 \gtrsim \varepsilon^6.$$

What is this thing? This is the convolution $A_z * \overline{B_z}(x)$, because we're taking in two inputs that add up to x (and that's the definition of convolution). This was the whole point — to get some convolution-looking thing. (If you use standard Fourier analysis you might eventually get this, but this is the cleanest way to see it.)

And then since we had an expectation over x (which we fold into a 2-norm), we now get

$$\mathbb{E}_z\left\|A_z * \overline{B_z}\right\|_2^2 \gtrsim \varepsilon^6.$$

And the nice thing about convolutions is that we have Fourier-analytic formulas for them — so this gives

$$\mathbb{E}_z\sum_t \widehat{A_z}(t)\overline{\widehat{B_z}(t)}^2 \gtrsim \varepsilon^6.$$

Now we can use Cauchy–Schwarz to separate into fourth powers of the two terms; and the A-terms contribute at most 1, so we conclude that

$$\mathbb{E}_z\sqrt{\sum_t\left|\widehat{B_z}(t)\right|^4} \gtrsim \varepsilon^6.$$

Now we do Cauchy–Schwarz again to get rid of the square root, and we get that

$$\mathbb{E}_z\sum_t\left|B_z(t)\right|^4 \gtrsim \varepsilon^{12}.$$

Now we have sums of fourt powers of something, so that's our good old friend the U^2-norm — and we get that

$$\mathbb{E}_z\left\|B_z\right\|_{U^2}^4 \gtrsim \varepsilon^{12}.$$

(This is kind of a shocker — you do all this strange stuff and somehow get a U^2 norm.) And now we're going to spell out the definition of the U^2-norm and see what we get — this gives

$$\mathbb{E}_z\mathbb{E}_{x,h_1,h_2}\omega_p^{\langle z,\varphi(x)\rangle-\langle z,\varphi(x+h_1)\rangle-\langle z,\varphi(x+h_2)\rangle+\langle z,\varphi(x+h_1+h_2)\rangle} \gtrsim \varepsilon^{12}$$

(writing out the definition of B_z — which is getting evaluated at four points, all with the same z). Now we rearrange this to

$$\mathbb{E}_{x,h_1,h_2}\mathbb{E}_z\omega_p^{\langle z,\varphi(x)-\varphi(x+h_1)-\varphi(x+h_2)+\varphi(x+h_1+h_2)\rangle}.$$

And now if this big thing is not 0, then we've got some Fourier character we're averaging over all z, so the expectation over z is 0. This means the inner expectation is precisely the indicator of this mess being 0. And so the entire expectation is precisely the probability of this over all x, h_1, h_2, and we're done — explicitly,

$$\mathbb{E}_z\omega_p^{\langle z,\varphi(x)-\varphi(x+h_1)-\varphi(x+h_2)+\varphi(x+h_1+h_2)\rangle} = \begin{cases}1 & \text{if } \varphi(x)-\varphi(x+h_1)-\varphi(x+h_2)+\varphi(x+h_1+h_2)=0\\ 0 & \text{else,}\end{cases}$$

giving us what we want. □

§24.7 Extracting linearity

Now we have this lemma, and from that we hope to extract some information about φ being linear itself. First let's recall a problem we saw earlier in the course (we saw some version in the class, and another in PS1).

Lemma 24.14

If $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ satisfies $\mathbb{P}[f(x)+f(y)+f(z) = f(x+y+z)] \geq \frac{1}{2}+\delta$, then there exists s with $|\widehat{f}(s)| \gtrsim \sqrt{\delta}$.

This says that if f weakly resembles a linear function (in the above sense), then it has a large Fourier coefficient.

We also know $\frac{1}{2}+\delta$ is the weakest thing you can hope for, because a random thing will achieve $\frac{1}{2}$.

Here the story is similar, but there are two differences. For one thing, the output of φ is a vector rather than a bit, so $\frac{1}{2}$ is no longer trivial. What we want to show is once you're much more than a random-looking thing, you have correlation with a linear function in this sense. So more precisely, what we're going to show — probably next time:

Lemma 24.15

Let φ be as above. Then there exists $\psi\colon \mathbb{F}_p^n \to \mathbb{F}_p^n$ such that ψ is a homomorphism, and a shift $s \in \mathbb{F}_p^n$, such that

$$\mathbb{P}_x[\varphi(x) = \psi(x)+s] \geq p^{-1/\varepsilon^C}$$

(for some constant C).

So the original lemma — that if you weakly exhibit local linear behavior then you must globally exhibit linear behavior — still holds here with vector-valued functions, though with worse bounds.

This is highly nontrivial. Next time we'll spend quite a bit of time proving it. You might first try to do what we did in the pset problem, but that actually completely fails. Instead, in order to prove this you need tools from additive combinatorics. Here's one explanation as to why this is the case.

We can define the *graph* of φ to be the tuples $(x, \varphi(x))$ where $x \in \mathbb{F}_p^n$; and we can think of this as a subset of the group $\mathbb{F}_p^n \times \mathbb{F}_p^n$. If we take x, $x+h_1$, $x+h_2$, and $x+h_1+h_2$ with the earlier property, then we get a 4-tuple here that adds up to 0. So what we get is that the number of solutions to $a+b=c+d$ in Γ (the graph above) is at least $\varepsilon^{12}\,|\Gamma|^3$.

For *any* set Γ, the most solutions you could have is $|\Gamma|^3$ (once you fix a, b, and c, this determines d). And you can achieve this by taking Γ to be a subgroup (since if $a, b, c \in \Gamma$ then $a+b-c$ is also in Γ). Because of the $+$ and $-$, you could also have a coset (a shift of a subgroup). And it's not hard to see that this is the only way to get the exact maximum.

So it stands to reason that if the number of solutions is at least 1% of what it could be, then there should be some subgroup structure inside Γ. This turns out to be true, and to show it you need several nice tools from additive combinatorics. This is the core of the argument — arguing about sets that look like subgroups in some statistical sense, and showing structure for them.

§25 May 9, 2024

Last time, we made the observation that if $\|f\|_{U^3}^8 \geq \varepsilon$, then when we take a random derivative, it'll be correlated with a character; this lets us define a mapping $\varphi\colon \mathbb{F}_p^n \to \mathbb{F}_p^n$ from directions to characters telling us which one we're correlated with, such that

$$\mathbb{E}_h \left| \mathbb{E}_x \partial_h f(x) \omega_p^{\langle \varphi(h), x \rangle} \right| \gtrsim \varepsilon^3.$$

This is intuitive because if f is a quadratic phase, then this is indeed the case; but φ is not arbitrary but in fact a linear function. Now our goal is to show that any φ satisfying this must be kind of like a linear function. Towards this end, by a wave of Cauchy–Schwarz we proved the following claim:

> **Claim 25.1** — We have
>
> $$\mathbb{P}_{x,h_1,h_2}[\varphi(x) - \varphi(x+h_1) - \varphi(x+h_2) + \varphi(x+h_1+h_2) = 0] \gtrsim \varepsilon^{12}.$$

This is kind of like a variant of BLR we saw in a problem set, but this function outputs a vector rather than a bit, and our probability is ε^{12} rather than something greater than $\frac{1}{2}$, so this is pretty different. So our goal is to salvage some structure from φ from this; and for that, we'll need additive combinatorics.

§25.1 Additive combinatorics

We'll change perspective a bit from looking at φ to a subset of an abelian group — we define

$$\Gamma = \{(x, \varphi(x)) \mid x \in \mathbb{F}_p^n\},$$

and we think of Γ as a subset of the abelian group $G = \mathbb{F}_p^n \times \mathbb{F}_p^n$. Now we want to phrase the condition of the claim using Γ.

Observe that if we call $a = (x, \varphi(x))$, $b = (x + h_1, \varphi(x + h_1))$, $c = (x + h_2, \varphi(x + h_2))$, and $d = (x + h_1 + h_2, \varphi(x + h_1 + h_2))$, then we can look at the corresponding equation $a + d = b + c$. If we look at the first coordinate, both sides have two x's and two directions, so the equality always holds. Meanwhile, if we look at the second coordinate, this equation is satisfied precisely when the event in the claim occurs — so $a + d = b + c$ occurs if and only if x, h_1, and h_2 satisfy the event in the claim. And this means

$$\#\{a + d = b + c \mid a, b, c, d \in \Gamma\} \gtrsim \varepsilon^{12} \left| \mathbb{F}_p^n \right|^3 = \varepsilon^{12} \left| \Gamma \right|^3.$$

We call the left-hand side the *additive energy* of Γ. If we look at this statistic, we can see it's always at most $|\Gamma|^3$ — once we pick a, d, and b, there's only one choice for c. So this tells you that the additive energy of Γ is at least e.g. 1% of the most it can be, and from this we're going to derive stuff.

§25.1.1 Balogh–Szemerédi–Gowers

Now we come to quite a major lemma.

> **Lemma 25.2** (Balog–Szemerédi–Gowers)
>
> Suppose G is an abelian group, and $\Gamma \subseteq G$ is such that the additive energy of Γ is at least $\eta \cdot |\Gamma|^3$. Then there exists $\Gamma' \subseteq \Gamma$ such that:
>
> (1) Γ' is sizeable — specifically, $|\Gamma'| \gtrsim \eta \, |\Gamma|$.
>
> (2) Γ' is an 'approximate subgroup' — specifically, $|\Gamma' + \Gamma'| \lesssim \eta^{-26} \, |\Gamma'|$

We use $|\Gamma' + \Gamma'|$ to denote $\{a + b \mid a, b \in \Gamma'\}$. What does this mean? Ignoring the power of η you get, if a set is an approximate subgroup in this sense, then its additive energy is going to be large — there's $|\Gamma|^2$ possible sums $a + b$, so if the number of total sums isn't large, then every sum is achieved many times. It's not hard to see that if a set is an approximate subgroup, then it has large additive energy. This lemma goes the other way — if you know Γ has large additive energy (which could just arise from a little structure in Γ), then you can in fact find this little structure, and it looks like an approximate subgroup.

> **Remark 25.3.** This statement was first proven by Balogh and Szemerédi, but not with these bounds — they used the Szemerédi regularity lemma, resulting in terrible tower-type bounds. But Gowers in his work on 4-APs couldn't afford these bounds, so he found a new proof getting much better (i.e., polynomial) parameters. (This is why the names in this lemma aren't alphabetical.)

The proof is surprising because it has hardly anything to do with groups — it's more about combinatorics.

To prove this, let Γ be as in the lemma. We know that the equation $a+d = b+c$ has at least $\eta\left|\Gamma\right|^3$ solutions; if we rearrange this, we get that $a - b = c - d$ also has at least $\eta\left|\Gamma\right|^3$ solutions.

But there are at most $\left|\Gamma\right|^2$ options for $a - b$ (we can choose a and then b). So this means there exist at least $\frac{1}{2}\eta\left|\Gamma\right|$ differences x such that $a - b = x$ has at least $\frac{1}{2}\eta\left|\Gamma\right|$ solutions. (What this is telling you is that there are many pairs that give you the same difference; and what we're saying is then we can pick a large collection of common differences which are pretty popular, meaning that the equation $a - b = x$ has lots of solutions.)

Now let $P = \{x \mid a - b = x \text{ has at least } \frac{\eta}{2}\left|\Gamma\right| \text{ solutions}\}$ be the set of popular differences. Now we're going to construct a bipartite graph H whose sides are each Γ, and whose edges correspond to P — so for $a \in \Gamma$ on the left and $b \in \Gamma$ on the right, we draw the edge ab if $a - b \in P$. In other words, $H = (\Gamma \cup \Gamma, E)$ where $E = \{(a, b) \mid a, b \in \Gamma,\ a - b \in P\}$. Right now, the only thing we know about this graph is that it has many edges — specifically, $\left|E\right| \geq (\frac{\eta}{2}\left|\Gamma\right|)^2 = \frac{\eta^2}{4}\left|\Gamma\right|^2$ (we have at least $\frac{1}{2}\eta\left|\Gamma\right|$ edges, and each gives at least $\frac{1}{2}\eta\left|\Gamma\right|$). We think of η as a constant, so the point is that this is a dense graph — the number of edges in this graph is a constant portion of what it could be. (The exact constants are not super important.)

This is pretty much all that we know about H, and using it, we're going to make some interesting conclusions.

> **Claim 25.4** — Let $H = (V \cup W, E)$ be a graph with N vertices on each side and $\left|E\right| \geq \alpha N^2$. Then for every $\xi > 0$, there exists $V' \subseteq V$ such that:
>
> (1) V' is sizeable — specifically, $\left|V'\right| \geq \frac{\alpha}{2}N$.
>
> (2) For at least $(1 - \xi)\left|V'\right|^2$ of the pairs $(v_1, v_2) \in V'$, there are at least $\frac{\xi\alpha^3}{16}N$ paths of length 2 between v_1 and v_2.

So what this claim is telling us is that if H is dense — meaning the average degree of all vertices is at least αN — then there exists a subset $V' \subseteq V$ which is very 'well-connected,' in the sense that there are many paths of length 2 between any two vertices.

In other words, we have a bipartite graph with the same number of vertices on the left and right, and we know it's fairly dense. Then what you can say is that if you look at the left side, there is a sizeable set in it that's very well-connected — the number of paths between two vertices in it is almost always a constant fraction of the maximum possible number.

Proof. We'll do this by *dependent random choice.* Often when you want to prove existence of something, it makes sense to sample at random and show that it works. But if you try to choose V' randomly, this is not going to work — it's not going to favor sets that are well-connected. So the idea is to choose V' in a random way, but in a *dependent* manner.

We use $\mathcal{N}(v)$ to denote the neighborhood of v, i.e., $\{w \in W \mid (v, w) \in E\}$. We say that v_1 and v_2 are *friendly* if $\left|\mathcal{N}(v_1) \cap \mathcal{N}(v_2)\right| \geq \frac{\varepsilon\alpha^3}{16}N$, and *unfriendly* otherwise. In the language of the claim, the statement that v_1 and v_2 are friendly translates to there being lots of length-2 paths between them — any common neighbor gives you a path of length 2, so the intersection of the common neighborhoods is precisely the number of length-2 paths (which is why this definition makes sense).

Now we choose $w \in W$ uniformly, and pick $V' = \mathcal{N}(w)$ — so we pick one guy on the right, and take all its neighbors.

First let's compute $\mathbb{E}_w |V'|$ — we have $\mathbb{E}_w |V'| = \frac{1}{N} \sum_w d(w) = \frac{1}{n} |E| = \alpha N$ (because with probability $\frac{1}{N}$ the size is $d(w)$). So in expectation, V' is quite significant in size.

Now spupose we fix v_1 and v_2 which are unfriendly; what's the probability that the w we picked is a common neighbor of them? This is

$$\mathbb{E}_w 1_{w \in \mathcal{N}(v_1) \cap \mathcal{N}(v_2)} \leq \frac{\xi \alpha^3}{16}.$$

Now we'll combine these two facts — let Z be the number of unfriendly pairs in V'. Then

$$\mathbb{E}_w Z = \mathbb{E}_w \sum_{(v_1, v_2) \text{ unfriendly}} 1_{w \in \mathcal{N}(v_1) \cap \mathcal{N}(v_2)} \leq \frac{\xi \alpha^3}{16} N^2$$

(there are at most N^2 unfriendly pairs).

Now let's consider two events — we have

$$\mathbb{P}\left[|V'| \geq \frac{\alpha}{2} N\right] \geq \frac{\alpha}{2}$$

(because $|V'|$ is always at most N, so if this proability were less than $\frac{\alpha}{2}$ then $\mathbb{E}|V'|$ would be too small). But Z has small expectation, so the probability it's large is small by Markov — specifically,

$$\mathbb{P}\left[Z \geq \frac{\xi \alpha^2}{4} N^2\right] \leq \frac{\alpha}{4}.$$

And $\frac{\alpha}{4} < \frac{\alpha}{2}$, so with positive probability the first event happens and the second doesn't — in particular, this means there exists w such that $|V'| \geq \frac{\alpha}{2} N$ and $Z \leq \xi (\frac{\alpha N}{2})^2 \leq \xi |V'|^2$, which is what we wanted. □

Remark 25.5. Part of the reason Dor is showing us this is these tricks like dependent random choice are very powerful and used all over the place; if you know how to apply these things well you can do a lot of stuff.

So this claim is nice — it tells us we can find a set on the left side such that nearly all pairs of vertices have many paths of length 2. The following claim will be a strengthening of this.

Claim 25.6 — Let $H = (V \cup W, E)$ be a bipartite graph with N vertices on each side and $|E| \geq \alpha N^2$. Then there are subsets $V' \subseteq V$ and $W' \subseteq W$ such that:

- Both are sizeable — $|V'|, |W'| \geq \frac{\alpha^2}{32} N$.
- For all $v \in V'$ and $w \in W'$, there are at least $\Omega(\alpha^7 N^2)$ paths of length 3 from v to w.

If we pick any two vertices, the number of paths of length 3 is at most N^2. So this says that for every pair of vertices in the two sets, there's a comparable number to the maximum. The main thing we gained going from the first claim to the second is that 'almost all' turned into 'all' (and the cost of that is increasing the length of the path).

Proof. The proof of this is not super interesting — it's mainly applying the previous claim, with some annoyances — but we'll do it anyway.

First, we remove from H all edges (v, w) where $d(v) \leq \frac{\alpha}{2} N$. This operation has two effects. First, the number of edges in H may decrease; but the number of edges we have left is at least $\frac{\alpha}{2} N^2$ — the worst-case scenario is if all the v's had this degree, and we could remove at most $\frac{\alpha}{2} N \cdot N$ many edges, but we had at least αN^2 to start with. And secondly, for every v, either $d(v) = 0$ or $d(v) \geq \frac{\alpha}{2} N$.

Now we still have a dense graph, so we can apply the previous claim to find $V' \subseteq V$ of size at least $\frac{\alpha}{4}N$ such that for at least a $(1-\xi)$-fraction of pairs in it, there are at least $\Omega(\xi\alpha^3 N)$ paths of length 2 between them. (Here ξ is a parameter that we'll choose later — we'll take it to be $\alpha^2/128$.)

Now we know that for most of the pairs in V', we have many paths of length 2. Now we need to do two things. The first of them is that we don't really want degrees to be 0 inside V', and to be honest it doesn't make sense to take things of degree 0 in V', so we just throw them away — we discard from V' all vertices of degree 0. This doesn't really affect the size of V' too much (at most a ξ-fraction of vertices in V' could have degree 0).

Secondly, we know the total number of pairs in V' that are friendly is at least $1-\xi$, so if we take one guy in V', it'll in expectation be friendly with at least $1-\xi$ of the other things. So we say $v \in V'$ is *nice* if for at least $(1-2\xi)$ of $\widetilde{v} \in V'$, the pair $(v, \widetilde{v})$ is friendly. By an averaging argument, it follows that at least half of the vertices $v \in V'$ are nice — so if we take $V'' = \{v \in V' \mid v \text{ is nice}\}$, we get that

$$\left|V''\right| \geq \frac{1}{2}\left|V'\right| \geq \frac{\alpha}{8}N.$$

(This is one of the steps where we gain things, because now *everything* is nice.)

We're going to take V'' to be the V' in the statement of the lemma; but now we need to construct W'. We want many paths of length 3 between V'' and W', so it makes sense to take W' to consist of vertices with many edges to V''. This is what we're going to do — for $w \in W$, let $d_{V''}(w) = |\mathcal{N}(w) \cap V''|$ be the number of neighbors of w in V''. Then

$$\mathbb{E}_w d_{V''}(w) \geq \frac{|V''|\,\frac{\alpha}{2}N}{N},$$

because all vertices in V'' have degree at least $\frac{\alpha}{2}N$, so if we look at all the edges adjacent to V'', the number of such edges is at least

$$\mathbb{E}_w d_{V''}(w) \geq \left|V''\right| \cdot \frac{\alpha}{2}N = \frac{\alpha}{2}\left|V''\right| \geq \frac{\alpha^2}{16}N.$$

(There's many edges that go outside, and if we choose w at random it'll catch at least a $\frac{1}{N}$-fraction of them.) So by an averaging argument, letting

$$W' = \left\{w \in V \mid d_{V''}(w) \geq \frac{\alpha^2}{32}N\right\},$$

we have $|W'| \geq \frac{\alpha^2}{32}N$.

Now we'll prove that V'' and W' have the property we want. First, the first item says that V'' and W' should be sizeable; this is true. Secondly, we want to argue that between any two vertices $v \in V''$ and $w \in W'$, there are many paths of length 3. For this, let's suppose we fix $v \in V''$ and $w \in W'$. The idea is that anything in W' has lots of neighbors in V'', and v is nice, so it's friendly to nearly everything in V''. And w in fact has so many neighbors that because v is nice, almost all of them are friendly with v.

Explicitly, let $v_1, \ldots, v_k \in V''$ be the neighbors of w, so that $k \geq \frac{\alpha^2}{32}N$. Since v is nice, (v, v_i) are friendly for at least $k - 2\xi V' \geq (\frac{\alpha^2}{32} - 2\xi)N$ values of i. This is the only place where the choice of ξ matters — we choose $\xi = \frac{\alpha^2}{128}$ so that 2ξ is small, and so we get at least $\frac{\alpha^2}{64}N$ neighbors of w which are friendly to v.

Now for such v_i, there's many common neighbors between v and v_i, and an edge between v_i and w. So we get that the number of paths of length 3 from v to w is at least

$$\frac{\alpha^2}{64}N \cdot \frac{1}{16}\xi\alpha^3 N = \Omega(\alpha^7 N^2)$$

(the number of vertices v is friendly with in $\mathcal{N}(w)$, times the number of paths of length 2 between v and something it's friendly with). □

The combinatorics part is now over; it's not clear why all of this is relevant to the discussion yet. So let's now see why this is useful.

Proof of Balogh–Szemerédi–Gowers. This discussion began with associating with Γ a graph — so let's consider the graph $H = (\Gamma \cup \Gamma, E)$ where $E = \{(a, b) \mid a - b \in P\}$. We saw that $|E| \geq \frac{\eta^2}{4} |\Gamma|^2$ is sizeable; here $\eta^2/4$ is going to be our α. And then by the second claim, we can find $A \subseteq \Gamma$ on the left and $B \subseteq \Gamma$ on the right such that $|A|, |B| \geq \frac{\alpha^2}{32} N$ and for all $a \in A$ and $b \in B$, the number of paths of length 3 from a to b is at least $\alpha^7 N^2$.

Let's fix a and b, and let's consider a path of length 3 and see what it tells us. Suppose we have a path (a, w, v, b); the edges correspond to differences, so we let $x = a - w$, $y = v - w$, and $z = v - b$. What's arithmetically interesting about this graph is that

$$x - y + z = a - b$$

is something that doesn't depend on the path, only its endpoints. So what we get is that every path of length 3 from a to b corresponds to writing $a - b$ in some way; and the fact that there are many paths of length 3 means there are many representations of $a - b$ in this form.

Now let's go one step further. There's a good reason (a, w) is an edge — this is because the difference x is popular. So now we're going to express each one of these terms in terms of differences in Γ — if we look at the equation

$$(\gamma_1 - \gamma_2) - (\gamma_3 - \gamma_4) + (\gamma_5 - \gamma_6) = a - b$$

over $\gamma_1, \ldots, \gamma_6 \in \Gamma$, then we claim there are many solutions in Γ — ignoring constants, we get $\alpha^7 N^2$ for the number of x, y, and z solving the equation; and then each of those differences are popular, so each corresponds to $\frac{\alpha}{2} N$ many pairs; this gives

$$\#\text{solutions} \gtrsim \alpha^7 N^2 \cdot \left(\frac{\alpha}{2} N\right)^3 \gtrsim \alpha^{10} N^5.$$

Now this is very interesting — we get that for all distinct differences $a - b$, there are a gazillion solutions to this equation. But how many possible things can you even represent as $(\gamma_1 - \gamma_2) - (\gamma_3 - \gamma_4) + (\gamma_5 - \gamma_6)$? There's 6 things we have to choose, so there's at most $|\Gamma|^6$ distinct differences $a - b$ that can be expressed in this way.

And each difference has a gazillion number of options to represent it, but there's at most this many representations whatsoever. So the number of distinct differences has to be at most

$$|A - B| \lesssim \frac{|\Gamma|^6}{\alpha^{10} N^5} \lesssim \alpha^{-10} |\Gamma|$$

(here $|\Gamma|$ and N are the same thing).

Now we're almost done. We promised that $|A + A|$ was going to be small (or maybe $|A - A|$). But right now we have two sets, and we want just A itself. So now we are going to use a fact which we don't really have enough time to prove (the proof is in the notes).

Fact 25.7 (Rusza's triangle inequality) — If $A, B, C \subseteq G$ (where G is some abelian group), then

$$|A| |B - C| \leq |A - B| |A - C|.$$

The proof of this is one or two paragraphs, but we'll skip it.

Then using this inequality in the right way, we get $|B| |A - A| \leq |A - B|^2$, which tells us that

$$|A - A| \leq \frac{|A - B|^2}{|B|} \lesssim \alpha^{-22} |\Gamma| \lesssim \alpha^{-24} |A|$$

(using the upper bound on $A - B$ and lower bounds on A and B).

Modulo the fact that we promised $A + A$ and instead got $A - A$, this is what we wanted (in fact $A - A$ may be more convenient to work with). □

In the final lecture, we'll use this approximate subgroup to construct a linear function that agrees with our φ.

§26 May 14, 2024

§26.1 Review

Last time, we proved the following:

Lemma 26.1

Suppose $\Gamma \subseteq G$ has significant additive energy — i.e., at least $\eta\,|\Gamma|^3$. Then there is some sizeable $\Gamma' \subseteq \Gamma$ which is nearly a coset — meaning that $|\Gamma'| \gtrsim \eta\,|\Gamma|$ and $|\Gamma' - \Gamma'| \lesssim \eta^{-52}\,|\Gamma'|$.

There's another fact, called Plünnecke's inequality — if your set behaves kind of like a coset with respect to *one* addition, then it continues to do so with respect to many.

Fact 26.2 (Plünnecke's inequality) — Suppose that $\Gamma \subseteq G$ and $|\Gamma - \Gamma| \le k\,|\Gamma|$. Then $|m\Gamma - \ell\Gamma| \le k^{m+\ell}\,|\Gamma|$.

By $m\Gamma$ we mean $\Gamma + \cdots + \Gamma$ (rather than $\{m\gamma \mid \gamma \in \Gamma\}$).

The proof is quite interesting, but we won't show it.

§26.2 Freiman homomorphisms

How are we going to use this? Now comes a very nice result, due to Gowers.

Definition 26.3. Let $\varphi\colon A \to G$ (where $A \subseteq G$). We say φ is a *Freiman homomorphism of order k* if for any $a_1, \dots, a_{2k} \in A$ such that $a_1 + \cdots + a_k = a_{k+1} + \cdots + a_{2k}$, we have $\varphi(a_1) + \cdots + \varphi(a_k) = \varphi(a_{k+1}) + \cdots + \varphi(a_{2k})$.

The way to think about this is — imagine that A is all of G. If φ is a Freiman homomorphism of order k (in fact, even if $k = 2$), then in fact φ is just a homomorphism. So what this definition means is it relaxes to just a subset of the input space; but as far as linearity is concerned, you *look* like a homomorphism as long as the summations we consider are not too large.

Now we're going to combine the facts above to get the following, through another nice argument of Gowers. Recall that φ tells us, for each direction, which character we are correlated with.

Lemma 26.4 (Gowers)

Suppose that $\varphi\colon \mathbb{F}_p^n \to \mathbb{F}_p^n$ satisfies

$$\mathbb{P}_{x,h_1,h_2}[\varphi(x) - \varphi(x+h_1) - \varphi(x+h_2) + \varphi(x+h_1+h_2) = 0] \ge \eta.$$

Then there exists a subset $A \subseteq \mathbb{F}_p^n$ with $\mu(A) \gtrsim \eta^{836}$ such that $\varphi|_A$ is a Freiman homomorphism of order 4.

We started off with the statement that φ looks weakly linear in the sense here. And we get the amazing conclusion that from this weak linearity-looking condition, we actually find a significant subset of inputs where you totally look linear.

> **Remark 26.5.** This looks like a linearity test that we studied 20 lectures ago, but it's different in that the output is a vector and this number is much smaller than $\frac{1}{2}$. You can try the original proof, but we don't know how to make that work. Instead, the last few lectures we developed a bunch of additive combinatorics, and now we can actually make this nice conclusion.

Proof. We first define the graph of φ — i.e., $\Gamma = \{(x, \varphi(x)) \mid x \in \mathbb{F}_p^n\}$. Then the given condition means the additive energy of Γ is at least $\eta \left|\Gamma\right|^3$. So by the above lemma, we can find $\Gamma' \subseteq \Gamma$ which is sizeable — $|\Gamma'| \geq \eta \left|\Gamma\right|$ — and with $|\Gamma' - \Gamma'| \lesssim \eta^{-52} \left|\Gamma'\right|$. Now we're going to apply Plünnecke's inequality — by Plünnecke's inequality, we get that $|8\Gamma' - 8\Gamma'| \leq k \left|\Gamma'\right|$, where $k \lesssim \eta^{-16\cdot 52}$. (This is close to the 836 number, but we'll lose a bit more.)

This is where the cleverness begins (so far, we've only used black-box stuff). Define

$$Y = \{y \mid (0, y) \in 4\Gamma' - 4\Gamma'\}.$$

(So you add Γ' to itself 4 times and then subtract; we know by Plünnecke's that there aren't too many such tuples.)

> **Claim 26.6** — We have $|Y| \leq k$.

Proof. The proof is by contradiction — if $|Y| > k$, then you can take a bunch of things from Γ' itself, add these to them, and get stuff in $8\Gamma' - 8\Gamma'$. You only have to make sure that these things are all distinct, but we can do that.

So assume for contradiction that $|Y| > k$, and take $|\Gamma'|$ tuples $(x_i, y_i) \in \Gamma'$ with distinct x_i's. (We know Γ is the graph of a function, so for every x there is only one y; and Γ' is some subset of Γ, so we can certainly pick tuples like this.)

Now let's fix another arbitrary $(a, b) \in \Gamma'$. Note that

$$(x_i - a, y_i + y - b) \in 8\Gamma' - 8\Gamma'$$

for all i and for all $y \in Y$ – this is because $(x_i - a, y_i - b) \in \Gamma' - \Gamma'$ and $(0, y) \in 4\Gamma' - 4\Gamma'$, and $(0, 0) \in 3\Gamma' - 3\Gamma'$. And when you add all these up you get this point.

But when we range over all i and all y, we get distinct points — because if we pick two different i's then the x-coordinates are different, but if we pick the same i but different y then the y-coordinates are different. So this gives you that $|8\Gamma' - 8\Gamma'| \geq |Y| \left|\Gamma'\right| > k \left|\Gamma'\right|$, which is a contradiction. □

So now we know Y is not very large. And we're going to use this fact to claim that we can find a large subspace which intersects Y trivially. Of course Y will always contain the 0-point — and any subspace we pick is going to contain 0. So what we're going to show is there's a subspace of small codimension for which this is the *only* intersection.

> **Claim 26.7** — There exists a subspace $W \subseteq \mathbb{F}_p^n$ with $\operatorname{codim}(W) \leq \log_p 2k$ such that $W \cap Y = \{0\}$.

Proof. In fact, the only information we need to use about Y is the fact that it's not very large.

When in doubt, we can try to do things randomly — let $r = \log_p 2k$, and pick $W \subseteq \mathbb{F}_p^n$ of codimension r randomly. Now we can compute $\mathbb{E}\,|W \cap (Y \setminus \{0\})|$. By linearity of expectation, this is

$$\sum_{y \in Y \setminus \{0\}} \mathbb{P}_W[y \in W].$$

And what's the probability that a given point y is in W? It's at most p^{-r} (roughly the relative size of W); and we have a sum of numbers, so this is at most $|Y|\, p^{-r} = \frac{1}{2} < 1$. And because the expectation is less than 1, we can certainly find some W for which it's less than 1, meaning it must be 0. □

Now is the last step of the proof — for each point $a \in \mathbb{F}_p^n$, define

$$\Gamma'_a = \{(x, y) \in \Gamma' \mid y \in a + W\}.$$

The point is that subspaces are closed under addition, and we want a set of points where you're a Freiman homomorphism. And because Γ' is sort of additive, if we have three points in it, and we sum two and subtract the last one, then if all of them were in $a + W$, the last one should also be. So it makes sense sets like this should give us something where we look like a Freiman homomorphism; and this indeed is the case.

Claim 26.8 — There exists a such that $|\Gamma'_a| \geq \frac{1}{2k} |\Gamma'|$.

Proof. Let's sample a randomly and calculate the expected size of Γ'_a — this is

$$\mathbb{E}_a\,|\Gamma'_a| = \sum_{(x,y)\in\Gamma'} \mathbb{P}_a[y \in a + W] = \sum_{(x,y)\in\Gamma'} \mathbb{P}_a[-a \in -y + W].$$

But now if you look at $-y + W$, this is a set of size $p^{-r}\left|\mathbb{F}_p^n\right|$; and a is uniform, so the probability that a is one of them is exactly $p^{-r} \geq \frac{1}{2k}$. So we get $\mathbb{E}_a\,|\Gamma'_a| \geq \frac{1}{2k}|\Gamma'|$, and we can find at least one a achieving the expectation. □

This $k\,|\Gamma'|$ is what's going to be the η^{-836}; that's all we lose, and now we're done.

Fix a as in the claim, and let $A = \{x \mid \exists y \text{s.t.} (x, y) \in \Gamma'_a\}$. So we've found some shift of W that captures lots of Γ', and we claim that if we look at these x's, then $\gamma|_A$ a Freiman homomorphism of order 4. (You can actually get whatever order you want; but you have to fudge the other numbers appropriately.)

To see why this is true, let $x_1, \ldots, x_8 \in A$ such that $x_1 + \cdots + x_4 = x_5 + \cdots + x_8$. What does the information that all of them are in A give us? It tells us there is some y with $(x, y) \in \Gamma'_a$. But there's only one y which is even a candidate, meaning $\varphi(x)$. So we get that $\varphi(x_i) \in a + W$ for all i. And therefore we get

$$\varphi(x_1) + \cdots + \varphi(x_4) - \varphi(x_5) - \cdots - \varphi(x_8) \in W.$$

But now let's call the left-hand side y. What's another thing we can say? Well, we can say that if we look at the point $(x_1 + \cdots + x_4 - x_5 - \cdots - x_8, y)$, then this is a point in $4\Gamma' - 4\Gamma'$, just by definition (we add 4 things from Γ', and then subtract 4 other things). But now the kicker is the first coordinate is 0 by assumption. And therefore we conclude that $(0, y) \in 4\Gamma' - 4\Gamma'$. And this means by definition that $y \in Y$. But now y is in a bit of trouble — it has to be in both W and Y, and there's only one shared point by how we chose W, namely 0. So we get $y = 0$, and we're done. □

Remark 26.9. Here we were a bit lucky because we were in a finite field setting, so we could choose W to be a subspace, which is fully closed under addition. If you work with the integers there are no subspaces, but there are things that *look* like subspaces, and this is a place you have to modify the argument.

Now we'll state a lemma, and maybe sketch a proof. The main lemma we got states that if you're weakly linear, then you're a Freiman homomorphism on a very large subset. Now we're going to go an extra mile and say that we actually have large agreement with an actual homomorphism. (We're actually going to use homomorphisms of order 8, but this proof can be modified to get that.)

Lemma 26.10

Suppose that $\varphi\colon A \to \mathbb{F}_p^n$ is a Freiman homomorphism of order 8, where $A \subseteq \mathbb{F}_p^n$ satisfies $\mu(A) \geq \eta$. Then there exists a homomorphism $\psi\colon \mathbb{F}_p^n \to \mathbb{F}_p^n$ and some shift $s \in \mathbb{F}_p^n$ such that $\mathbb{P}_x[\varphi(x) = s + \psi(x)] \geq p^{-O(1/\eta^3)}$.

This is where you kind of see you're in good position — we started with something very weak (that you're weakly linear), and got that now you're properly fully linear on a large chunk of your space. So you've got *global* structure, as opposed to the local thing that you started with.

Proof sketch. First we look at the set $2A - 2A$; we claim that we can define a function $\widetilde{\psi}\colon 2A - 2A \to \mathbb{F}_p^n$ in a 'sensible' way. If we have $a_1, \ldots, a_4 \in A$, then we define

$$\widetilde{\psi}(a_1 + a_2 - a_3 - a_4) = \psi(a_1) + \psi(a_2) - \psi(a_3) - \psi(a_4)$$

in the obvious way. We have to check that this is actually well-defined — right now that's not obvious, because maybe we could choose two a_1, $\ldots$, a_4 giving us the same LHS but different RHS, in which case this definition would be bogus. But this is where Freiman homomorphisms come in — note that if $a_1 + a_2 - a_3 - a_4 = b_1 + b_2 - b_3 - b_4$, then we can rearrange this to $a_1 + a_2 + b_3 + b_4 = b_1 + b_2 + a_3 + a_4$. And we're a Freiman homomorphism of order 8, so we're also one of order 4; and then we get that $\varphi(a_1) + \varphi(a_2) + \varphi(b_3) + \varphi(b_4) = \varphi(b_1) + \varphi(b_2) + \varphi(a_3) + \varphi(a_4)$, and when you move these to the other sides then you get that what we wrote really is well-defined.

And now here's where things become nice. What good does it buy us to look at $2A - 2A$? The only good thing about thi sset is there was a problem in the problem set about it — on Pset 3 or 4, we proved that there exists a subspace $W \subseteq 2A - 2A$ of codimension at most $1/\mu(A)^3 = 1/\eta^3$ or something like this. So we have $\widetilde{\psi}$, and it's defined on a set that contains a subspace. Now what we're going to do is look at the restriction $\widetilde{\psi}|_W$.

Now the annoying thing we're not going to do, because it'll be like the same computation above but with 16 things instead of 8, is that it's easy to see $\widetilde{\psi}|_W$ is a Freiman homomorphism of order 2. (The way you get this is you plug in the definition of $\widetilde{\psi}$, and then you get some equation and apply the fact that φ is a Freiman homomorphism.)

But wait a minute! If we have a function defined on a subspace that's a Freiman homomorphism of order 2, then it's actually just a homomorphism (because subspaces are closed under addition).

And now what you do is once you have a homomorphism on a subspace, you can extend it to a homomorphism on the entire space — so we extend to a homomorphism $\psi\colon \mathbb{F}_p^n \to \mathbb{F}_p^n$ arbitrarily.

At this point we will leave the proof, because it gets a bit more painful. But it stands to reason that if you started with a function fully defined using φ and got some nice homomorphism, just by extending it, then ψ is going to agree with φ up to a shift. Roughly speaking, you let a_1 be variable and a_2, $\ldots$, a_4 be some fixed thing; that'll be the shift, and ψ will agree with φ up to this shift. So that's going to be the shift s in the theorem. (Doing this more precisely is a bit painful, and the details appear in the notes.) □

> **Remark 26.11.** When you generalize to integers, do you just replace things with Bohr sets? It depends on what argument you use; here any interval will do (the feature you use is that if you sum it with itself, then it grows by a factor of at most 2). (Gowers used intervals in his original paper.)

§26.3 Back to inverse Gowers

So that's neat. And now we end this detour of 2 and a half lectures, and actually return to our original motivation.

Recall that all this discussion stated with having a function f with $\|f\|_{U^3}^8 \geq \varepsilon$. From this we concluded there existed $\varphi \colon \mathbb{F}_p^n \to \mathbb{F}_p^n$ such that

$$\mathbb{E}_h \left| \mathbb{E}_x \partial_h f(x) \omega_p^{\langle \varphi(h), x\rangle} \right| \gtrsim \varepsilon.$$

And we proved that this implies

$$\mathbb{P}[\varphi(x) - \varphi(x+h_1) - \varphi(x+h_2) + \varphi(x+h_1+h_2)] \gtrsim \varepsilon$$

(maybe with a different ε, but it doesn't matter). And from all the stuff we did in the last lecture and this lecture, we're able to conclude that φ 'looks like a homomorphism' — meaning that there exists $\psi \colon \mathbb{F}_p^n \to \mathbb{F}_p^n$ and a shift $s \in \mathbb{F}_p^n$ such that

$$\mathbb{P}[\psi(h) = \varphi(h) + s] \geq p^{-1/\varepsilon^{O(1)}}.$$

So you get $\varphi(h)$ agrees with this homomorphism; and it stads to reason maybe you can replace φ in the original equation and replace it with ψ. Trying to do this naively (with a union bound) doesn't work; but it turns out that you can do it using a 'list decoding' argument (which we are not going to do). This basically allows you to find (ψ, s) such that

$$\mathbb{E}_h \left| \mathbb{E}_x \partial_h f(x) \omega_p^{\langle \psi(h)+s, x\rangle} \right| \gtrsim \varepsilon'.$$

And this is most of the battle, but it's still not completely trivial.

To simplify our life, let's say $s = 0$ for simplicity. Also, ψ is a homomorphism, so we can write it as $\psi(h) = Mh$ for some $M \in \mathbb{F}_p^{n\times n}$. Then from all of this discussion, we get that

$$\mathbb{E}_h \left| \mathbb{E}_x \partial_h f(x) \omega_p^{\langle Mh, x\rangle} \right| \gtrsim \varepsilon'.$$

You know the derivative is correlated with this, so it makes sense to think, is there any antiderivative for this thing? Because then maybe that'd mean f itself is correlated with the antiderivative.

Finding the antiderivative is a bit annoying, so we're only going to do it in the case M is symmetric. (This is the place of the argument where Gowers writes 40 pages. The symmetric case is not very hard, but actually reducing to the symmetric case is some Cauchy–Schwarz magic where you use a bunch of Cauchy–Schwarz to show M is almost symmetric. Gowers instead did some extremely complicated thing, and this trick was only found later.)

Say $M = M^\intercal$, and define $g(x) = \omega_p^{\frac{1}{2}\langle Mx, x\rangle}$. (So this is some quadratic function.) And let's look at the derivative here — then we get

$$\partial_h g = g(x)\overline{g(x+h)} = \omega_p^{\frac{1}{2}(\langle Mx,x\rangle - \langle M(x+h), x+h\rangle)}.$$

And now we open this up; the $\langle Mx, x\rangle$ cancels, and we get a bunch of other things, so we end up with

$$\omega_p^{\frac{1}{2}(\langle Mx,h\rangle + \langle Mh,x\rangle + \langle Mh,h\rangle)}.$$

And the first two terms are the same because M is symmetric — so we get that this is $\omega_p^{-\langle Mx,h\rangle - \frac{1}{2}\langle Mh,h\rangle}$. The second term is constant, so we can ignore it, and we've found an antiderivative. If you plug this in, you get that

$$\mathbb{E}_h \left|\mathbb{E}_x \partial_h f \partial_h \overline{g}\right| \gtrsim \varepsilon'.$$

And if you think about it for a moment, you see that

$$\partial_h f \partial_h \overline{g} = \partial_h (f\overline{g})$$

(we're taking multiplicative derivatives), so you get that

$$\mathbb{E}_h \left|\mathbb{E}_x \partial_h (f\overline{g})\right| \gtrsim \varepsilon'.$$

This is telling you that if you take a derivative, then the function is correlated with a constant; and that means $f\overline{g}$ itself is correlated with a linear function $\omega_p^{\langle k,x\rangle}$. (This is the basic thing we started with — if your derivative has a bias then you yourself are correlated with a linear function; you can also do this with the U^2 norm if you want.) This implies

$$\left|\langle f, g\omega_p^{\langle k,x\rangle}\rangle\right| \gtrsim \varepsilon'.$$

And this is what you were after — we've got that f is correlated with a quadratic thing. And this finishes the inverse Gowers for U^3.

The non-symmetric case is not much more difficult, but you need one more trick to show that M *has* to be close to symmetric.

§26.4 Density increment

We've worked for four lectures to get this statement, but it's still not clear how to get a density increment from it. Let's call $g\omega_p^{\langle k,x\rangle} = G$. Then what turns out to be the case:

Fact 26.12 — Let G be a quadratic function $G\colon \mathbb{F}_p^n \to \mathbb{C}$. Then there exists a subspace $W \subseteq \mathbb{F}_p^n$ of large dimension such that G is constant on each coset $a + W$.

This is a general fact we're not going to prove; you can toy with examples like $\omega_p^{x_1^2+\cdots+x_n^2}$ and see how to get something like this. One example is to look at the space where $x_1 = \cdots = x_p$, $x_{p+1} = \cdots = x_{2p}$, and so on; if you add up p things and they're the same, then things are going to be constant. And this is the sort of thing you can do.

Remark 26.13. For this example, let's say $p \mid n$. Then we claim if we look at $W = \{x \mid x_1 = \cdots = x_p, x_{p+1} = \cdots = x_{2p}, \ldots\}$, then this is constant — because $\omega_p^p = 1$.

This has large dimension — you pay a constant factor in the dimension, but you get a density increment. (This is good enough — you go from dimension n to dimension n/p, and density α to $\alpha + \alpha^5$. This is tolerable, and it's why you get these famous $\log\log\log$ things.)

So we got that f is correlated with G, and now we can partition the whole space into these cosets; G is going to be constant on each of those cosets, and what this ends up telling you is that f has to vary on cosets, which means there is some coset on which it'll be larger than the average.

So this is how you get a density increment out of the correlation.

§26.5 The GHz game

Let's end with something easier. Dor came to know about this by studying the GHz game from computer science. In computer science we like to call people verifiers, so we have a verifier and three players Alice, Bob, and Charlie. The verifier is going to sample (x, y, z), which are three bits that add up to 0. And he's going to send x to Alice, y to Bob, and z to Charlie; adn they're going to give him back constants a, b, and c. And he's going to accept if and only if $a + b + c = x \vee y \vee z \pmod 2$.

In other words, the verifier could choose challenge $(0, 0, 0)$, in which case the sum of answers should be 0; and in any other case, the sum of answers should be 1.

Now the question is, what's the best thing the players can do to make the verifier accept?

Fact 26.14 — The players can win with probability at most $\frac{3}{4}$.

Basically there are 4 possible things, three of which give an OR of 1, and one 0; so they can guess the result will be 1 and try to go for that, and you can prove nothing better is possible.

Question 26.15. What if the verifier instead samples n independent coordinates, such that $x_i + y_i + z_i = 0$ for each $i \in [n]$?

Now we send $x_1, \ldots, x_n$ to Alice, $y_1, \ldots, y_n$ to Bob, and $z_1, \ldots, z_n$ to Charlie, and gets back $a_1, \ldots, a_n$ from Alice and so on. And the players win if and only if they win on every coordinate — meaning $a_i + b_i + c_i = x_i \vee y_i \vee z_i \pmod 2$ for all i. This is something called *parallel repetition*, and we could give a full lecture about it but we won't.

The idea is the verifier gives multiple challenges to the players at the same time, with the hope that they can win with much smaller probability.

Question 26.16. What's the probability the players can win?

It feels like it should be $(\frac{3}{4})^n$, but this is not known.

Parallel repetition has a long history. It turns out that just raising the base value to the nth power is not correct in many other cases, so this initial intuition is actually false and something more interesting happens.

For a while, the best-known bound was something like $\mathbb{P}[\text{win}] \lesssim 1/\log^* n$ (this is a very heavy hammer from combinatorics called the density Hales–Jewett theorem). This goes to 0.

Then a few years ago, it was improved to $1/n^{\Omega(1)}$. Dor looked at this paper and it was very interesting, but he couldn't make sense of what was happening and how to improve it.

But it turns out youc an prove a bound of $2^{-\Omega(n)}$ — this was actually 7 pages (while the above was 30). It turns out that all you have to do is make the following observation, and then be knowledgeable about the stuff we've discussed.

Claim 26.17 — Suppose $x + y + z = 0$. Then the condition $a + b + c = x \vee y \vee z \pmod 2$ is equivalent to $2a + 2b + 2c = x + y + z \pmod 4$.

This can be checked by brute force; how you get to it is a different question. And the nice thing is that this is a completely linear equation — so it means

$$(2a - x) + (2b - y) + (2c - z) = 0.$$

So now let's define a function $F(x) = 2a(x) - x$ (note that a is the answer of Alice, which is a function of just x), and $G(y) = 2b(y) - y$ and $H(z) = 2c(z) - z$. And we get that

$$\mathbb{P}_{x+y+z=0}[F(x) + G(h) + H(z) = 0] \geq \varepsilon,$$

where ε is the winning probability; and the goal is to show ε is small.

What's the big point here? The big point is that F, G, and H look as if they're linear functions. So if F, G, and H were the *same* function, then this would say if we had three inputs adding to 0, their images do too; this is very similar to the weak form of homomorphisms we saw at the beginning of the lecture. And it turns out you can apply all the stuff we did today to get this sort of structure statement — e.g., that they're Freiman homomorphisms on large sets. And once you apply all these tools, you can actually show that they can't exist.

So sometimes just knowing this stuff and how to apply it can be very powerful.