1 LECTURE 1: Introduction

I will focus on the methods, theory, and perhaps sometimes algorithms for high dimensional statistical inference. There is a long history of this topic. Most of the results probably happened since 2000, back when I was a grad student. There were also many applications, I will touch upon some of these. The main focus of the course is to go through some of the typical methods and analyses from theoretical perspectives. I will show not necessarily the deepest or sharpest, but the ones which give the most theoretical insight. A lot of the methods we will talk about rely on the algorithm used to calculate, so we will also talk about that.

1.1 What is high-dimensional statistical analysis?

What is high dimension? People often refer to this as "large p, small n". We often organize data as a data matrix. We think of each row as a sample. Typically we have a tall matrix $\mathbb{R}^{n \times p}$. This is the setting of classical statistics. But consider gene expression data, which started to get analyze back in the 2000s. Here you get very short, fat matrices instead.

I don't quite like this terminology however. We are statisticians. When I hear small n, I think there's not much we can do. Also, you can often do similar things mathematicaly for both matrices, when you're doing something completely different statistically (e.g., SVD is PCA in the first case, and something else in the second).

When you have the large p, small n situation you have to be careful. With a limited sample size there is only so much you can do. From a theoretical side, there are other names which are often used: For instance, **finite sample analysis**. This is counter to the classical way of thinking about asymptotics. You typically use things like central limit theorem and law of large numbers. You assume sample sizes are very large so you can do asymptotic analysis. This allows you to neglect small-order terms. Small order terms, however, often become important in high dimensions.

So the idea of finite-sample analysis is as follows. We are interested in parameter θ, and construct estimator $\hat{\theta}$. The classical way of thinking about this is writing $\hat{\theta} - \theta =$ Leading term $+ o_n(\cdot)$. Finite sample analysis says, let us be more precise. Let us write $\hat{\theta} - \theta = f(n, p)$. You want $f(n, p)$ to be the same as the asymptotic analysis when you take n large enough. However, in practice, as a statistician, this will give you rather useless bounds in practice if your sample size is five.

So what do I actually mean by large p small n: Really, I actually do have large sample sizes, but I don't want to treat this as infinity. I have even larger p. Here, I care less about relative order between n and p: I just want **both** to be large. I want a lot of variables and a lot of samples. The tricky part is often times it's not about which goes to infinity. Rather, it's about the interplay between the sample size and the dimensionality.

1.2 What is the fuss about high-dimensional inference?

The reason we are interested in this is mostly because of applications: This is why there was an explosion of popularity about this topic. The mathematics etc. is interesting, but there are numerous applications which we will touch upon throughout the course. There are other interesting things: The method. There are many new ways to think about statistical methodology. If you open a statistical textbook written before 2000, you see the following: Someone writes down the likelihood function, write down the maximum likelihood estimate, and proceed. Or to be fancier you can use regularization and model selection. These are the common strategies, suppose you can solve it, and then analyze the properties you have. When it comes to high-dimensional analysis, a key difference that becomes important is you take into consideration computation. One of the most relevant examples is to think about **variable selection**. When I was a grad student, we talked about AIC and BIC. Nowadays we talk about Lasso. Lasso is useful and can be applied to large scale problems – unlike AIC and BIC. Forward selection and AIC/BIC are still used actually – people tend to just put a fancier name. But in terms of methodological development, we care about computation a lot. In recent years, a lot of the growth in high-dimensional statistics is the tradeoff between statistical and computational resources. Are the optimalities you saw in introductory statistics still possible given fixed computational resources?

From a theoretical point of view, there are also a lot of new phenomena you must consider. Most of you have probably heard of the curse of dimensionality. The reason you care about high-dimensional statistical inference is there are a lot of new perspectives. A lot of the problems we consider are the same as fifty years ago. But a lot of the challenges we now encounter are different. I have a very good friend who works on numerical analysis. So I asked him, what's the most important question for you guys – he said, how to deal with large scale problems. Fifty years ago, this was the same problem: What changed? The definition of "large". That is the same problem statisticians are facing. We will focus a lot on linear regression: We've been doing this for years. But we have new challenges because of the new types of data we have.

1.3 Specific Topics

We will have evolving content throughout this course. I have a tentative schedule for the topics I will cover. Starting today, I will give some basic technical tools. I'll talk a little bit about high-dimensional geometry. I want to talk a little to give you some intuition which is important. When we talk about high-dimensional data, you want to first build some intuition. What can you expect from random datasets? Another category of tool which is essential is concentration inequalities. This analogy may not be precise, but concentration inequality to high-dim inference is kind of like law of large numbers to classical statistics. LLN lets you derive any classical result. Similarly, concentration inequalities let you derive high-dimensional results.

Then, we will go on to talk about sparse recovery/compressed sensing. The basic idea here is you have a high-dimensional vector and you want to reconstruct it. You assume it

has some sort of sparsity. This case may not have noise. From my perspective, I think this is essentially statistics. Much of the analysis is fundamental in understanding high-dimensional linear regression. There are several things we want to talk about in high-dimensional linear regression. First of all, we want to talk about parameter estimation. Relatedly, we want to talk about prediction. When you have high-dimensions, a related concept is variable selection. In high-dimensional data analysis, variable selection may not be the right way to phrase it: Instead, you call it support recovery (this is really the same as variable selection). The last thing we want to talk about here is to think about uncertainty: How does uncertainty propagate, and how do we do inference?

Then, I will talk about high-dimensional matrix problems. I will focus on three things: First, I'll provide background on matrix concentration inequalities. Then once we have these technical tools, I will talk about matrix completion or more commonly known nowadays as the Netflix problem. The right way to formulate this is not "matrix completion", but rather matrix regression. This is based on the assumption that the matrix is low-rank. The last topic I will talk about here is covariance matrix estimation. The basic problem is that we want to estimate a high-dimensional covariance matrix. We will talk about a few classes: sparse covariance matrices and sparse inverse covariance matrices. In the Gaussian case there is a direct correspondance between inverse covariance matrices and graphical models. Finally, we are also interested in things you can compute from covariance matrices – particularly, PCA.

This will be evolving content, I have these basic topics in mind. We will probably spend two or three classes on the topics. The last four projects we will do class presentations. There is a huge literature on high-dimensional inference. I am going to make this more systematic in the class by choosing main areas and topics. There are numerous other topics hard to explain in detail throughout the course. Thus I will ask you to do in-class presentation. We will decide in a week or two how to proceed with class presentations.

1.4 Organization of the course

This is a topical class: You want to learn something here, I don't have a strict way of grading. I want to give you a good formula to make the best of the course. In the class, as I mentioned, I will talk about results I find the most illustrative of a specific topic. Each class, I will give you papers or references based on the things I discuss in class. So to make the most of course, you don't necessarily need to prepare in advance. In class, you try to follow what we discuss. I don't have written lecture notes or slides: I will go through the material on the board. I encourage you to come and go through the lectures with me. Then follow up and go through the papers. If you have questions, you can come to office hours, which I am thinking of having right after the class. You can always email me to discuss further.

Regarding class presentations, make an appointment and talk to me about specifically what you want to present in the class.

2 High-Dimensional Data

We are interested in the dimension of the data being high. How is this different from classical statistics? High-dimensional data has some new features: some are good (can take advantage of) and some are bad. These are the blessings and curses of dimensionality. Bellman in 1957 invented the term curse of dimensionality: This is our inability to approximate/estimate/optimize a highi dimensional function. I want to construct an approximation of it, or I want to estimate it, or I want to optimize it. These problems are trivial for linear problems, but in high dimension, this can be difficult. Let me give a simple example.

Example 2.1. Suppose I have a uniform distribution between 0 and 1. If I sample at random, I'll get a point in $[0.2, 1]$ with 80% chance. If I go up in dimension to a square to sample with 80% of the chance, the smallest coordinate will be 0.45. In three dimensions, if I look at a cube and want to sample with 80% chance, the smallest coordinate will be 0.58.

However it's important to be careful about the subtetly of dimensionality. One of Kolmogorov's more famous theorems is as follows:

Theorem 2.2. *Kolmogorov's Theorem.*
Suppose I want to look at continuous one-dimensional functions on the unit interval $C([0,1])$. Suppose then I look at continuous functions on the unit square $C([0,1]^2)$. Here it should feel like the problem gets harder. But you can say these classes of functions are of similar complexity. You can always find $g_1, \cdots, g_5 \in C([0,1])$ so that the following happens: For all $f \in C([0,1]^2)$, then you can find $\phi_f \in C([0,1])$ and $f(s,t) = \sum_{i=1}^{5} \phi_f(g_i(s) + \sqrt{2}g_i(t))$.

So sometimes curse of dimensionality is not so simple. Here the curse of dimensionality comes in terms of the modularity of the function, which is not so simple. There are ways you can approximate multivariate functions with "simple" functions. This is actually kind of the view of deep learning. This is also one of the answers to Hilbert's 13^{th} problem.

Now there are also positive things which come with dimensionality. This is the "blessing of dimensionality". One of the simplest examples is as follows.

Example 2.3. Consider a fat matrix in $\mathbb{R}^{n \times p}$. I can still do SVD: The main reason this works is because p is large. This is sort of a superficial example.

Example 2.4. Your data, although in a classical low-dimensional setting, it may have a lot of irregular behavior. In high-dimensions, the data has a certain regular behavior. Suppose I have a random vector $X \sim U([0,1]^d)$, and I am interested in $\|X\|^2$. We know that $\max \|X\|^2 = d, \min \|X\|^2 = 0$. The average is $d/3$.

In high-dimensions, most random vectors are going to be very close to $d/3$. The reason is simple. Suppose I want to look at

$$\mathbb{P}\left\{\|X\|^2 \geq (1+\epsilon)(d/3)\right\}$$

The distortion ϵ is small. We can use Chebyshev's inequality to calculate that this is $< \frac{\mathbb{E}\left[\|X\|^2 - d/3\right]^2}{(\epsilon d/3)^2} = \frac{d\mathbb{E}\left[x_1^2 - 1/3\right]^2}{\epsilon^2(d^2/9)} = \frac{d(1/5 - 1/9)}{\epsilon^2 d^2/9} = \frac{4}{5\epsilon^2 d}$ which goes to 0 as d goes to infinity. So,

$$\mathbb{P}\left\{(1-\epsilon)d/3 \leq \|X\|^2 \leq (1+\epsilon)d/3\right\} \to 1$$

This is the concept of concentration behavior.

2.1 High-dimensional geometry

I want to talk about the high-dimensional cube and high-dimensional balls. ℓ_∞^d means ℓ_∞ norm in a d-dimensional space, and ℓ_2^d means Euclidean norm in a d-dimensional space. If we want to talk about a ball of radius r, we will refer to $B_r(\ell_\infty^d) = \{x \in \mathbb{R}^d : \|x\|_{\ell_\infty} \leq r\}$ and $B_r(\ell_2^d) = \{x \in \mathbb{R}^d : \|x\|_{\ell_2} \leq r\}$.

We often want to be able to visualize and have intuition of parameter space. That's why it helps to know about high-dimensional geometry. When you have high-dimensional data, a lot of interesting things happen. Most often your intuition about two-dimensional and three-dimensional objects no longer holds. Suppose I have an ℓ_∞ unit ball and an ℓ_2 unit ball. We have $B_1(\ell_2^d) \subseteq B_1(\ell_\infty^d)$. You basically have a ball saturate the cube, and then there are some corners sticking out in dimensions 2 and 3. Is this still the case when d is large? What percentage of volume of the ℓ_∞ ball is occupied by the ℓ_2 ball? We have $\text{Vol}(B_1(\ell_\infty^d)) = 2^d$. We also want to calculate $\text{Vol}(B_1(\ell_2^d))$. The easiest way to compute this is to use polar coordinates. We can write this as a multiple integral over a multi-angle:

$$\text{Vol}(B_1(\ell_2^d)) = \int_{S^{d-1}} \int_0^1 r^{d-1} dr d\theta = \frac{\text{Area}(d)}{d}$$

What this says is very simple: The volume is the surface area divided by d. $d = 2$ is a sanity check. Now how do we calculate $A(d)$? The way of computing this goes back to Gaussian distributions. Let

$$I(d) = \int e^{-(x_1^2 + \cdots + x_d^2)} dx_1 \cdots dx_d = \pi^{d/2}$$

since this is the normalizing constant for a Gaussian density. This integrand is also invariant to rotation, so we can write in polar coordinates as $\int_{S^{d-1}} \int_0^\infty e^{-r^2} r^{d-1} dr d\theta$. Realize the inner term is the Gamma function. Then, what you'll get is that this term is $\frac{\Gamma(d/2)}{2} A(d)$. Thus,

$$A(d) = \frac{2\pi^{d/2}}{\Gamma(d/2)}$$

So this means that the denominator grows much much faster: As d increases, the area goes to zero! And since the volume is the area divided by d, the volume also goes to zero! So something very unusual happens: In the ℓ_∞ ball, the volume increases expontentially, very fast. But the ℓ_2 ball is growing smaller and smaller: for very large d, it essentially vanishes. Almost every point in my ℓ_∞ ball will be different from what you expect for ℓ_2 ball. So how do you reconcile this kind of issue?

The main takehome message is the following: You are still cutting corners with the ℓ_2 ball and the ℓ_∞ ball: But in high-dimension, **everything is a corner**: The ℓ_∞ ball is super pointy. So there is very little interior for the ℓ_2 ball.

We are still interested in the ℓ_2 ball. Historically, we love Gaussian distributions. We impose Gaussian assumptions – not because we necessarily believe that the data follows Gaussian distribution, but it is a good approximation; that is what the Central Limit Theorem tells us. In high-dimension, you have to be very careful. Now, the ℓ_2 ball is related to the Gaussian distribution. Everyone will more or less lie on the sphere of the unit ℓ_2 ball if we have a Gaussian distribution. So we are interested in what happens. Let me make this more precise.

Suppose you can sample from the unit ℓ_2 ball – where are the points? Where is most of the mass? All of the mass is around the outermost shell: Most of the mass is toward the edge. Let the radius be 1, the inner shell radius is $1-\epsilon$. Then we are looking at

$$\frac{\text{Vol}(B_1(\ell_2^d)) - \text{Vol}(B_{1-\epsilon}(\ell_2^d))}{\text{Vol}(B_1(\ell_2^d))} = \frac{V(d) - (1-\epsilon)^d V(d)}{V(d)} = 1 - (1-\epsilon)^d$$

So this basically goes to 1. Why is this important and useful? Let us say we have $X \sim \mathcal{N}(\mu, I)$ in $\mathbb{R}^d$. Maybe you would use Stein's estimate to estimate it, or **any other estimate**. We want to estimate $\|\hat{\mu} - \mu\|^2$, but this will always be of order d if we only see one example. This is minimax. Suppose I give prior information that $\|X\|^2 = o(d)$. Then if you use an esimate of 0, you incurr error of order $< d$, beating Stein's estimate. But there is a contradiction here. I'm going to handwave a bit: Uniform distribution over unit ball is not the same as normal, but you can do a similar calculation. Anyways, you know that $X \sim \mathcal{N}(0, I)$ which has expected value d is concentrated: You know that $\|X\|^2 - d = o_p(\sqrt{d})$ – you can calculate this using χ^2-random variable distribution properties. What does this tell us? This tells us something strange. If μ^2 is of order $\sqrt{d}$, if you ask me to test it, I can reject the hypothesis. This is connected to the geometry we saw earlier. If you move the center of the ball a little bit, I know you moved the ball, but I cannot tell you where the center is, since all the volume is concentrated around the shell. So, you can tell that something is wrong, but the best estimator is still 0.

Your minimax estimation rate is of order d. Suppose I tell you the risk is of order d. But I'm going to tell you a priori that the size of μ is $\sqrt{d}$. This means that zero is the best estimator. However, with confidence, I can definitely say the mean is definitely not zero, but you also cannot construct any estimate which is better than zero.

The next claim is the following: Most of the point masses are not in the shell, but around the equator. Suppose I cut some slices with height ϵ. Just like all point masses are around the shell, I can also claim that most of the point masses is in this equator slice. Here, I am computing the volume by fixing one coordinate (I'm saying $x_1 \in (-\epsilon, \epsilon)$). Let's calculate (do it as an exercise):

$$\frac{1}{V(d)} \int_{-\epsilon}^{\epsilon} (1 - x_1^2)^{(d-1)/2} V(d-1) dx_1$$

You will find that this is essentially 1. So, the mass is mostly around the equator. So

you want to keep an open mind, there are a lot of interesting things that could happen. You have to think about this sort of high-dimensional geometry. What kind of features do you observe in the data? What is signal and what is noise?

There are two things I did here: The first message is in the comparing of ℓ_2 and ℓ_∞: When you think about parameter space, you have ℓ_2 and ℓ_∞ ball: All that matters to me is that you realize all the points in the ℓ_∞ ball, all the points are sort of "extreme" in a certain way: The "unusual points" (the corners of a cube) become the usual points, relative to the initial ball. If you are a geometer, the key point is that maybe the Lesbegue measure is not maybe the right measure to calculate the volume.

You also need to think carefully about how you formulate your statistical problem. Potentially there is low-dimensional structure in high-dimensional space. This is the essence of the methods we will discuss in the course. Simply because of high-dimensional data, you won't have enough degrees-of-freedom to deal with all high-dimensional problems. You want to find the flexible subclass of high-dimensional problems you can deal with. How do you restrict the parameter space, staying in high-dimensions, so that it becomes more manageable.

Now we will discuss the Johnson-Lindenstrauss theorem. Suppose I have a dataset with n points. JL asks the following question: How high is the dimension of the data points? This sounds like a weird question: You collected the data, you know the dimension. But it could be on a low-dimensional space. Suppose your data is really really high dimension. Their motivation is storage. Suppose all the data are high-quality images. Now the question is, if you want to do photo analysis, I'm going to vectorize it and put it in a database. Later on I'm doing analysis, I don't care about the individual images, just their relative distances. Can I put them in a lower dimensional space?

Theorem 2.5. *Johnson-Lindenstrauss.*
Given n points $x_1, \cdots, x_n$ in Euclidean space $\mathbb{R}^d$, I want to preserve pairwise distances. I want to find $f : \mathbb{R}^d \to \mathbb{R}^k$ with $k < d$ such that for all i, j

$$(1-\epsilon)\|x_i - x_j\|^2 \leq \|f(x_i) - f(x_j)\|^2 \leq (1+\epsilon)\|x_i - x_j\|^2$$

where ϵ is called distortion factor. If $k = n-1$, there is no problem at all: You can get this exactly because n points lie in $n-1$ dimensions. However, JL can go much beyond this: $k = \frac{C}{\epsilon^2}\log(n)$ for some constant C.

This theorem means you can always project into a low-dimensional space while preserving pairwise distances. This is kind of strange: You know that you cannot have an isometric mapping, there must be some distortion. If you are very stringent, then there is not much you can do: You can only get $k = \min(n-1, d)$. If you allow for a little distortion, you can all of a sudden do a whole lot: You can get $\log n$. This is approximate isometry. Some people take this as a way for using data compression. Secondly, depending on what you want to do later on, this may not be good. This tells you about pairwise distances, but not the general geometry. It won't preserve any structure about points being all equidistant to each other for instance. If you think about relative distance it makes sense, but if you think about absolute

distances, this can be very different. Finally, the dependence $k = \mathcal{O}(\frac{1}{\epsilon^2}\log n)$ is optimal. There do exist examples where you can do no better. If I give you n points equidistant, if you don't allow for any distortion you have to put them in $n-1$ dimensional space. How do you verify this is essentially the lower bound? If you have n points in a k-dimensional space, distances becomes $(1 \pm \epsilon)$. For each point, draw a ball of radius $(1-\epsilon)/2$. You want to put all the smaller balls into a ball of size $1+\epsilon$. This will give you the lower bound, roughly, after some calculation.

3 LECTURE 2: High-dimensional Geometry and JL

Last time we talked about high dimensional balls and cubes. We essentially discussed the volume of high-dimensional balls with respect to cubes: In a way, you can think of high-dimensional balls as shrinking as dimension increases. If you think of cubes, you can fix the volume as 1 over dimensions: The ball, on the other hand, will have its volume go to zero. Perhaps more importantly for us is the concentration behavior: Most of the volume of the ball, from a sampling point of view, is most likely to be in certain regions of the ball. In particular, it could be on the sphere. We did a calculation: Most of the mass will be on a thin shell. It will also be concentrated around the equator area, which is counter-intuitive. This violates the 3-dimensional example. But you will get this behavior in high-dimension. It's concentrated both around the sphere and around the equator. I would like to emphasize we talk about this for two reasons: Keep an open mind for weird behavior, and also get intuition for when you analyze high-dimensional data. There are often fake figures in high-dimensional data analysis, you have to be careful of the features you extract.

Today, we will talk about the implications of high-dimensional data. In particular, the Johnson-Lindenstrauss theorem. Oftentimes, it is referred to as a lemma in the literature, since in the original paper it is called a lemma. But really, I think this "lemma" deserves status as a theorem.

Theorem 3.1. *Johnson-Lindenstrauss.*
I have $\{x_1, \cdots, x_n\} \in \mathbb{R}^d$, for d large. If d is large, this can be too large. We want to compress it while compressing main features. We can find a mapping $f : \mathbb{R}^d \to \mathbb{R}^k$, where we want k to satisfy certain conditions. Here, $k \geq c \cdot \frac{1}{\epsilon^2}\log n$ where c is a numerical constant. The condition this mapping satisfies is:

$$(1-\epsilon)\|x_i - x_j\|_2^2 \leq \|f(x_i) - f(x_j)\|_2^2 \leq (1+\epsilon)\|x_i - x_j\|_2^2$$

for all $i, j \in [n]$.

This theorem essentially states the following. If you have n points in d-dimensional space. You can always find a map from $\mathbb{R}^d \to \mathbb{R}^k$ such that you preserve ℓ_2 distances up to distortion ϵ. Now what does this theorem not say? A common mistake in interpreting JL is to think of it as saying d-dimensional space and k-dimensional space are isometric to each other: This is false! The key difference is that this function is for a *specific* set of n points: The specific

map f does NOT hold for *any* set of n points. The f depends on the dataset. In fact, this would be impossible: d-dimensional space is different from k-dimensional space.

How can we prove this result? One of the most common way of proving this is by constructive proof. This proof is actually "seemingly" constructive. Typically, constructive means, how do I infer from dataset how do I construct f? Basically, write a program with dataset as input and output the function f, which applies to any data. But the proof doesn't work exactly like that. It is still unknown how to deterministically construct such a mapping. What I can prove is the following: I can show a random algorithm that produces something which I don't know precisely, but which I can generate. This is the original idea by Johnson and Lindenstrauss. More specifically, we can take f to be a projection. We can always write $f(x) = Ax$, where $A \in \mathbb{R}^{k \times d}$ is a projection matrix ($AA^T = I_{k \times k}$). So JL originally said we can find some A such that the JL properties are satisfied. However, there is an extra scaling factor, to adjust for the fact that we are taking the sum of squares of k things, and we want to compare with sums of squares of d things (in the original space):

$$(1-\epsilon)\|x_i - x_j\|_2^2 \leq \frac{d}{k}\|A(x_i - x_j)\|_2^2 \leq (1+\epsilon)\|x_i - x_j\|_2^2$$

Thus, to be precise, we have $f(x) = \sqrt{\frac{d}{k}}Ax$. Johnson-Lindenstrauss said we can generate a whole bunch of projection matrices, and then argue that at least one of them is good. Here, we only need to know average behavior in a sense. So, you can uniformly sample from a Grassmanian $\mathcal{G}(d,k)$ (these are the projection matrices $k \times d$) to get A. If you only sample one, you cannot be certain this works. So you want to sample a bunch of them. First, we look at if we sample one, what kind of distribution follows.

Proof. The first step is to reduce this into a property of a projection: What do we want here? In order to prove Johnson-Lindenstrauss, we want to show the following: If A is uniformly sampled, then we want to show

$$\mathbb{P}\left\{(1-\epsilon)\|x_i - x_j\|_2^2 \leq \frac{d}{k}\|A(x_i - x_j)\|_2^2 \leq (1+\epsilon)\|x_i - x_j\|_2^2\right\} \geq 1 - \frac{2}{(n+1)^2}(*)$$

The only randomness here is over the random projection. We will argue that this is true for random projects, and show that this implies the theorem. The next simplification is to focus on just one i, j pair.

Now, assuming $(*)$ is true, then

$$\mathbb{P}\left\{(1-\epsilon)\|x_i - x_j\|_2^2 \leq \frac{d}{k}\|A(x_i - x_j)\|_2^2 \leq (1+\epsilon)\|x_i - x_j\|_2^2, \forall i,j\right\} \geq 1 - \frac{2}{(n+1)^2}\cdot\binom{n}{2} = \frac{1}{n+1}$$

by union bound. But this doesn't solve our problem yet, this probability is pretty small. Then we can play the following game: For any randomly sampled A, we have the above equation. We can sample $n \log n$ different As: Then we claim that at least one of them will work. The reason is simple: the property holds for $A_{\tilde{j}}$ for some $\tilde{j}$ with probability at least

$1/(n+1)$, and the probability that it holds for **none** of the A is $1-(1/(n+1))^{n\log n} \to 1$. This is the probability amplification technique. You can make this probability as close to 1 as you want by generating more random projections.

Thus, we only need to prove the lower bound on the probability of a uniformly sampled Grassmanian satisfying the JL property. Recall, we want to prove:

$$\mathbb{P}\left\{(1-\epsilon)\|x_i-x_j\|_2^2 \leq \frac{d}{k}\|A(x_i-x_j)\|_2^2 \leq (1+\epsilon)\|x_i-x_j\|_2^2\right\} \geq 1-\frac{2}{(n+1)^2}$$

for randomly sampled A. This is a scale-free statement, since we can multiply by any scale. Let us re-write

$$(1-\epsilon) \leq \frac{d}{k}\|A\left(\frac{x_i-x_j}{\|x_i-x_j\|_2}\right)\|_2^2 \leq 1+\epsilon$$

Then we can consider $w = \frac{x_i-x_j}{\|x_i-x_j\|_2}$, which is a point on the unit sphere. Thus we want to prove

$$1-\epsilon \leq \frac{d}{k}\|Aw\|^2 \leq 1+\epsilon$$

Now we need to talk about uniform Grassmanians: This theorem can be seen as a statement about them.

The second step is to draw the following observation. We are going to think in a different way: What if we fix a projection, and uniformly sample a point? Let point u be uniformly distributed on d-dimensional unit sphere, and take $P \in \mathcal{G}(d,k)$. Then, Pu will have the same distribution as Aw, where A is randomly sampled from $\mathcal{G}(d,k)$ and w on the d-dimensional unit sphere is fixed. We can say this by uniformity: When you do uniform sampling from projection, you are doing uniform sampling from sphere.

For intuition, let's make it simpler. Suppose $d=3, k=3$: We are going to uniformly transform a point uniformly over the unit sphere. What uniformly sampled projection does is move it around the same unit sphere. No matter where you start, because you're doing uniform random sampling, you will end up with a uniform random variable.

So, what does the Grassmanian do? It will move the point around the sphere in a uniform fashion. This is clear for $d=k$. If $k<d$, you can think of this as random rotation in d-dimensional space, and then truncate the coordinates. This is a very powerful idea. If you really think about this proof, this is the key. The key difference is $\|Aw\|$: we don't know how to calculate the probability for uniformly sampled Grassmanian. But now, all of a sudden, we are looking at a uniformly sampled point on a sphere, which we have a much better idea of how it works. So, we only need to show that

$$\mathbb{P}\left\{(1-\epsilon) \leq \frac{d}{k}\|Pu\|^2 \leq 1+\epsilon\right\} \geq 1-\frac{2}{(n+1)^2}$$

Here, the probability is with respect to u, and not P (above, when we wrote this expression, the probability was with respect to A). Now, how do we do this? This holds true for any projection. We will define $Pu = ((Pu)_1, \cdots, (Pu)_k)^T = (\tilde{u}_1, \cdots, \tilde{u}_k)^T$. We extra the first k coordinates, and we want to know how big its norm is.

Note: For more rigorous proof of the trick for uniform sampling used above, see this Wikipedia article, section "Associated measure": We use the Haar measure over the orthogonal group, where the measure is invariant under group actions (e.g. rotations). You can sample a normal random matrix and take the singular vectors: This a uniform random variable in the Grassmanian. Intuition is the same as sampling uniformly from the sphere, using Gaussians.

This is what we will do in Step 3: We want to know the norm $\|Pu\|^2 = \tilde{u}_1^2 + \cdots + \tilde{u}_k^2$. Now we would like to characterize the uniform distribution on the sphere: Essentially, $x \sim \mathcal{N}(0, I_{d\times d})$ has $\frac{x}{\|x\|_2}$ is uniform on the unit sphere in d-dimensions: $\mathcal{U}(S^{d-1})$. You could prove this algebraically, but it's much easier to think about it geometrically. The reason this holds because the multivariate normal distribution is spherically symmetric. Therefore,

$$\|Pu\|^2 \equiv^d \frac{x_1^2 + \cdots + x_k^2}{x_1^2 + \cdots + x_d^2}$$

for normally distributed $x_i \sim \mathcal{N}(0,1)$ drawn i.i.d. We show

$$\mathbb{P}\left\{\frac{d}{k}\frac{x_1^2 + \cdots + x_k^2}{x_1^2 + \cdots + x_d^2} \leq 1 - \epsilon\right\} \leq \frac{1}{(n+1)^2}$$

and

$$\mathbb{P}\left\{\frac{d}{k}\frac{x_1^2 + \cdots + x_k^2}{x_1^2 + \cdots + x_d^2} \geq 1 + \epsilon\right\} \leq \frac{1}{(n+1)^2}$$

These imply the desired statement. The proof for both of these statements is the same. It comes down to some linear algebra.

$$d(x_1^2 + \cdots + x_k^2) \leq k(1-\epsilon)(x_1^2 + \cdots + x_d^2)$$

$$(k(1-\epsilon) - d)(x_1^2 + \cdots + x_k^2) + k(1-\epsilon)(x_{k+1}^2 + \cdots + x_d^2) \geq 0$$

$$(k(1-\epsilon) - d)\chi_k^2 + k(1-\epsilon)\chi_{d-k}^2 \geq 0$$

What is the probability that the left-hand expression is greater than zero? We can take exponent on both sides since it is monotone, for $\lambda > 0$:

$$\mathbb{P}\left\{\exp(\lambda\left((k(1-\epsilon) - d)\chi_k^2 + k(1-\epsilon)\chi_{d-k}^2\right) \geq 1)\right\} \leq \mathbb{E}\left[\exp(\lambda\left((k(1-\epsilon) - d)\chi_k^2 + k(1-\epsilon)\chi_{d-k}^2\right)\right]$$

applying Markov's inequality. Now we have reduced things to calculating the moment generating function of χ^2, which is known. We have

$$\mathbb{E}\left[\exp(\lambda\left((k(1-\epsilon) - d)\chi_k^2 + k(1-\epsilon)\chi_{d-k}^2\right)\right] = \mathbb{E}\left[\exp(\lambda\left((k(1-\epsilon) - d)\chi_k^2\right)\right] \times \mathbb{E}\left[\exp(\lambda\left(k(1-\epsilon)\chi_{d-k}^2\right)\right]$$

$$= (1 - 2\lambda(k(1-\epsilon) - d))^{k/2} \times (1 - 2\lambda(k(1-\epsilon) - d))^{(k-d)/2}$$

Then, by taking a derivative with respect to λ to minimize this quantity, and choosing the appropriate λ (since we know the inequality holds true for all λ), we see that we can get the desired result that this probability is upper bounded by $1/(n+1)^2$.

□

The key idea here is random projections. A huge idea nowadays is random sketching. You have a high-dimensional linear regression problem, and solve a low-dimensional linear regression problem. You want to know if you can get a solutions which is at least qualitatively comparable to the real one. The answer is yes, which may be surprising. A lot more goes into this when you do random projection. The second reason we want to talk about this is that the proof itself is very useful. Many of the ingredients in the proof apply to many other high-dimensional problems. The first one is the random projection part. When you want to prove the existence of something, we want to do constructive proof. This method gives you a way of showing you can find one. This is commonly used in randomized algorithms, which is a huge area in computer science.

Here, we proved a statement about the probability that the ratio of two chi-square random variables is greater than $1 - \epsilon$. It often seems like high-dimension is hard: But, from a concentration perspective, it is actually a good thing! It often happens in high-dimension you can do the X/Y trick, even though X and Y are not independent. Because of high-dimensionality, the expectation is close to 1. Summing a lot of things makes things centered around the expectation, and everything becomes relatively easy to compute and get an idea of where things are. Thus concentration inequalities are really good.

Back to JL: This construction is due to Johnson and Lindenstrauss. This particular proof is due to Gupta and Dasgupta, and is much more simplified, so the idea becomes much clearer. There are many other ways to construct this sort of linear transformation, instead of doing a uniformly sampled random projection. There are many ways to do this though. Here is a list of possible "f":

- uniformly sampled projection (what we saw)
- random ensemble: $f(x) = c \cdot Ax$, $A = (a_{ij})$: $1 \leq i \leq k, 1 \leq j \leq d$. Here, a_{ij} can be i.i.d. $\sim \mathcal{N}(0,1), \sim$ Bernoulli(1/2), and so on. These are essential properties of random matrices. In general, subGaussian random variables will work.

I would suggest you to do some simple calculations yourself. If every entry of A is a Gaussian standard normal random variable, can you verify that this is true?

Two other points:

- optimality: $k = \Omega(\frac{1}{\epsilon^2} \log n)$. Choose n equidistant points and do a volume calculation. This proves a lower bound, showing you must be of order $\log n$.
- is ℓ_2 necessary here? We have $x_1, \cdots, x_n \in \mathbb{R}^d$. We want to preserve the distances (for some distance metric) between points up to distortion ϵ. When does this work? In particular, we could replace all the distances with ℓ_p distance. If $1 \leq p \leq 2$, you can use the same construction, just with a different scaling factor. For $p > 2$, this is active research. Note that here, we are NOT embedding ℓ_2 into ℓ_1: We replace all norms in the statement with the ℓ_p norm. There are impossibility results (see Brinkman and Charikar (2005)) for JL statements with $(1 - \epsilon)\| \cdot \|_1 \leq \| \cdot \|_2 \leq (1 + \epsilon)\| \cdot \|_1$.

3.1 Concentration Inequalities

Suppose we have $X_1, \cdots, X_n \sim F$ i.i.d. with $\mathbb{E}[X] = \mu$. Then let $S_n = X_1 + \cdots + X_n$, and $\mathbb{E}[S_n] = n\mu$. Here, we are concerned with bounding $\mathbb{P}\{|S_n - n\mu| \geq t\}$. We essentially want to reduce things to calculating moment generating functions. Fortunately for us, usually we can do this. We can do the following typical thing with Markov's inequality, taking $\lambda > 0$:

$$\begin{aligned}
\mathbb{P}\{S_n \geq n\mu = t\} &= \mathbb{P}\{\exp(\lambda S_n) \geq \exp(\lambda(n\mu + t))\} \\
&= \frac{\mathbb{E}[\exp(\lambda S_n)]}{\exp(\lambda n\mu + \lambda t)} \\
&= (\mathbb{E}[\exp(\lambda X)])^n \exp(-\lambda n\mu - \lambda t) \\
&= \exp(n \log M(\lambda) - \lambda n\mu - \lambda t) \\
&\leq \inf_{\lambda > 0} \exp(n \log M(\lambda) - \lambda n\mu - \lambda t)
\end{aligned} \tag{1}$$

where $M(\lambda)$ is the moment-generating funtion (MGF). This is sometimes referred to as the Cram'er device.

3.1.1 Chernoff bounds

We only need things to be independent in order to apply the tricks above.

Example 3.2. Binomial random variables.
We have $X_i \sim \text{Bin}(1, p_i)$ and $S_n = X_1 + \cdots + X_n$, with $\mathbb{E}[S_n] = p_1 + \cdots + p_n = \mu_n$. Then,

$$\begin{aligned}
\mathbb{P}\left\{S_n \geq (1 + \frac{\epsilon}{\lambda})\mu_n\right\} &= \mathbb{P}\left\{e^{\lambda S_n} \geq e^{\lambda(1+\epsilon)\mu_n}\right\} \\
&= e^{-\lambda(1+\epsilon)\mu_n} \prod_{i=1}^{n} (\mathbb{E}\left[e^{\lambda x_i}\right]) \\
&\leq e^{(e^\lambda - 1)\mu_n - \lambda(1+\epsilon)\mu_n} \\
&\leq e^{(e^\lambda - 1 - \lambda(1+\epsilon))\mu_n} \\
&\leq \inf_{\lambda > 0} e^{(e^\lambda - 1 - \lambda(1+\epsilon))\mu_n} \leq e^{-\frac{\mu_n \epsilon^2}{3}}
\end{aligned} \tag{2}$$

since $p_i e^\lambda + 1 - p_i = 1 + p_i(e^\lambda - 1) \leq e^{p_i(e^\lambda - 1)}$, and where the last part is by Taylor expansion.

You can do the calculation for $1 - \epsilon$, instead of the $1/3$ factor in the exponent, you get factor $1/2$. This calculation is fairly standard, you should give it a try after the class. The idea is simple and is just calculus.

Notice that the dependence on ϵ is ϵ^2: Imagine binomial trial, they are all sort of independent. Then if all p_i are the same, then $\mu_n = np_0$. Then, you can take ϵ as small as $1/\sqrt{n}$. This reminds us of the central limit theorem. For instance, we can look at $\frac{1}{\sqrt{n}}(S_n - np_0) \to_d \mathcal{N}(0, p_0(1-\beta)) \to \mathbb{P}\{\sqrt{n}\mathcal{N}(0, p_0(1-p_0)) \geq \epsilon np_0\} = \mathbb{P}\{\mathcal{N}(0, p_0(1-p_0)) \geq \epsilon\sqrt{n}p_0\}$. So this can be viewed as a generalization of central limit theorem.

Now how can we use this in a more statistical setting.

We can suppose we get $X_1, \cdots, X_n$ are i.i.d. Then let the null hypothesis H_0 be that $X_i \sim \mathcal{N}(0,1)$, and H_a be the hypothesis that $X_i \sim (1-\pi_n)\mathcal{N}(0,1)+\pi_n\mathcal{N}(\mu_n,1)$: a mixture of Gaussians. You can think of the X_i as z-scores for n different hypotheses. If for every single one the null hypothesis holds, then it follows a standard normal distribution by definition z-score. Under the alternative hypothesis, they will still have a normal distribution, just with a different mean. π_n, μ_n are the only differences between H_0, H_a. We are interested in what happens in large n situation with very small proportion as $\pi - n$ goes to zero. You think of this as data generating mechanism. You have $\pi_n > 0$, but you don't know it. How do you estimate π_n. In high-dimension problem, it's important to ask the right question. Estimating π_n is much more difficult problem than estimating this thing. In the high-dimensional setting testing and estimation are very different problems.

The setting we are considering here is the simplest one – testing. Suppose you are in the alternative, then you can estimate π_n, μ_n. In the classical setting, these two questions are the same. The third question would be the most difficult: Not interested in global testing, but interested in multiple testing. You want to now identify indices i such that you can tell which distribution is followed for each i: This is a mixture model case, where for each distribution you can attach a label. This is genuine multiple testing. In the classical setting, the differences between these three are quite small. In the high-dimensional setting, these are very different. If you want to do multiple testing, then there is a subset of all indices such that the distributions follow the H_0 distribution rather than the H_a distribution. We will assume that $L = 0$ with probability $1 - \pi_n$ and $L = 1$ with probability π_n. Then we have $X_i|L = 1 \sim \mathcal{N}(\mu_n, 1)$, and $X_i|L = 0 \sim \mathcal{N}(0,1)$. Then we can re-write our hypotheses as $H_{0,i} : L_i = 0$ and $H_{a,i} : L_i = 1$. If you do these hypothesis tests many times, you are going to end up making mistakes just by chance. Hypothesis tests are always a probabilistic statement. If you do this many times, you almost surely are going to make a mistake. If you don't want to make any mistakes, then you have to keep increasing the bound on the probability. For every single one of them, you're only going to be right in rejecting a hypothesis 95% of the time – this gets reduced to a very small probability if you repeat the experiment many times. Let's do a quick calculation: If we have $X_1, \cdots, X_n$ drawn i.i.d. from $\mathcal{N}(0,1)$. Then, $\max_i X_i = \sqrt{2\log n}(1 \pm o_p(1))$.

Thus, in the regime $\mu_n = \sqrt{2r\log n}$, $\pi_n = n^{-\beta}$ there is no way to do consistent individual hypothesis testing (meaning individually test). But, it is possible to do consistent *global* hypothesis testing (meaning, look at the original H_0, H_a).

4 Lecture 3: Higher Criticism

This is a paper by Donoho. What we are going to look at is the following. We will look at the maximum value. We have two classes of observations, we don't know which comes from which. In order to differentiate them, we want to find a small subset with elevated means. These will tend to be the larger observations. We don't necessarily want to look at the largest one observation, we want to look at the largest bunch and analyze their collective

behavior. Collectively, they appear as a bigger cluster that tends to be large enough. We have X_i which we can view as z-scores. We then convert them into p-values. We want to look at the largest bunch of z-scores, or the smallest bunch of p-values. So we sort the p-values in increasing order: $p_1, \cdots, p_n$. These are like order statistics of uniform random variables. We expect p_k to be essentially like k/n. Then, we will normalize it. We have a r.v., and subtract its mean and standardize by standard deviation. We write

$$\frac{\sqrt{n}(p_{(k)} - k/n)}{\sqrt{p_{(k)}(1-p_{(k)})}}$$

How do we choose the k? Well, we can take a maximum over $\max_{1 \le k \le \alpha n}$, where α is arbitrary fraction kind of. This is called higher criticism test statistics (HC^*). There are many test statistics seem to make sense which break down in higher dimension. How do we argue this works in high dimension? We need to decide on the rejection region in a consistent manner. Note that you can always simulate from the null distribution using Monte Carlo and decide when to reject to ensure certain significance levels. How do we decide on it? One of the basic properties is the law of the iterated logarithm.

Let $HC^* = \sqrt{2 \log \log n}(1 + o_p(1))$. Reject H_0 iff

$$HC^* \ge \sqrt{2(1+\delta) \log \log n}$$

Under the null hypothesis, with probability tending to 1, this is true. Under H_a, we can apply Chernoff bound. First, we want to make Chernoff applicable to this setup. Instead of converting X_i into p-values, we convert it into an indicator random variable $X_i \to \mathbf{1}(X_i \ge x) = z_i(x)$. So, we look at $\tilde{HC}^* = \sup_x \frac{\sum_{i=1}^n z_i(x) - (1-\Phi(x))}{\sqrt{n(1-\Phi(x))\Phi(x)}}$ where Φ is the CDF of a standard normal distribution. This is like a modification of higher criticism. We want to show that

$$\mathbb{P}\left\{\tilde{HC}^* \ge \sqrt{2(1+\delta)\log\log n}\right\} \to 1$$

We need to think about how x^* needs to be:

$$x_* = \sqrt{2q \log n}$$

We will find the right q. We write

$$\mathbb{P}\left\{\frac{\sum_{i=1}^n z_i(x^*) - (1-\Phi(x))}{\sqrt{n(1-\Phi(x))\Phi(x)}} \ge \sqrt{2(1+\delta)\log\log n}\right\} \to 1$$

This looks like Chernoff bounds. We have $\sum z_i(x_*) \ge n(1-\Phi(x_*)) + \text{poly}\log(n)\sqrt{n(1-\Phi(x_*))\Phi(x_*)}$. Then,

$$\mathbb{E}\left[z_i(x_*)\right] = (1-\pi_n)(1-\Phi(x_*)) + \pi_n(1-\Phi(x_* - \mu_n))$$

Then, we have

$$1 - \Phi(x_*) = \int_{x_*}^{\infty} \phi(u)du \approx \text{poly}\log(n) n^{-q}$$

(Calculation is an exercise.) Then, $(1-\Phi(x_*-\mu_n))$ is the same story. This will also be $\text{poly}\log(n)n^{-(\sqrt{q}-\sqrt{r})^2}$. Putting everything together

$$\sum_i z_i(x_*) \geq \text{poly}\log(n)\cdot n^{1-q} + \text{poly}\log(n)n^{(1-q)/2}$$

Basically, the RHS of the inequality is $(1-\delta)\mu$ in a Chernoff bound. The first term is

$$n\left((1-n^{-\beta})\text{poly}\log(n)n^{-q} + n^{-\beta}\text{poly}\log(n)n^{-(\sqrt{q}-\sqrt{r})^2}\right) = \text{poly}\log(n)n^{1-q} + \text{poly}\log(n)n^{1-\beta(\sqrt{q}-\sqrt{r})^2}$$

All we need to make sure is that $1-\beta-(\sqrt{q}-\sqrt{r})^2 \geq \frac{1-q}{2}$. Suppose this holds. In both cases, the leading term is $\text{poly}\log(n)n^{1-q}$. If $\delta\mu$ is the second term, it is at least the square root of the first term. The probability that $\sum z_i(x_*) \geq \text{poly}\log(n)\cdot n^{1-q} + \text{poly}\log(n)n^{(1-q)/2}$ therefore goes to 1.

The goal is to make sure that $1-\beta-(\sqrt{q}-\sqrt{r})^2 \geq \frac{1-q}{2}$ has a solution for q. Recall that r is a parameter determined by the regime of μ_n. Then we get a quadratic equation in q: Let $s := \sqrt{q}$. We get

$$s^2 - 4s\sqrt{r} + (2\beta + 2r - 1) < 0$$

If it has roots, we want $16r > 4(2\beta + 2r - 1)$ which implies $r > \beta - 1/2$. Recall that β represents the fraction that the alternative holds. r is how big you want to be. Since this is a parabola, it can have two roots. You want one root to be between 0 and 1. Thus, you want

$$\frac{4\sqrt{r} - \sqrt{8r - 8\beta + 4}}{2} < 1$$

So

$$r > \left(1 - \sqrt{1-\beta}\right)^2$$

Higher criticism is basically taking supremum of a standardized binomial trials process. We only care about larger values. Under null hypothesis, we know that it's going to be of the order $\sqrt{2\log\log n}$. We need to make sure that the value under alternative, after we binarize the test statistics, is large enough. So we just need a combination of β and r such that under the alternative, higher criticism exceeds that value.

The only trick in the calculation is approximating the survival function of the normal: It's of the same order as the density. Then the rest is just Chernoff bounds. The idea is fairly simple.

We are not going to spend much time talking about multiple testing in this course. But nevertheless, it represents a lot of the challenges posed by the high-dimensional setting. So it is a big part of high-dimensional data analysis. I want to use this example to illustrate how things will tie to what we discuss in the class. Many of the results are recent and the result is elegant, but the main technique is sort of more simple.

4.1 Concentration Inequalities

There are many concentration inequalities, some of which are useful in various settings. I talked about Chernoff bounds: Here are some others.

Theorem 4.1. *Hoeffding's inequality.*
We have X_i independent, such that they have distributions on bounded intervals $[a_i, b_i]$. Then

$$\mathbb{P}\left\{ |\sum_i X_i - \mathbb{E}\left[X_i\right]| \geq t \right\} \leq 2e^{-t^2/(2\sum_i (b_i - a_i)^2)}$$

Now you can forget about dealing with binomial trials.

Theorem 4.2. *Bernstein's inequality.*
For bounded random variables and for unbounded random variables. It is same spirit of Hoeffding, but takes into account how large the random variables can be. Unbounded replaces the condition that variable needs to be bounded with moment conditions. This applies to subGaussian r.v.'s.

These are both examples of Bennet inequalities.

A third category is bounded difference inqualities. The most well-known is

Theorem 4.3. *Azuma's Inequality.*
Say we have $X_0, X_1, \cdots, X_n$. No i.i.d. assumption. We want $|X_k - X_{k-1}| \leq c_k$. When the increments are independent, then you expect

$$\mathbb{P}\left\{|X_n - X_0| \geq t\right\} \leq 2e^{-t^2/(2\sum c_k^2)}$$

You can apply this to general functions. You can also sometimes get rid of the independent increments assumption: McDiarmid's Inequality, which can be applied in the martingale setting.

If you are really interested in concentration, there is also Talagrand's inequality. Talagrand asks can you get concentration with a sup over all x – it turns you can. This is used in empirical process theory.

These are the tools we will be using.

5 Sparse Recovery (Compressed Sensing)

This is a main topic of the course. The problem is deceptively simple. Forget about the statistics part for now. Say we want to sovle a system of linear equations. Suppose we have $y = \Phi x$, where $y \in \mathbb{R}^m, x \in \mathbb{R}^n, \Phi \in \mathbb{R}^{m \times n}$. In statistics we immediately relate this with linear regression. When $m < n$, this has many many solutions. What if we look at a particular solution, suppose the sparsest one? We may want to solve

$$\min \|x\|_0 \text{ s.t. } y = \Phi x$$

where $\|\cdot\|_0$ is the ℓ_0 "norm", in other words the count of non-zero entries of x.

The first question to address is under what conditions this equation has a unique solution. When is the sparsest solution well-defined? Donoho and Huo in 2001 identified the following example.

Example 5.1. Suppose $\Phi = [\Phi_1, \Phi_v]$, both orthogonal bases. Φ_1 is the time domain, Φ_v is the frequency domain. Look at $\left[I_M, \left(\frac{1}{\sqrt{M}}\exp(\frac{2\pi j}{M})\right)\right]$. If you have a signal sparse in the time domain, it cannot be sparse in the frequency domain and vice-versa: This is the **uncertainty principle**.

We want to say things like if things are sparse enough, there is a unique solution. A useful concept for doing this is *coherence*.

Definition 5.2. Coherence.
Define

$$\mu(\Phi_1, \Phi_2) := \max_{\phi_i \in \Phi_1, \phi_j \in \Phi_2} \langle \phi_i, \phi_j \rangle$$

assuming the two matrices are orthonormal bases. Since the columns are normalized, we always have $\frac{1}{\sqrt{M}} \leq \mu \leq 1$. To see this we can write one matrix in terms of the other as orthonormal basis.

The lower bound is achievable in the time-frequency domain case.

We will use the uncertainty principle in some of our proofs.

Definition 5.3. Uncertainty principle.
Suppose $\Phi_1 w_1 = \Phi_2 w_2$ and $w_1, w_2 \neq 0$. We must have

$$\|w_1\|_0 + \|w_2\|_0 \geq \frac{2}{\mu(\Phi_1, \Phi_2)}$$

Theorem 5.4. *If the sparsity is* $\leq \frac{1}{\mu(\Phi_1, \Phi_2)}$*, the solution must be unique.*

Proof. By contradiction. Suppose we have $\Phi z = \Phi x^*$. Then $\|z\|_0 = \|x^*\|_0, z \neq x^*$. This implies that $\Phi(x^* - z) = 0$. Then look at coordinates of $x^* - z$ corresponding to first system and second system. Call this value δ. Then,

$$[\Phi_1, \Phi_2] \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} = \Phi_1 \delta_1 + \Phi_2 \delta_2 = 0$$

Thus,

$$\|z\|_0 + \|x^*\|_0 \geq \|\delta\|_0 = \|\delta_1\|_0 + \|\delta_2\|_0 \geq \frac{2}{\mu(\Phi_1, \Phi_2)}$$

using the uncertainty principle.

But we had that the sparsity is $\leq \frac{1}{\mu(\Phi_1, \Phi_2)}$, so the sum $\|\delta_1\|_0 + \|\delta_2\|_0$ cannot be $\geq \frac{2}{\mu(\Phi_1, \Phi_2)}$. □

6 Lecture 4:

Recall that last week we used Chernoff bound to show the consistency of Higher Criticism for multiple-testing. We have $H_0 : X_1, \cdots, X_n \sim_{i.i.d.} N(0,1)$ and $H_{a,i} : X_1, \cdots, X_n \sim_{i.i.d.} (1-\epsilon_n)N(0,1) + \epsilon_n N(\mu_n, 1)$. We have $\epsilon_n \sim n^{-\beta}$, $\mu_n \sim \sqrt{2r \log n}$. HC is consistent iff $r > \max\{\beta - 1/2, (1-\sqrt{1-\beta})^2\}$.

Another thing we could have done was EM algorithm (maximum likelihood estimate and make inferences) because this is a parametric problem. But an issue here is that our problem happens right at the boundary (when $\epsilon_n \to 0$, or $\mu_n \to 0$). If our mixture is right around $1/2$, it's an easier problem. But as $\epsilon_n \to 0$, this problem becomes irregular and a standard MLE argument doesn't work so well.

Then we started about sparse recovery. The main idea is we want to solve a linear system $y = \Phi x$, where Φ is $m \times n$ with $m < n$. We want the sparsest representation.

6.1 Uncertainty Principle

How do we prove the uncertainty principle?

Theorem 6.1. *Uncertainty Principle.*
We have two orthonormal bases ϕ_1, ϕ_2. Suppose $\phi_1 \alpha = \phi_2 \beta \neq 0$. Then

$$\|\alpha\|_0 + \|\beta\|_0 \geq \frac{2}{\mu(\phi_1, \phi_2)}$$

Proof. WLOG suppose $\|\alpha\|_2 = 1$. Then $\|\phi_1 \alpha\|_2 = 1$ which implies $\|\beta\|_2 = 1$. Then,

$$1 = \|\phi_1 \alpha\|_2^2 = \alpha^T \phi_1^T \phi_1 \alpha = \alpha^T \phi_1^T \phi_2 \beta \leq \|\alpha\|_1 \|\beta\|_1 \mu(\phi_1, \phi_2)$$

because we are summing over all coords of α, β, and multiplying by an entry from the correlation matrix. So you can pick the maximum (the coherence) and remove the sums. Thus we get

$$\frac{1}{\mu(\phi_1, \phi_2)} \leq \|\alpha\|_1 \|\beta\|_1 \leq \frac{1}{2}\left(\|\alpha\|_1^2 + \|\beta\|_1^2\right)$$

and we can replace $\|\alpha\|_1^2 \leq \|\alpha\|_0 \|\alpha\|_2 = \|\alpha\|_0$ by Cauchy-Schwarz and since $\|\alpha\|_2 = 1$. □

This case (orthonormal bases) gives us all the intuition we want about why coherence is important.

6.2 Applying Coherence

Now suppose we have $y = \Phi x$ where Φ is $m \times n$ with $m < n$. Let

$$\mu(\Phi) = \max_{i \neq j} \frac{|\langle \phi_i, \phi_j \rangle|}{\|\phi_i\| \|\phi_j\|}$$

Suppose $\sqrt{\frac{n-m}{n(m-1)}} \leq \mu(\Phi) \leq 1$. In the case that Φ is a Gaussian ensemble (each entry is an i.i.d. standard normal) then $\mu(\Theta) \sim \sqrt{\frac{\log(mn)}{m}}$. What's the main result here in terms of sparsest solution?

Theorem 6.2. $\Phi x_* = y$. *If* $\|x_*\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\Phi)}\right)$ *there is a unique solution. This isn't as sharp as the two orthonormal case: If you take* Φ *as two orthonormal bases, you will be off by a factor of* $1/2$. *This is what you sacrifice for generality. So this bound is not always sharp, but it is general.*

Proof. In the original proof by Donoho, they used the notion of Kruskal rank.

Definition 6.3. Kruskal rank.
The **Kruskal rank** krank(Φ) is

$$\max\{|S| : \phi_S \text{ is linearly independent }\}$$

This is more of a uniform requirement for rank. If you take any $|S|$ columns, they are *all* linearly independent.

This concept was first identified by Kruskal in the study of the identifiability of tensors. This is useful for us as well. We have x_*, we want to show that x_* is the sparsest solution. Let us assume the contrary, and try to derive a contradiction. Suppose $\Phi x_* = \Phi z$. Then $\Phi(x_* - z) = 0$. If both x_* and z are both sparse, so is the difference between the two. Then, krank$(\Phi) \leq \|x_* - z\|_0 - 1 \leq \|x_*\|_0 + \|z\|_0 - 1$ by definition of Kruskal rank. Thus it suffices to show that krank$(\Phi) \geq \frac{1}{\mu(\Phi)}$ to show the contradiction of there existing two sparse solutions x_*, z: You'd get krank$(\Phi) < 1 + \frac{1}{\mu(\Phi)} - 1 = \frac{1}{\mu(\Phi)}$ as well, which is the contradiction. Let's assume that all ϕ_i have unit norm. Let's look at $A = \Phi^T\Phi$. Suppose $S \subseteq [n]$. Then $A_{SS} = (\phi_i^T\phi_j)_{i,j\in S}$. Then if A_{SS} is invertible (if it's positive definite instead of just PSD), such that $|S| \leq k$, this automatically implies krank$(\Phi) \geq k$. Let's look at A_{SS}: The diagonal is 1. If we show that this is diagonally dominating, then it will be positive definite. Thus look at the sum of the off-diagonals. There are at most $k-1$ of them, and each is at most $\mu(\Phi)$. Thus if $(k-1)\mu(\Phi) < 1$, A_{SS} will be positive definite. Thus $k < \frac{1}{\mu(\Phi)} + 1$ implies a constraint on the sparsity which we wanted. This is the general result using coherence to show uniqueness. □

6.3 Summary of coherence

So first, coherence is easy to compute relatively: You just look at the off-diagonal correlations of the design matrix. It's also easy to use in analyzing algorithms.

There also disadvantages. One is specifically because this is generally applicable: It doesn't give you the best result. Too general in a way. Secondly, the worst case result is sub-optimal. If we have $\mu(\Phi) > 1/\sqrt{M}$, then $\|x_*\|_0 < \sqrt{M}$, which is not optimal. You can get around this result with different design matrices.

6.4 Computational Considerations

Now, the ℓ_0 optimization problem is NP-hard. Also we will assume normalized columns. When m, n are large, this problem becomes hard. People have come up with various algorithms which have had varying success. What can you say when an algorithm works and when it doesn't work? We will highlight a few of them, the ones most commonly used in practice.

The first one is called **one-step-thresholding** (OST) or **marginal screening**. The idea is simple: Suppose you know the sparsity k of the sparsest solution. For $j = 1, \cdots, n$, we write $r_j = \phi_j^T y$. Then define set $\Lambda = \{j : |r_j| \text{ is in the top } k\}$. Basically take the largest k. You can think of r_j as a p-value. Then define $\hat{x}_{\Lambda^c} = 0, \hat{x}_\Lambda = (\Phi_\Lambda^T \Phi_\Lambda)^{-1} \Phi_\Lambda^T y$, a.k.a. linear regression. In what kind of situations does this algorithm work?

Definition 6.4. Support.
The support of a vector v is defined

$$\text{supp}(v) = \{j : v_j \neq 0\}$$

Then $k = |\text{supp}(x_*)| = \|x_*\|_0$.

Theorem 6.5. *OST (one-step thresholding) works ($\hat{x} = x^*$) if $\|x_*\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\Phi)} \cdot \frac{x_{min}}{x_{max}}\right)$. Here, x_{min} is the minimum absolute value of x_* over the support and x_{max} is the ℓ_∞ norm of x_*.*

Proof. All you need to show is that Λ is the support of x^*. We have $v_j = \Phi_j^T \Phi x_*$. All we need to show is that if you take any index from the support, you'll get a larger correlation than if you take from anyone outside the support. We treat these two situations separately. What if $j \notin \text{supp}(x_*)$? Then $|v_j| = \left|\sum_{i \in \text{supp}(x_*)} \phi_j^T \phi_i x_{*,i}\right| \leq \sum_{i \in \text{supp}(x_*)} |\phi_j^T \phi_i||x_{*,i}| \leq \|x_*\|_\infty \cdot k \cdot \mu(\Phi)$.

Now, if j comes from support what happens? If j doesn't, everyone behaves the same way. If it does, then we want to single out the j^{th} index. We will end up with $|v_j| = \left|x_{*,j} + \sum_{i \in \text{supp}(x_*), i \neq j} \phi_j^T \phi_i x_{*,i}\right| \geq |x_{*,j}| - (k-1)\mu(\Phi)\|x_*\|_\infty \geq x_{*,min} - (k-1)\mu(\Phi)\|x_*\|_\infty$. Thus, we have

$$x_{*,min} - (k-1)\mu(\Phi)\|x_*\|_\infty \leq |v_j| \leq \|x_*\|_\infty \cdot k \cdot \mu(\Phi)$$

to get the result.

□

In sparse recovery, we haven't talked about noise yet. It turns out you can do the same type of calculation if you do have noise. What happens in the regression setting? Say we have $y = \Phi x + \epsilon$, where $\epsilon_i \sim N(0, 1)$. Do same calculation. Only difference is $r_j = \phi_j^T \Phi x_* + \phi_j^T \epsilon$. Note that $\phi_j^T \epsilon$ will still follow $N(0, 1)$. So in addition to the things we saw before, we'll get a noise term. You'll end up with if $r_j \notin \text{supp}(x_*)$,

$$|r_j| \leq k\mu(\Phi)\|x_*\|_\infty + \max_{j \notin \text{supp}(x_*)} |\phi_j^T \epsilon|$$

We know this maximum is upper bounded by $\sqrt{2\log(n-k)}$. If $r_j \in \text{supp}(x_*)$, then $|r_j| \geq \cdots \sqrt{2\log k}$. Then you play the same game to see when marginal thresholding will work. Thus, once you understand sparse recovery setting, adding subGaussian noise is fairly easy to handle when focusing on coherence. It could get more complicated in other settings.

Now we talk about stepwise regression (forward or backwards). It's easier to do forward regression. You can always fit a smaller model. The next idea is **forward selection**. In signal processing literature, it's called orthogonal matching pursuit (OMP). Now it's an iterative procedure. Here, r stands for residual. We have $r^0 = y$. Then $\Lambda^0 = \phi$. Let us write

$$s_j = \phi_j^T r^{iter}$$

Instead of taking top k, we will take top 1. We will take

$$j_* = \text{argmax}_{j \notin \Lambda^{iter}} \{|S_j|\}$$

Then, $\Lambda^{iter+1} = \Lambda^{iter} \cup \{j_*\}$. Now we run linear regression on this set:

$$x^{iter+1} = \left(\Phi^T_{\Lambda^{iter+1}} \Phi_\Lambda\right)^{-1} \Phi^T_\Lambda y$$

We also have $x_j^{iter+1} = 0$ for all j not in Λ^{iter+1}. Then, $v^{iter+1} = y - \Phi x^{iter+1}$. We then update iter. What is the stopping criterion? In sparse recovery, we will recovery the exact solution. So here, the stopping criterion will be $r = 0$. This is the difference between regression and sparse recovery.

Theorem 6.6. *If* $\|x_*\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(\Phi)}\right)$, *then OMP recovers* x_*. *Thus,* $\hat{x} = x_*$.

Thus OMP works to some degree. This matches perfectly with coherence.

Proof. As long as you don't reach 0, at any iteration, the one you selected is going to be from the support of x_*. We will prove by induction. Suppose that the current set doesn't use this variable. I want to make sure I don't select something from outside the support. We proceed by proof by induction over Λ always being a subset of the support of x_*. There are only k things I can select from, so after k steps I am done. In the first step the selected set is empty. We want to prove $\Lambda^{iter} \subseteq \text{supp}(x_*)$. Then, $r^{iter} = y - \Phi x^{iter} = y - \Phi_\Lambda x_\Lambda$. We write $y = \Phi x_*$. Either we have a coordinate in Λ or outside Λ. All we care about are the following: we can write this expression as Φz. The support of $z \subseteq \text{supp}(x_*)$ since Λ is a subset of the support of x_*. This turns out to be good enough for our proof. Sufficient sparsity allows us to show uniqueness. Now look at $\phi_j^T \Phi z$: if j is not from the support of x_*, then $|S_j| \leq k\mu(\Phi)\|z\|_\infty$ as in one-step thresholding. Then if $j \in \text{supp}(x_*)$, $|S_j| = |z_j| - (k-1)\mu(\Phi)\|z\|_\infty$. We want to show the one we selected is from the support. We select j that gives the maximum. Then the one we selected will be at least be bigger than the value achieved at this particular j.

In one-step thresholding, you want to make sure the weakest among your group is selected ahead of others. That's why you end up with x_{min}. Here you want to make sure the best you selected is better than the rest. If you take a particular j, you'll ensure that the OMP

condition will make yours better than the rest. This will show that it comes from the support of x_*. You could keep selecting the same variable again and again, but we need to ensure this does not happen. You are projecting the other entries on the set of Λ. The correlation must be 0 on the projection of Λ, because you are running least squares. What this does is make sure that your residual is orthogonal to anyone from your current active set. (There is also matching pursuit, where there is no such guarantee). □

7 Lecture 5: (Get notes from Rishabh)

8 Lecture 6: (Get notes from Rishabh)

9 Lecture 7:

Last time, we finished talking about sparse recovery. The idea basically is we talked about how to use RIP to get around the square root bottleneck. If we want to solve $y = \Phi x$, minimizing $\|x\|_0$ or $\|x\|_1$ such that equality holds, then $x = x^*$ if $\delta_{2k} + \theta_{2k,k} < 1$, where δ_{2k} is the restricted isometry constant. Thus for Gaussian ensembles, $k := \|x_*\|_0 \sim \frac{m}{\log n}$. So this is much better than a coherence type analysis.

We talked about high-dimensional linear regression. Now we look at a linear equation $y = X\beta + \epsilon$, where $y \in \mathbb{R}^{n\times 1}, X \in \mathbb{R}^{n\times d}$, and $\beta \in \mathbb{R}^{d\times 1}$. Here, we assume $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$. We want to exploit the sparsity of β. We are interested in the case where $d >> n$. There is no hope you can solve this without additional constraints. The interesting thing here is that what we are interested in is not just exact sparsity, since these models are approximations to reality. So we want to talk about potential sparsity of β, and study how well we can do to quantify how difficult the problem is.

We discuss

Definition 9.1. Parameter spaces for approximate and exact sparsity.

(a) **approximate sparsity**: We will look at $\beta \in \Theta_{r,R} = \{v \in \mathbb{R}^d : \|v\|_{\ell_r} \leq R\}$, for $r > 0$. This is one parameter space.

(b) **exact sparsity**: $\beta \in \Theta_{0,k} = \{v \in \mathbb{R}^d : \|v\|_0 \leq k\}$.

These cases are related. We can incur some bias by solving the approximate case, and treat it as an approximation to the exact space. We want to know how difficult this problem is. So what is the most difficult case? Typically, this is easier to construct from the exact sparse situation. Then we'll consider the approximate sparse case.

In high-dimensional linear regression, we have different situations: we could have a random design situation, where the columns are generated from a random distribution. We can also have fixed design, meaning that X is not stochastic. We will make a distinction between

the two. In the analysis, we'll be conditioning on X. We will also normalize each of the columns of $X = (x_1, \cdots, x_d)$ so that $\|x_i\|_2^2 = n$. This makes the sample standard deviation to be 1. If the standard deviations are very different, the interpretations of β will be very different as well. We want to look at things on the same scale. In the high-dimensional case when d is large, this is very important – you could have different columns which scale very differently. Thus we scale.

9.1 Regression

So how do we do estimation and inference in this particular setup? Let us first look at estimation. Let us look at several loss functions: Consider $\|\hat{\beta} - \beta\|_{\ell_q}^q$, where $q \geq r$ and $\beta \in \Theta_{r,R}$ (otherwise, if $q < r$, the thing we estimate may not be in our parameter space). We can look at a minimax risk bound:

$$R_q(\Theta_{r,R}) = \inf_{\hat{\beta}} \sup_{\beta \in \Theta_{r,R}} \mathbb{E}_\epsilon \left[\|\hat{\beta} - \beta\|_{\ell_q}^q \right]$$

The infimum is taken over all estimates you could get. We want to know how big this is. This amounts to a lower bound. We also want to construct an estimate which can achieve this rate – this is the upper bound, and can be achieved by Lasso or Dantzig selector. Can we establish a lower bound? Let $\delta_n = \delta/\sqrt{n}$. We have the following: Let

$$\lambda_{mm,r} = \begin{cases} \delta_n \sqrt{2\log(d/k)} & r = 0 \\ \delta_n \sqrt{2\log(\frac{\delta_n^r d}{R^r})} & r > 0 \end{cases}$$

This represents the weakest signal that you can just estimate. If your signal is smaller than this, then if you look at the magnitude of β_j, it becomes impossible to recovery. This is the threshold, and depends on the parameter set which is why it depends on r. Once we have this, the bound is fairly easy:

$$R_q(\Theta_{r,R}) = R^r \lambda_{mm,r}^{q-r}(1 + o(1))$$

This result is due to (Ye and Zhang, 2010). The main technical step is due to an earlier paper. How to deal with the general regression setting? The main idea is as follows: We want to do a sup and an inf. You have to get rid of one of them first. One typical way of doing this is to get rid of the sup part first: How bad can the estimation be? You can use a Bayesian prior on β. We want this distribution to be defined over the parameter space. For simplicity, we can just take some distribution which has non-zero mass on the parameter space. Let the prior be denoted as π. Then,

$$\inf_{\hat{\beta}} \sup_{\beta \in \Theta_{r,R}} \mathbb{E}\left[\|\hat{\beta} - \beta\|_{\ell_1}^q | \beta \right] \geq \inf_{\hat{\beta}} \mathbb{E}_\pi \left[\mathbb{E}\left[\|\hat{\beta} - \beta\|_{\ell_1}^q | \beta \right] \right] = \inf_{\hat{\beta}} \mathbb{E}_\pi \left[\sum_{j=1}^d \mathbb{E}\left[|\hat{\beta}_j(x,y) - \beta_j|^q | \beta_j \right] \right]$$

Now we want to decouple these $\hat{\beta}_j$. We can write $z_j = x_j^T(y - x_{-j}\beta_{-j})/n = \beta_j + x_j^T\beta/n \sim \mathcal{N}(\beta_j, \sigma_n^2) \geq \inf_{\hat{\beta}} \mathbb{E}_\pi \left[\sum_{j=1}^d \mathbb{E}\left[|\hat{\beta}_j(x, y, \beta_{-j}) - \beta_j|^q | \beta_j\right]\right] = \inf_{\hat{\beta}} \mathbb{E}_\pi \left[\sum_{j=1}^d \mathbb{E}\left[|\hat{\beta}_j(z_j) - \beta_j|^q | \beta_j\right]\right]$.

Thus we are able to de-couple. Then, Donoho-Johnston in 1994 look at the following problem (this is most of the heavy lifting). We have $x \sim \mathcal{N}(0, \sigma_n^2 I_d)$, which is a Gaussian sequence model, which is different from the regression model we have. But you can do the same kind of derivation. The only difference is in our case, the z_js are correlated. It doesn't matter if they are dependent because we are computing an expectation over the prior. So what Donoho-Johnston showed was that

$$\inf_{\hat{\mu}} \sup_{\mu \in \Theta} \mathbb{E}\left[\|\hat{\mu} - \mu\|_q^q\right] = R^r \lambda_{mm,r}^{q-r}$$

They do this by appropriately choosing a prior: There is a lot of mass on the parts of the distribution which end up being hard to estimate. So given we know what $\lambda_{mm,r}$ is, we can assign a lot of mass to the threshold area. So,

$$\beta_j = \begin{cases} \lambda_{mm,r}(1-\epsilon) & \text{w.p. } p \\ 0 & \text{w.p. } 1-p \end{cases}$$

If you choose such a prior, it really does not matter if you are doing regression or Gaussian sequence case. Thus

$$\inf_{\hat{\beta}} \mathbb{E}_\pi \left[\sum_{j=1}^d \mathbb{E}\left[|\hat{\beta}_j(z_j) - \beta_j|^q | \beta_j\right]\right] = R^r \lambda_{mm,r}^{q-r}(1 + o(1))$$

But I cheated a little bit: this all holds if our prior distribution is supported on the set. It may not be though! So, we need to argue that with probability tending to 1, β is going to come from $\Theta_{r,R}$ under the prior π. This has to do with how we choose p, so that β indeed comes from that set. If we look at $\|\beta\|_r^r \asymp \lambda_{mm,r}^r(1-\epsilon)^r \cdot d \cdot p$. We can then use a Chernoff bound to show that the probability of belonging in the set is concentrated around 1, provided we choose p so that $(1-\epsilon)^r p \leq R^r$.

9.2 Prediction

Let's suppose we want to evaluate the error of prediction. If our design matrix is very well conditioned on the subset $\text{supp}(\beta) = S$. This is exactly the premise of restricted isometry. We will have a more general way of dealing with this.

Definition 9.2. Restricted Eigenvalue.
Let $\phi_{k,+}$ denote the eigenvalue as follows:

$$\phi_{k,+} = \sup_{\|u\| \leq 1, \|u\|_0 \leq k} \frac{1}{n} \|Xu\|_2^2$$

You can think of this as the max eigenvalue of a sparse u. You can also interpret it as a property of the Gram matrix: It's the largest eigenvalue of the $k \times k$ principal of the Gram matrix. Similarly, we can define

$$\phi_{k,+} = \inf_{\|u\|=1, \|u\|_0 \leq k} \frac{1}{n} \|Xu\|_2^2$$

We have $\frac{1}{n}\|Xu\|_2^2 = \frac{1}{n} u^T (X^T X) u$, where $\bar{\Sigma} = (1/n) X^T X$. This actually holds true regardless of whether or not you center your observation. In some applications, it makes sense to not center it. In regression model for prediction, you indeed may want to center it for interpretation purposes. This is more of a Gram matrix more than a covariance matrix.

So if you think about RIP, the constant is going to be related to these constants very directly.

$$\inf_{\hat{\beta}} \sup_{\beta \in \Theta_{0,k}} \mathbb{E}\left[\|X\hat{\beta} - X\beta\|^2\right] \geq \mathcal{O}(k\lambda_{mm,0}^2)$$

But you can actually say a bit more than this – you can get the constant right as well. The constant is $\frac{\phi_{2k,-}}{4} \cdot 1 + o(1)$. Minimax analysis is about uniform performance over a whole subset. This ϕ doesn't show up in the estimation case because you're looking at all directions. Here, it shows up in the prediction risk because multiplying by X will change the scale: only a small number of directions might matter! However, we do require that $\phi_{2k,-} > 0$: otherwise, the result is completely meaningless since we would essentially be predicting onto a zero direction, and there would be no way to measure prediction error at all: everything would be zero! Also remember that this is a lower bound, not an upper bound: The lower the scale, the more things are possible. It doesn't mean that the upper bound is not large.

9.3 Estimating the Support

Let's make the parameter space a bit different. Consider $\tilde{\Theta}_k = \left\{v \in \mathbb{R}^d : \|v\|_0 = k, \min_{r:v_j \neq 0} |v_j| \geq \beta_*\right\}$. The magnitude of each non-zero entry needs to be β_*. We have $Y = X\beta + \epsilon$. Now suppose we want to estimate the support of β. In the high-dimensional case, the relationship between these two problems is much murkier: You can construct a good estimator of β that does not give a good estimator of the support of β. In the case where d is very very large, things will be bad. In low-dimensional setting, the probability is bounded and you can do well. In high-dimensional case, the probability can be arbitrarily bad. So what is the probability for us to estimate the support well? We look at

$$\inf_{\hat{\beta}} \sup_{\beta \in \tilde{\Theta}_k} \mathbb{P}\left\{\text{supp}(\hat{\beta}) \neq \text{supp}(\beta)\right\} \geq 1 - \frac{2\beta_*^2 + \delta_n^2 \log 2}{\delta_n^2 \log(d-k)}$$

Ideally you want this probability to go to 0 for consistent selection. Thus, you want $\frac{2\beta_*^2 + \delta_n^2 \log 2}{\delta_n^2 \log(d-k)}$ to go to 1. Thus, if β_* is large, that's a good thing! This is a lower bound. How large should

β_* be? It needs to be at least of order $\delta_n^2 \log(d-k)$ to get consistent selection. So β_* should be of order $\delta\sqrt{\frac{\log(d-k)}{n}}$, and we want $d/k \to \infty$. Thus, in order to recover any signal, the β-min condition is required to recover, β_* is the smallest non-zero entry.

The way to prove this is via Fano's lemma.

Lemma 9.3. *Fano's lemma.*
You have a parameter set and you want to establish a lower bound. This is a covering type of argument. If two probability measures are sufficiently different then we can distinguish between them. If KL divergence is small, it's hard to distinguish. For any given pair, we want to be able to distinguish between the two of them. There are two determining factors. One is how close are any two points. The other factor is how many points there are in total. We kind of want them to be spaced apart. So Fano's lemma says the following: Suppose we have random variable Z, and we have probability measures $P_1, \cdots, P_m$ for Z. Then Kullback-Leibler distance gives $KL(P_j, P_k) = \mathbb{E}_{P_j}\left[\log \frac{dP_j}{dP_k}\right]$. Then, let $\delta(z) \to [m]$ be the function telling us which probability measure the sample comes from. Fano's lemma tells us the probability we are correct.

$$\frac{1}{m}\sum_{j=1}^{m} \mathbb{P}_j\left\{\delta(z) = j\right\} \leq \frac{\frac{1}{m^2}\sum_{1\leq j,k\leq m} KL(P_j, P_k) + \log 2}{\log(m-1)}$$

This is a classical tool for establishing minimax lower bounds.

So how do we prove selection lower bounds? We just need to find the m probability measures. The z here are just our x, y, what we observed. All these probability measures are induced by β. So we need to figure out m different sets of β so that each of them are sufficiently similar so that I cannot tell. Then apply Fano's lemma.

In this case, it is simple to construct all these β. We can draw a line from $1 \to d$. First, we fix the last $k-1$ to be just β_*. Then look at the first $d-k+1$ of them. We define P_ℓ as the case where $\beta_j = \beta_*$ for $j = \ell$ and 0 otherwise. Keeping in mind that y is normally distributed with respect to the noise with mean $X\beta$, $\mathrm{KL}(P_j, P_k) = \frac{\|X_j\beta_* - X_k\beta_*\|_2^2}{2\sigma^2} \leq 2n\beta_*^2$. Then you use Fano's lemma, and you get the result with $\log d - k$ since you have $m = d - k + 1$.

In order to get upper bounds, you will need conditions on the design matrix. To get lower bounds, you typically don't except for the case of prediction.

Also, we have so far been only looking for perfect recovery. What if we allow for some error? What if we change the metric? That's still a subject of research. What's the minimax lower bound under a general loss function?

9.4 Upper bounds

Now we would like to get upper bounds on the estimation, prediction, and support errors. We will use the Lasso algorithm. We want to optimize

$$\min_\beta \frac{1}{2n}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

which gives you the Lasso estimate $\hat{\beta}(\lambda)$. The biggest advantage of this approach is that this is a convex optimization problem. $L_\lambda(\beta)$ is convex, and is also separable. It also has disadvantages. The main disadvantage of using Lasso is biased-ness. Why is it biased? The ℓ_1 penalty is $\sum |\beta_j|$. It doesn't matter if β_j is large or small, you penalize them the same way. For small ones, it is probably helpful. When β_j is close to 0, you're better off just shrinking it to 0. If your β_j is large, you're pretty sure it's not zero – but you're still shrinking it. This the cause of bias: You don't take the magnitude of the coefficient into account. The other disadvantage is the fact that there is very little known about $\hat{\beta}_j$ – it's hard to make a specific estimate about a specific coordinate. If you want to make inference about β_j, it seems it is not a good starting point. Say you want to construct a confidence interval for β_j, how do you do it? You start with a unbiased estimate, calculate standard deviation, and construct the interval. But here, you don't have the variance or unbiasedness. You also cannot run Lasso one time and develop a confidence statement for β_j.

Let's explore the advantage of Lasso, and see what kind of properties we have. Basically, we have $0 \in \partial L_\lambda(\hat{\beta})$. Here, $\partial f(x) = \{v : f(x+h) - f(x) \geq v^T h\}$. Consider linear regression: $f(\beta) = \|y - X\beta\|^2$, then $\partial f(\beta) = -2X^T(y - X\beta)$. Another example, suppose $f(\beta) = |\beta|$. Then this is not differentiable. We have

$$\partial f(\beta) = \begin{cases} -1 & \beta < 0 \\ [-1, 1] & \beta = 0 \\ 1 & \beta > 0 \end{cases}$$

Then,

$$\partial L_\lambda(\beta) = -\frac{1}{n} X^T(y - X\beta) + \lambda \text{sgn}(\beta)$$

where $\text{sgn}(\beta_j) = \partial|\beta_j|$.

10 Lecture 8: Low-rank Matrix Estimation

We would like to recover a low-rank matrix from measurements. There are several different settings:

(a) Standard regression setting. You have a matrix $M \in \mathbb{R}^{m_1 \times m_2}$, which is unknown to us. Then, we have a measurement matrix $X \in \mathbb{R}^{m_1 \times m_2}$. In the simplest case, X's entries are standard Gaussians. Then, the output is $Y = \langle M, X \rangle + \eta$, where the inner product is taken in Euclidean space (treating the matrices as vectors) and η is random noise. After observing i.i.d. copies $(X_1, Y_1), \cdots, (X_n, Y_n)$, we want to recover M.

This is not much different from the sparse regression problem: The only difference is we have a low-rank assumption on M, rather than a sparse assumption on M. Actually, if you apply the techniques from the sparse regression setting, they will work here as well: You can naturally generalize RIP and restricted eigenvalue to the low-rank setting as well.

(b) Matrix completion setting.

10.1 Matrix Completion

10.1.1 The Netflix Problem

Ten years ago, Netflix publicized a dataset of users $\times$ movies matrix of rating data: The goal is to predict users' ratings for movies they have not yet seen. We have a rating matrix $M \in \mathbb{R}^{m_1 \times m_2}$, where m_1 is the number of users and m_2 is the number of movies. Every entry is assumed to be non-negative (in particular, ratings may be integers from $1 to 5$). The entry $M(i_1, i_2)$ denotes the rating of movie i_2 from user i_1. We only observe a subset of the entries. Let $\Omega \subset [m_1] \times [m_2]$ denote the subset we see. The goal is to recover M from Ω (fill in the unobserved entries).

The winner of Netflix contest won with an algorithm called collaborative filtering. Every user i_1 is assumed to be characterized by a feature vector $r_i \in \mathbb{R}^k$. Every movie i_2 is also characterized by a feature vector $c_j \in \mathbb{R}^k$. Then, they assume that the rating $M(i_1, i_2) = \langle r_{i_1}, c_{i_2} \rangle$. Then, the goal is to solve the optimization problem

$$\min_{r,c} \sum_{(i_1,i_2)\in\Omega} (M(i_1, i_2) - \langle r_{i_1}, c_{i_2} \rangle)^2$$

They solve the optimization problem by gradient descent. You can choose k by cross-validation. This formulation is equivalent to solving the following problem:

$$\min_{X \in \mathbb{R}^{m_1 \times m_2}, \text{rank}(X) \leq k} \|(M - X)_\Omega\|_F^2$$

Note that this is a non-convex program, which is in general difficult to compute. In recent years, there's a lot of important progress towards solving this problem. For the first formulation, some people have proven that every local optima is a global optima.

10.1.2 Quantum State Tomography

One fundamental problem is to estimate the state of a quantum system in quantum computing. You may have a single qubit: an electron, or an ion. Each qubit has a state given by a matrix $\rho \in H_2$, called the **density matrix**, which is the set of 2×2 Hermitian matrices (ρ is conjugate). This is important for quantum computing. Some properties of ρ: It is positive semidefinite and its trace is 1. These properties are just a way of imposing the notion of a probability distribution over matrices.

The measurement of a single qubit is written as a matrix $X \in H_2$. We would like to measure the density matrix. If $X = \lambda_0 P_0 + \lambda_1 P_1$ (called spectral decomposition), by measuring X on density matrix ρ, the output Y is a random variable and the distribution of Y has form

$$\mathbb{P}\{Y = \lambda_0\} = \langle \rho, P_0 \rangle$$
$$\mathbb{P}\{Y = \lambda_1\} = \langle \rho, P_1 \rangle$$

Thus, $\mathbb{E}[Y] = \langle \rho, X \rangle$.

Definition 10.1. Pauli measurement.
There are four Pauli measurements:

(a) $\sigma_0 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

(b) $\sigma_x = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

(c) $\sigma_y = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & i \\ -i & 0 \end{bmatrix}$.

(d) $\sigma_z = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$.

These constitute an orthogonal basis.

Since the Pauli measurements are an orthogonal basis, estimating ρ is just a matrix completion problem. If there are b qubits, then the density matrix $\rho \in H_{2^b}$. Thus, the Pauli measurements become the tensorizations of the Pauli measurements: $\{\sigma_1 \otimes \sigma_2 \otimes \cdots \otimes \sigma_b : \sigma_i \in \{\sigma_0, \sigma_x, \sigma_y, \sigma_z\}\}$. There are 4^b different Pauli measurements.

10.1.3 Statistical Properties of Matrix Completion

An unknown matrix $M \in \mathbb{R}^{m_1 \times m_2}$ has $\text{rank}(M_0) = r_0$. Let $B(m_1, m_2)$ be an orthonormal basis of $\mathbb{R}^{m_1 \times m_2}$.

Example 10.2. $B(m_1, m_2) = \{e_{i_1} \otimes e_{i_2} : i_1 \in [m_1], i_2 \in [m_2]\}$.

Example 10.3. $B(m_1, m_2) = \{\sigma_1 \otimes \sigma_2 \otimes \cdots \otimes \sigma_b : \sigma_i \in \{\sigma_0, \sigma_x, \sigma_y, \sigma_z\}\}$, the Pauli basis.

Basically, the setting is fixed w.r.t. an orthonormal basis. Measurements are taken according to the basis.

Now, let (X, Y) be a random sample such that X is uniformly sampled from $B(m_1, m_2)$ then $\mathbb{E}[Y] = \langle M_0, X \rangle$. The data is $(X_1, Y_1), \cdots, (X_n, Y_n)$ i.i.d. copies of (X, Y). Our goal is to estimate matrix M_0.

Consider the expectation $\mathbb{E}[YX] = \mathbb{E}_X[\langle M_0, X \rangle X]$. Since X has a uniform distribution over orthonormal basis: So, we can write

$$\mathbb{E}_X[\langle M_0, X \rangle X] = \frac{1}{m_1 m_2} \sum_{B \in B(m_1, m_2)} \langle M_0, B \rangle B = \frac{1}{m_1 m_2} M_0$$

since B is an orthonormal basis. Therefore, $m_1 m_2 YX$ is an unbiased estimator of M_0. We can then average over all data points. Consider

$$M_{\text{data}} := \frac{m_1 m_2}{n} \sum_{i=1}^{n} Y_i X_i$$

as our unbiased estimator. We now want to take into account the rank constraint.

We begin with estimator 1 as a rank-constrained estimator: We minimize

$$\frac{1}{n}\sum_{i=1}^{n}\left(\langle M, X_i\rangle - Y_i\right)^2$$

subject to $\text{rank}(M) \leq k$, where we assume k is pre-determined. Because we are looking for low-rank solutions, we have to restrict M: But how do we do this? This is nonconvex. Let's modify it by first changing the objective function. We can rewrite it as

$$\frac{1}{n}\sum_{i=1}^{n}\langle M, X_i\rangle^2 - \frac{2}{n}\sum_{i=1}^{n} Y_i\langle M, X_i\rangle$$

(the last term doesn't matter because it doesn't depend on M). By central limit theorem, we have that $\frac{1}{n}\sum_{i=1}^{n}\langle M, X_i\rangle^2$ should concentrate, since these are independent random variables. We have

$$\mathbb{E}\left[\langle M, X\rangle^2\right] = \frac{1}{m_1 m_2}\sum_{B\in B(m_1,m_2)}\langle M, B\rangle^2 = \frac{\|M\|_F^2}{m_1 m_2}$$

because B is orthonormal bases. Thus, we can instead try to minimize

$$\frac{\|M\|_F^2}{m_1 m_2} - \frac{2}{n}\sum_{i=1}^{n} Y_i\langle M, X_i\rangle$$

It is equivalent to minimize

$$\left\| M - \frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i \right\|_F^2$$

such that $\text{rank}(M) \leq k$. The solution has an explicit form $\hat{M}_k = SVD_k\left(\frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i\right)$, where $SVD_k(A) = \sum_{i=1}^{k}\sigma_i(A)u_i v_i^T$. So this is good for computation, for instance $\mathcal{O}(km_1m_2)$. Tropp has also given methods of order $\mathcal{O}(m_1 m_2 \log k)$. How good is it statistically?

So first, what is an upper bound for $\|\hat{M}_k - M_0\|_F^2$? We have that $\text{rank}(\hat{M}_k - M_0) \leq \text{rank}(\hat{M}_k) + \text{rank}(M_0) \leq k + r_0$. Thus, $\|\hat{M}_k - M_0\|_F \leq \sqrt{k + r_0}\|\hat{M}_k - M_0\|_{\text{op}}$. Thus, we only need to bound the operator norm of the matrix. We have

$$\begin{aligned}
\|\hat{M}_k - M_0\|_{\text{op}} &= \left\| SVD_k\left(\frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i\right) - M_0 \right\|_{\text{op}} \\
&\leq \left\| \frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i - M_0 \right\|_{\text{op}} + \left\| SVD_k\left(\frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i\right) - \frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i \right\|_{\text{op}} \\
&= \left\| \frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i - M_0 \right\|_{\text{op}} + \sigma_{k+1}\left(\frac{m_1 m_2}{n}\sum_{i=1}^{n} Y_i X_i\right)
\end{aligned} \tag{3}$$

Then by Weyl's theorem, we have $\sigma_k(A+B) \leq \sigma_1(A) + \sigma_k(B)$. Therefore,

$$\sigma_{k+1}\left(\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i\right) = \sigma_{k+1}\left(\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i - M_0 + M_0\right) \leq \sigma_{k+1}(M_0) + \sigma_1(\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i)$$

Then, if $k \geq r_0$, we have $\sigma_{k+1}(M_0) = 0$. Now, we assume that $k \geq r_0$. Thus,

$$\|\hat{M}_k - M_0\|_{\text{op}} \leq 2\|\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i\|_{\text{op}}$$

where the latter term is an unbiased estimator for the matrix M_0. What can we conclude? If $k \geq r_0$, then

$$\|\hat{M}_k - M_0\|_F \leq 2\sqrt{k + r_0}\left\|\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i\right\|_{\text{op}}$$

Now, what is the operator norm of the unbiased estimator? Now, we need to introduce one of the most important tools for analyzing random matrices in statistics: Matrix-Bernstein inequality. This inequality allows us to control the operator norm of $Z_1 + \cdots + Z_n$, where Z_i are random matrices.

Theorem 10.4. *Matrix-Bernstein inequality (bounded version, Tropp 2012).*
Suppose that $Z_1, \cdots, Z_n \in \mathbb{R}^{m_1 \times m_2}$ are independent with $\mathbb{E}[Z_j] = 0$. Suppose that $\|Z_j\|_{\text{op}} \leq U$ a.s. Let $\sigma^2 = \max\{\|\sum_{j=1}^n \mathbb{E}\left[Z_j Z_j^T\right]\|_{\text{op}}, \|\sum_{j=1}^n \mathbb{E}\left[Z_j^T Z_j\right]\|_{\text{op}}\}$. Then for all $t > 0$, with probability at least $1 - e^{-t}$,

$$\|Z_1 + Z_2 + \cdots + Z_n\|_{\text{op}} \leq 2\max\left[\sigma\sqrt{t + \log(m_1 + m_2)}, U(t + \log(m_1 + m_2))\right]$$

Now let's try to apply this bounded version of Matrix-Bernstein to this problem. Let $Z_j = \frac{m_1 m_2}{n} Y_i X - \frac{M_0}{n}$, then $\mathbb{E}[Z_j] = 0$. Now we will assume that if Y is uniformly bounded by some U almost surely, then $\|Z_j\|_{\text{op}} \leq \frac{m_1 m_2}{n} U + \frac{\sqrt{m_1 m_2} U}{n}$, noting that $\mathbb{E}[Y] = \langle M, X\rangle$ and thus since $|Y| \leq U$ and X is a basis, we have $|\langle M, X\rangle| \leq U \implies \|M\|_F \leq \sqrt{m_1 m_2} U$ (and keeping in mind that $\|M\|_{\text{op}} \leq \|M\|_F$). Thus, we can upper bound $\|Z_j\|_{\text{op}} \leq \frac{m_1 m_2 U}{n}$.

Then, we need

$$\|\mathbb{E}\left[ZZ^T\right]\|_{\text{op}} = \|\mathbb{E}\left[\frac{m_1^2 m_2^2}{n^2} Y^2 X X^T - \frac{M_0 M_0^T}{n^2}\right]\|_{\text{op}}$$

$$\leq \frac{m_1^2 m_2^2}{n^2}\left\|\mathbb{E}\left[Y^2 X X^T\right]\right\|_{\text{op}} + \frac{m_1 m_2 U^2}{n^2}$$

Now we need to get $\|\mathbb{E}\left[Y^2 X X^T\right]\|_{\text{op}} = \sup_{u \in \mathbb{R}^{m_1}, \|u\|_2 \leq 1} \mathbb{E}\left[\langle Y^2 X X^T, u \otimes u\rangle\right]$. We have $|Y| \leq U$ a.s., so this expression is

$$\leq U^2 \sup_u \mathbb{E}\left[X X^T, u \otimes u\right] = U^2 \sup_u \|X^T u\|_2^2 = U^2 \sup_u \sum_{i=1}^{m_2} \mathbb{E}\left[\langle X, u \otimes e_i\rangle^2\right]$$

$$= U^2 \sup_{u \in \mathbb{R}^{m_1}} \sum_{i=1}^{m_2} \frac{\|u \otimes e_i\|_F^2}{m_1 m_2} = \frac{U^2 m_2}{m_1 m_2} = \frac{U^2}{m_1}$$

Thus,

$$\frac{m_1^2 m_2^2}{n^2} \left\| \mathbb{E}\left[Y^2 X X^T\right] \right\|_{\text{op}} + \frac{m_1 m_2 U^2}{n^2} \leq \frac{m_1 m_2^2 U^2}{n^2} + \frac{m_1 m_2 U^2}{n^2} \leq 2\frac{m_1 m_2^2 U^2}{n^2}$$

We also need to bound σ^2: I'll skip the steps for now, and just state

$$\sigma^2 \leq 2 \max\{\frac{m_1 m_2^2 U^2}{n}, \frac{m_1^2 m_2 U^2}{n}\}$$

Finally, applying Matrix-Bernstein with these bounds gives

$$\left\| \frac{m_1 m_2}{n} \sum_{i=1}^{n} Y_i X_i - M_0 \right\|_{\text{op}} \leq U \max\left\{ 2\sqrt{\frac{m_1 m_2 (m_1 + m_2)(t + \log(m_1 + m_2))}{n}}, \frac{m_1 m_2 (t + \log(m_1 + m_2))}{n} \right\}$$

interpolating between the cases where $m_1 > m_2$ and vice-versa.

If $n \geq (m_1 + m_2)(t + \log(m_1 + m_2))$, we get the first term of the max function as the smaller upper bound. Then in this case, we get

$$\frac{\|\hat{M}_k - M_0\|_F}{\sqrt{m_1 m_2}} \leq 4U \sqrt{\frac{(m_1 + m_2)(k + r_0)(t + \log(m_1 + m_2))}{n}}$$

holds with probability at least $1 - e^{-t}$. The term on the left-hand side is the average Frobenius norm. Note that $(m_1 + m_2)(k + r_0)$ is essentially the degrees of freedom. To summarize the conditions required for this to work again:

(a) $\|Y\| \leq U$ a.s. (for Bernstein)

(b) $k \geq r_0$

(c) $n \geq (m_1 + m_2) \log(m_1 + m_2)$

We still haven't figured out how to determine k, if the rank is unknown. Are there more efficient ways? This is why some people consider the matrix nuclear norm, which is a convex relaxation for the rank function. We may not have time to cover that right now.

Let me introduce an unbounded version of the Matrix Bernstein inequality.

Theorem 10.5. *Matrix-Bernstein inequality (unbounded version, Kultchinskii,* 2012*). Suppose that $Z_1, \cdots, Z_n \in \mathbb{R}^{m_1 \times m_2}$ are independent with $\mathbb{E}[Z_j] = 0$. Suppose that $\| \|Z_j\|_{\text{op}} \|_{\psi_\alpha} \leq U_\alpha$ for $\alpha \geq 1$, where $\|\eta\|_{\psi_\alpha} = \inf\left\{ \mathbb{E}\left[\exp(\frac{|\eta|^\alpha}{c^\alpha} \right] \leq 2 \right\}$ (sub-Gaussian if $\|\eta\|_{\psi_2} < \infty$ or sub-exponential tail if $\|\eta\|_{\psi_1} < \infty$). Let $\sigma^2 = \max\{\| \sum_{j=1}^n \mathbb{E}\left[Z_j Z_j^T\right] \|_{\text{op}}, \| \sum_{j=1}^n \mathbb{E}\left[Z_j^T Z_j\right] \|_{\text{op}}\}$. Then for all $t > 0$, with probability at least $1 - e^{-t}$,*

$$\|Z_1 + Z_2 + \cdots + Z_n\|_{\text{op}} \leq C_1 \max\left[\sigma\sqrt{t + \log(m_1 + m_2)}, U_\alpha \log^{(1/\alpha)}\left(\frac{U_\alpha}{\sigma}\right)(t + \log(m_1 + m_2)) \right]$$

11 Lecture 9: Low-rank Matrix Estimation Continued

First, note that one requires symmetric matrices to apply matrix Bernstein. However, if you have a non-symmetric matrix X, note that you can apply the dilation property and write $\begin{bmatrix} 0 & X \\ X^T & 0 \end{bmatrix}$. Then apply Matrix-Bernstein to this matrix instead.

Now, suppose X is a symmetric matrix. Let $f : \mathbb{R} \to \mathbb{R}$ be a function. We define $f(X) = Uf(\Sigma)U^T$ where $X = U\Sigma U^T$ (eigen-decomposition).

11.1 Proving Matrix Bernstein

Let's give a sketch of the proof of Matrix-Bernstein (Tropp, 2012) from last time.

Proof. Let $\lambda_{\max}$ be the largest singular value operator. Then for $\theta > 0$,

$$\begin{aligned}
\mathbb{P}\left\{\lambda_{\max}(\sum_{j=1}^n Z_j) \geq t\right\} &= \mathbb{P}\left\{\lambda_{\max}(\theta\sum_{j=1}^n Z_j) \geq \theta t\right\} \\
&= \mathbb{P}\left\{e^{\lambda_{\max}(\theta\sum_{j=1}^n Z_j)} \geq e^{\theta t}\right\} \\
&\leq e^{-\theta t}\mathbb{E}\left[e^{\lambda_{\max}(\theta\sum_{j=1}^n Z_j)}\right] \\
&= e^{-\theta t}\mathbb{E}\left[\lambda_{\max}(e^{\sum_{j=1}^n \theta Z_j})\right] \\
&\leq e^{-\theta t}\mathbb{E}\left[\mathrm{Tr}(e^{\sum_{j=1}^n \theta Z_j})\right]
\end{aligned} \tag{4}$$

Thus, we would now like to obtain an upper bound on $\mathbb{E}\left[\mathrm{Tr}(e^{\sum_{j=1}^n \theta Z_j})\right]$. Here we will need a very important lemma by Lieb:

Lemma 11.1. *(Lieb).*
Let H be a fixed symmetric matrix. Let X be a random symmetric matrix. Then

$$\mathbb{E}\left[\mathrm{Tr}(e^{H+X})\right] \leq \mathrm{Tr}(e^{H+\log\mathbb{E}\left[e^X\right]})$$

This is kind of similar to Jensen's inequality, but more powerful. There are some simple proofs for this inequality.

We can write

$$\begin{aligned}
\mathbb{E}\left[\mathrm{Tr}(e^{\sum_{j=1}^n \theta Z_j})\right] &= \mathbb{E}\left[\mathrm{Tr}(e^{\sum_{j=1}^{n-1} \theta Z_j + \theta Z_n})\right] \\
&= \mathbb{E}_{Z_1,\cdots,Z_{n-1}}\left[\mathbb{E}_{Z_n}\left[\mathrm{Tr}(e^{\sum_{j=1}^{n-1} \theta Z_j + \theta Z_n})\right]\right] \\
&\leq \mathbb{E}_{Z_1,\cdots,Z_{n-1}}\left[\mathrm{Tr}(e^{\sum_{j=1}^{n} \theta Z_j + \log\mathbb{E}_{Z_n}\left[e^{\theta Z_n}\right]})\right]
\end{aligned} \tag{5}$$

Then, you can apply this lemma recursively, peeling off one-term at a time to eventually obtain

$$\mathbb{E}\left[\mathrm{Tr}(e^{\sum_{j=1}^n \theta Z_j})\right] \leq \mathrm{Tr}(e^{\sum_{j=1}^n \log \mathbb{E}\left[e^{\theta Z_j}\right]})$$

Then we can apply this upper bound to the probability:

$$\mathbb{P}\left\{\lambda_{\max}(\sum_{j=1}^n Z_j) \geq t\right\} \leq e^{-\theta t}\mathrm{Tr}(e^{\sum_{j=1}^n \log \mathbb{E}\left[e^{\theta Z_j}\right]}) \leq e^{-\theta t} \cdot m \cdot \lambda_{\max}\left(e^{\sum_{j=1}^n \log \mathbb{E}\left[e^{\theta Z_j}\right]}\right)$$

Since the trace is less than the number of terms on the diagonal times the maximum singular value,

$$= me^{-\theta t}e^{\lambda_{\max}(\sum_{j=1}^n \log \mathbb{E}\left[e^{\theta Z_j}\right])}$$

Now, we use the following lemma:

Lemma 11.2. *If* $\mathbb{E}[Z_j] = 0$ *and* $\|Z_j\|_{op} \leq 1$, *then* $\mathbb{E}\left[e^{\theta Z_j}\right] \leq e^{(e^\theta - \theta - 1)\mathbb{E}\left[Z_j^2\right]}$.

Proof. Note that $f_\theta(x) = \frac{1}{x^2}(e^{\theta x} - \theta x - 1)$ is an increasing function for $0 \leq x \leq 1$. □

Now, we can apply the lemma to the RHS of the previous bound:

$$\begin{aligned}\mathbb{P}\left\{\lambda_{\max}(\sum_{j=1}^n Z_j) \geq t\right\} &\leq me^{-\theta t}e^{\lambda_{\max}(\sum_{j=1}^n (e^\theta - \theta - 1)\mathbb{E}\left[Z_j^2\right])} \\ &= me^{-\theta t + (e^\theta - \theta - 1)\lambda_{\max}(\sum_{j=1}^n Z_j^2)} \\ &= me^{-\theta t + (e^\theta - \theta - 1)\sigma^2}\end{aligned} \tag{6}$$

where $\sigma^2 = \lambda_{\max}(\sum_{j=1}^n Z_j^2)$. Then minimizing over θ gives the bound

$$\mathbb{P}\left\{\lambda_{\max}(\sum_{j=1}^n Z_j) \geq t\right\} \leq \begin{cases} me^{-3t^2/8\sigma^2} & t \leq \frac{\sigma^2}{U} \\ me^{-3t/8U} & t \geq \frac{\sigma^2}{U} \end{cases}$$

Note that in order to get the U in there, we need to apply the previous lemma to UZ_j – this will result in θ/U appearing everywhere, yielding the desired results. □

11.2 Back to Matrix Completion

Now we return to the matrix completion problem. Let $M_0 \in \mathbb{R}^{m_1 \times m_2}$, $\mathrm{rank}(M_0) \leq r_0$. Let $\mathcal{X} \subset \mathbb{R}^{m_1 \times m_2}$ be an orthonormal basis of $\mathbb{R}^{m_1 \times m_2}$. Let X be uniformly sampled from $\mathcal{X}$, and take output $Y : \mathbb{E}[Y] = \langle M_0, X\rangle$.

Given $(X_1, Y_1), \cdots, (X_n, Y_n)$, our goal is to recover M_0. Last time we introduced the low-rank projection estimator. We had

$$\hat{M}_k = \mathrm{Proj}_k\left(\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i\right)$$

as the solution to minimizing over M, subject to $\text{rank}(M) \leq k$, $\|\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i - M\|_F^2$. We also showed

$$\frac{\|\hat{M}_k - M_0\|_F}{\sqrt{m_1 m_2}} \leq C \cdot U\sqrt{\frac{(m_1+m_2)K(t+\log(m_1+m_2))}{n}}$$

if $|Y| \leq U$ a.s. and $n \geq (m_1+m_2)\log(m_1+m_2)$, $k \geq r_0$. However, this estimator is optimal only when $k = r_0$. In other words we need to know the true rank.

11.3 Nuclear Norm estimator

Here, we introduce a different estimator based on the matrix nuclear norm, which is actually similar to the ℓ_1 norm in compressed sensing. The true rank r_0 is unknown. So $\hat{M}_k$ is not statistically optimal if $k \neq r_0$.

We have

$$\hat{M}_\lambda = \text{argmin}_{M \in \mathbb{R}^{m_1 \times m_2}} \left\| \frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i - M \right\|_F^2 + \lambda \|M\|_*$$

Let the objective function be denoted $f_\lambda(M)$.

Definition 11.3. Matrix Nuclear Norm $\|\cdot\|_*$.

$$\|M\|_* = \sum_{i=1}^{\min(m_1,m_2)} \sigma_i(M)$$

where σ_i are the ordered singular values of M. Equivalently,

$$\|M\|_* = \sup_{\|T\|_{op} \leq 1} \langle M, T \rangle$$

We can easily show it's a convex function (it's a norm). We can write

$$\|\lambda_1 M_1 + \lambda_2 M_2\|_* = \sup_{\|T\|_{op}\leq 1} \langle \lambda_1 M_1 + \lambda_2 M_2, T\rangle \leq \sup_{\|T\|_{op}} \lambda_1 \langle M_1, T\rangle + \sup_{\|T\|_{op}} \lambda_2 \langle M_2, T\rangle = \lambda_1 \|M_1\|_* + \lambda_2 \|M_2\|_*$$

Then, both terms in the objective are convex: Indeed, the first term is strongly convex. Thus, there is a unique solution and furthermore there are fast algorithms to find it. So $\hat{M}_\lambda$ is a global minima of $f_\lambda(M)$. However, this function isn't smooth, so we need to look at subgradients.

11.3.1 Sub-gradient for $f_\lambda(M)$

We want $0 \in \partial f_\lambda(\hat{M}_\lambda)$ as an optimality condition.

Definition 11.4. If $f : \mathbb{R}^m \to \mathbb{R}^t$ is convex, then the subgradient of f at $x \in \mathbb{R}^m$ is defined as

$$\partial f(x) = \{y \in \mathbb{R}^m : f(z) - f(x) \geq \langle y, z - x\rangle \forall z \in \mathbb{R}^m\}$$

This is a generalization of the definition of the gradient. Let us now calculate $\partial\|M\|_*$:

Lemma 11.5. *For $M \in \mathbb{R}^{m_1 \times m_2}$, rank$(M) \leq r$. Then*

$$\partial\|M\|_* = \left\{UV^T + U_\perp W V_\perp^T : \|W\|_{op} \leq 1\right\}$$

If $M = UD_mV^T$, where $U \in \mathbb{R}^{m_1 \times r}$, $V \in \mathbb{R}^{m_2 \times r}$ and $U_\perp$ and $V_\perp$ are the orthogonal complements of U and V respectively (the components which make up the full SVD). Also recall that $\text{Proj}_{U_\perp} = U_\perp U_\perp^T$.

Proof. We will show set containment in both directions. For now, we will only show that every element of the set defined in the lemma belongs to the subgradient.

Consider a matrix $W \in \mathbb{R}^{m_1 \times m_2}$, $\|W\|_{op} \leq 1$, we show that $UV + P_{U_\perp} W P_{V_\perp}$ is indeed a subgradient. For any matrix $Y \in \mathbb{R}^{m_1 \times m_2}$,

$$\langle Y - M, UV + P_{U_\perp} W P_{V_\perp}\rangle = \langle Y, UV + P_{U_\perp} W P_{V_\perp}\rangle - \langle M, UV + P_{U_\perp} W P_{V_\perp}\rangle$$

$$\leq \|Y\|_* \|UV^T + P_{U_\perp} W P_{V_\perp}\|_{op} - \langle UD_mV^T, UV^T + P_{U_\perp} W P_{V_\perp}\rangle = \|Y\|_* - \|M\|_*$$

by Holder inequality, and noting that the second term is equal to $\|M\|_*$.

The second direction is by contradiction, but it is more involved. □

So, the subgradient for $f_\lambda(M)$ is given by

$$\partial f_\lambda(M) = \left\{2(M - \frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i) + \lambda(UV^T + P_{U_\perp} W P_{V_\perp}) : \|W\|_{op} \leq 1\right\}$$

Since $\hat{M}_\lambda$ is the global minimizer, then $0 \in \partial f_\lambda(\hat{M}_\lambda)$. Actually, we can show that $\hat{M}_\lambda = S^+_{\lambda/2}\left(\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i\right)$, where S^+_λ is the soft-thresholding function:

Definition 11.6. Soft-thresholding function.

$$S^+_{\lambda/2}(x) = \max\{x - \lambda/2, 0\}$$

11.3.2 Optimality of $\hat{M}_\lambda$

Now we can just show that 0 will be in the subgradient.

Let $T = \frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i = UDV^T$. Let $\hat{M}_\lambda = US^+_{\lambda/2}(D)V^T$, and then plug everything into the formula for the subgradient. Then $\hat{M}_\lambda - T = \sum_{j:\sigma_j(T) \geq \lambda/2}(-\lambda/2)u_j(T)v_j^T(T) - \sum_{j:\sigma_j(T) < \lambda/2}\sigma_j(T)u_j(T)v_j^T(T)$. Then,

$$\partial\|\hat{M}_\lambda\|_* = \left\{\sum_{j:\sigma_j(T) \geq \lambda/2} u_j(T)v_j^T(T) + P_{U_\perp} W P_{V_\perp}\right\}$$

where we write $U = [u_1, \cdots, u_j]$ such that $\sigma_j(T) \geq \lambda/2$, and V likewise. Then you can see that you can cancel by choosing W with operator norm ≤ 1. Other terms will already cancel.

11.3.3 Statistical Performance of $\hat{M}_\lambda$

We have

$$\hat{M}_\lambda = S^+_{\lambda/2}\left(\frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i\right)$$

Theorem 11.7. *Statistical Performance.*
If $\lambda \geq 2\left\|M_0 - \frac{m_1 m_2}{n}\sum_{i=1}^n Y_i X_i\right\|_{op}$, *then*

$$\left\|\hat{M}_\lambda - M_0\right\|_F \leq \lambda\sqrt{2\text{rank}(M_0)}$$

This is a deterministic theorem, but we have a random term – thus, we want to apply Matrix-Bernstein inequality.

Theorem 11.8. *If* $\lambda = 2U\sqrt{\frac{m_1 m_2(m_1+m_2)(t+\log(m_1+m_2))}{n}}$, *then with probability* $1-e^{-t}$, *you have*

$$\frac{1}{\sqrt{m_1 m_2}}\|\hat{M}_\lambda - M_0\|_F \leq 4U\sqrt{\frac{(m_1+m_2)\text{rank}(M_0)(t+\log(m_1+m_2))}{n}}$$

when $n \geq (m_1+m_2)\log(m_1+m_2)$.

Thus here, we did not need to know the true rank. We only needed to know the uniform upper bound U.

To summarize, there is a huge literature on matrix completion. Here I did not pose any condition on incoherence – originally, they assumed incoherent type conditions. This kind of condition is useful for establishing exact matrix completion. In this case, we're not interested in exact recovery because we have stochastic errors here. We end up not needing incoherence conditions here.

12 Student Presentations: Sparse PCA

12.1 Inconsistency of regular PCA

This is the first paper that observed in high-dimension that you have interesting behavior – it's historically interesting. Your estimates in high-dimension are perpendicular to the truth, so it's useless! This articulates this behavior.

The thresholding approach in this paper is very simple: Look at diagonal elements and check how big they are. The other presentations are either relying on difficult estimators or SDP relaxations. This is the simplest approach which is easiest to implement.

12.1.1 On Consistency and Sparsity for Principal Components Analysis in High Dimensions

Basically, this title tells you everything about this paper. The problem is classical PCA will get into trouble under high dimensions, because it won't be a consistent estimator.

First, the model we consider is $x_i = v_i\rho + \sigma z_i$ for $i \in [n]$ where $\rho \in \mathbb{R}^p, v_i \sim \mathcal{N}(0,1), z_i \sim \mathcal{N}(0, I_p)$. ρ is the top principal component we want to get.

There's something called smoothed PCA:

$$\max_u \frac{\text{Var}(ux_i)}{\|u\|_2 + \lambda\|\theta^2 u\|_2}$$

We have $S = \frac{1}{n}\sum_{i=1}^n x_i x_i^T$, where $\hat{\rho}$ is an eigenvector of $\lambda_{\max}(S)$. Our distance matrix is the sin of $\hat{\rho}, \rho$. We have the assumption that $p_n/n \to c$ as $n \to \infty$. Then also $\lim_{n\to\infty} \frac{\|\rho\|^2}{\sigma^2} \to \omega > 0$ – your principal component is detectable.

Theorem 12.1. *Almost surely,*

$$\lim_n \cos^2(\mathcal{A}(\hat{\rho}_n, \rho)) = \frac{[\omega^2 - c]_+}{\omega^2 + c\omega}$$

When $\omega^2 \leq c$, i.e. $\lim_n \frac{p}{n}\frac{\sigma^4}{\|\rho\|^4} \geq 1$, then $\cos A(\hat{\rho}_n, \rho) \to 0$, and thus the sin goes to 1 – so the estimated principal component will be almost orthogonal to the true principal component!

They then propose a new estimator: What if you only consider a part of the ρ: The key idea is to use sparsity. Use $\lim_n \frac{k}{n}\frac{\sigma^4}{\|\rho_k\|^4}$, where ρ_k is a k-sparse subset. So, we assume some sparsity of the principal component. Namely, your principal component has a sparse representation in an orthogonal basis. Now, we want to only consider part of the principal component because we know it's sparse. How do we decide to threshold? Well, we can look at $\{v : \hat{\sigma}_v^2 \geq \sigma^2(1+\alpha_n), \hat{\sigma}_v^2 = \frac{1}{n}\sum_{i=1}^n x_{iv}^2\}$.

Theorem 12.2. *If* $\log(\max p, n)/n \to 0$*, and* $\|\rho_n\|$ *toL* > 0 *and each* ρ_n *is weak* ℓ_q *decay, then the distance between* $\hat{\rho}$ *and* ρ *goes almost surely to* 0*.*

Proof. l is the estimated subset. We have triangle inequality to decompose into $d(\hat{\rho}_l, \rho_l)$ and $d(\rho_l, \rho)$, where l is the chosen subset. The first term will go to zero because $\lim_n (k/n)(\sigma^4/\|\rho\|^4)$ and $k = o(n)$ goes to zero, by sparsity assumption on l. How about the second part? We can upper bound it by something else which will go to zero. □

The algorithm stated by the paper is as follows:

(a) First compute the basis for each x_i.

(b) Then calculate the sample variances and threshold the variance, using the median of the variances. This gives l.

(c) Do PCA on the submatrix inexed by set l.

(d) Then filter the estimator by hard thresholding – they don't prove this, it's just empirical.

(e) Then reconstruct to original signal domain.

12.2 Minimax Lower Bounds

12.2.1 Minimax sparse principal subspace estimation in high dimensions

The first paper discussed is "Minimax sparse principal subspace estimation in high dimensions", by Vincent Vu and Jing Lei from 2013. It's a pretty heavy paper. What I'm trying to do to present on time is skipping lots of details and presenting the result in a very special case.

In general, PCA is the problem of finding lower dimensional subspaces. We have observations $X_1, \cdots, X_n \in \mathbb{R}^p$. These are i.i.d. with unknown mean μ and covariance matrix Σ. We are looking for a lower subspace to project onto to get maximum variance on that subspace. If $G_{p,d}$ is the Grassmanian manifold of d-dimensional subspaces on $\mathbb{R}^p$. If we look at the eigenvalue decomposition of the covariance matrix: $\Sigma = \sum_{i=1}^p \lambda_i v_i v_i^T$, sorted $\lambda_1 \geq \cdots \geq \lambda_p$. We are looking for maximum variance on projection. This space is spanned by first d eigenvectors. The classical approach to this problem is to just look at the empirical mean of the covariance matrix and taking the SVD. The problem in high dimension is $\hat{v}_i$ is an inconsistent estimator. So we're going to add additional assumptions to the problem – namely sparsity, which I will define. Instead of solving the problem over the whole Grassmanian manifold, we will show it for only a part of the space. We will show for this definition of sparsity that we can get the optimal minimax rate.

We need a distance over the Grassmanian in order to compare the estimation with the true subspace. It is defined as follows: $\mathcal{E}, \mathcal{F} \in G_{p,d}$, and E, F orthogonal projections. Then we can look at $EF_\perp = E(I - F)$. If we look at the singular vluaes of this matrix, we'll get $s_1, \cdots, s_d$ – at most d are nonzero. If we sort them, then $\theta_i = \arcsin(s_i)$ are called "canonical angles" between E and F. We have $\Theta(\mathcal{E}, \mathcal{F}) = \text{diag}(\theta_1, \cdots, \theta_d)$. In the case of $d = 1$, it is the usual angle between two lines. The distance between $\mathcal{E}$ and $\mathcal{F}$ is defined as $\|\sin(\Theta(\mathcal{E}, \mathcal{F}))\|_F^2 = \|EF^\perp\|_F^2$. All results in this paper are presented in the square form. It can be shown that the square root of the distance is an actual metric on the Grassmanian. The Stiefel manifold, described by $\mathcal{V}_{p,d}$ is the space of all unitary matrices in $\mathbb{R}^{p \times d}$.

The notions of sparsity we will deal with are row-sparsity and column-sparsity.

Definition 12.3. Row sparsity.
If $A \in \mathbb{V}_{p,d}$, and you look at rows of A as $[a_{1*}, \cdots, a_{p*}]$, then

$$\|A\|_{2,0} = \|(\|a_{1*}\|_2, \cdots, \|a_{p*}\|_2)\|_0$$

Note that $\|A\|_{2,0} \leq R_0$, and it's invariance under multiplication by unitary matrix. We have that $d \leq R_0 \leq p$.

Definition 12.4. Column sparsity.
Analagously,

$$\|A\|_{*,0} \max_{1\leq j\leq d} \|a_{*j}\|_0$$

We want to solve the problem $\max \sum_{i=1}^n \|\Pi_{\mathcal{G}}\|^2$. This is equivalent to $\max\langle S_n, UU^T\rangle$ s.t. $U \in \mathbb{V}_{p,d}$. They give results for both $\|U\|_{2,0} \leq R_0$ or $\|v\|_{*0} \leq R_0$.

We also assume that our data is coming from the distribution $X = \mu + \Sigma^{1/2} Z_i$, where all Z_i are sub-gaussian variables. They also need some control on the effective noise variances, which are defined as following: They need a gap between $\lambda_1\lambda_{d+1}/(\lambda_d - \lambda_{d+1})^2 \leq \sigma^2$.

The minimax rate is given as following:

$$\inf_{\hat{S}} \sup_{P_0(\sigma^2, R_0)} d(\hat{S}, S) \geq c(R_0 - d)\frac{\sigma^2}{n}(d + \log\frac{Rd}{R_0 - d})$$

They argue this error happens because of variable selection. The d-term is the variance on each coordinate. Their claim is tha thtese rates are fundamental. They prove that any solution of the row-sparse problem, and we have $n \geq 36R_0(d + \log p)$, then

$$d(\hat{S}, S) \leq cR_0\frac{\sigma^2}{n}(d + \log p)$$

with probability $1 - \frac{4}{n-1} - \frac{6\log n}{n} - \frac{1}{p}$.

12.2.2 Sparse PCA: Optimal rates and adaptive estimation

12.3 Estimation under Polytime Computational Resources

12.3.1 Optimal detection of sparse principal components in high dimension

Today we're interested in finding the level at which we can do sparse PCA in the first place. I'm not going to focus on the estimation side of things; we'll just look at the threshold at which we can do it in the limit.

I'm going to talk about a special case, but you can extend things in the obvious way; i.e. normal $\approx$ sub-gaussian.

We can think of it as a hypothesis testing problem. We have as our null hypothesis as $H_0 : X \sim N(0, I_p)$, $H_1 : X \sim N(0, I_p + \theta vv^T)$. If we can tell these things apart, we can do PCA. If we can't tell them apart, doing PCA is kind of meaningless. We know that the empirical covariance $\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^n X_i X_i^T \to \Sigma$. We can look at the maximum eigenvalue of the estimation matrix $\lambda_{\max}(\hat{\Sigma})$. Classically, things work out. In high-dimension, $\lambda_{\max}(\hat{\Sigma}) \to (1 + \sqrt{\alpha})^2$ under the null a.s. if $p/n \to 0$, so it'll go to infinity. So you can't do this anymore in high dimensions.

What we're going to do instead is assume that vector v, $\|v\|_2 = 1$, is k-sparse. Then, what we're going to look at instead is the k-sparse maximum eigenvalue. This is defined by looking only at k-sparse norm 1 vector for maximizing over the quadratic form. We will

argue that we get separation between these quantities and thus a detection threshold with some low-probability of error in the sparse case, under both the null and the alternative.

It's usually easier to look at the alternative case first. We basically want to find a lower bound of this quantity. We can just choose some vector x and hope it's a good guess. Our good guess would be to choose v. Then we can get a lower bound

$$\lambda_{\max}^k(\hat{\Sigma}) \geq v^T\hat{\Sigma}v = \frac{1}{n}\sum_{i=1}^n (X_i^T v)^2 \sim \mathcal{N}(0, 1+\theta)$$

Using the usual concentration stuff, we'll get a bound of the form $\geq 1+\theta-2(1+\theta)\sqrt{\frac{\log(1/\delta)}{n}}$ with probability $1-\delta$.

Under the null hypothesis, if you get $\|u\|_2 = 1$, k-sparse. Let's say $u \in \mathbb{R}^k$, $S \subseteq [p]$, and let $\hat{u}_j = u_j$ if $j \in S$, 0 otherwise. In that case, we find that $\left(u^T\hat{\Sigma}_S u\right)^{-1} = \frac{1}{n}\Sigma_{i=1}^n((\hat{u}^T X_i)^2 - 1)$. We then end up wanting to take a union bound over all the choices of S. Then we do ϵ-net arguments, and take a union bound over vectors and only pay a constant price for doing so. Then under the null, you can get an upper bound

$$\lambda_{\max}^k(\hat{\Sigma}) \leq 1 + 4\sqrt{\eta} + \eta$$

where $\eta = \frac{k\log(9ep/r)+\log(1/\delta)}{n}$. And it turns out that this is minimax optimal.

It turns out you can reduce this problem to an NP complete problem. So the estimation problem is hard. However, you can relax it to solving the following SDP:

$$\lambda_{\max}^k(A) = \max \operatorname{Tr}(AZ)$$

such that $\operatorname{Tr}(Z) = 1, Z \succeq 0, |Z|_0 = k$. This is non-convex, but you can relax to look at $|Z|_1$ instead. The SDP is easy enough to analyze, and you get an upper bound of the form

$$\text{SDP}_k(\hat{\Sigma}) \leq 1 + 2\sqrt{\frac{k^2\log(4p^2/\delta)}{n}} + \cdots$$

So to summarize, we can show that $\lambda_{\max}^k(\hat{\Sigma})$ or $\text{SDP}_k(\hat{\Sigma})$ are $\leq 1 + C_\alpha\sqrt{\frac{r^\alpha\log(1/\delta)}{n}}$ with probability $1-\delta$.

So now the question is whether there's a computation and statistics tradeoff. There is – we cannot do better using an SDP type method, otherwise you'll end up breaking a conjecture to do with a certain graph detection problem. They have a paper which came out a year after this one where they go into more detail. They want to reduce down to what's known as the planted clique problem, and then argue that if you have this bound, you have the planted clique problem in a certain regime, and it's conjectured that there is no polytime algorithm to solve it. But because you're doing the SDP, you have a polytime algorithm.

This paper is quite important: Previous two presentations focused on minimax optimality, which is traditional in statistics. This took into account the computational component,

reducing to a computational conjecture – if this is how much computation you have, this is the best you can do. There's a lot of follow up work based on this. A lot of the follow-up work reduces to planted clique. The main value of this work is that there's a new way to think about minimax optimality, taking computational cost into consideration.

12.3.2 Do semidefinite relaxations solve sparse PCA up to the information limit?

This paper proves that SDP method can recover at the same level as diagonal thresholding for the first principal component. Their method shows that SDP method performs similarly to diagonal thresholding. In the end, they propose a covariance thresholding method which can recover well. Exhaustive search can reduce $n \geq k \log p$, no methods work $n \leq k \log p$. Diagonal thresholding works for $n \geq k^2 \log p$. We study the SDP method.

Here we have $x_i = \sqrt{\beta} u_i z + \epsilon_i$, where β is signal strength. We have a planted spike z as a k sparse unit vector.

The relaxed SDP version of the sparse PCA problem, at the same sparsity level as diagonal thresholding, can recover the support. Under a rank-one condition, SDP can reach the information limit of the sparse PCA problem. Thus SDP is optimal at rank 1 constraint since it reaches the information limit.

But, unfortunately, this paper proves that the SDP solution is not rank 1. Thus, at sparsity level $k = \Omega(\sqrt{n})$, SDP will not obtain a rank 1 solution. They propose a covariance thresholding method, which is consistent for $k = O(\sqrt{n})$ – the proof is not given, but it is proved in a different setting in Montanari's papers.

It turns out that $\|X - zz^T\| \geq 1/3$ with probability tending to 1 for $p \geq 150^4 n$, where X is solution of SDP. The constant 150^4 can be reduced. It's possible to bound $\langle \hat{\Sigma}, X \rangle$ both below $((1-\zeta) * (1+p/n)$ and above $(1+\zeta)(1+\sqrt{\beta}+\sqrt{p/n})^2)$: The ratio between the upper and lower bound tends to 1 as $p, n \to \infty$. Every solution of SDP method has this bound. However, every rank 1 solution of SDP method is bounded $\langle \hat{\Sigma}, Y \rangle \leq \frac{8}{9}\frac{p}{n}$. So SDP solution cannot be rank 1.

The covariance thresholding method outperforms diagonal thresholding empirically.