CS-GY 9223 D: Lecture 13
Compressed Sensing + Sparse Recovery

NYU Tandon School of Engineering, Prof. Christopher Musco

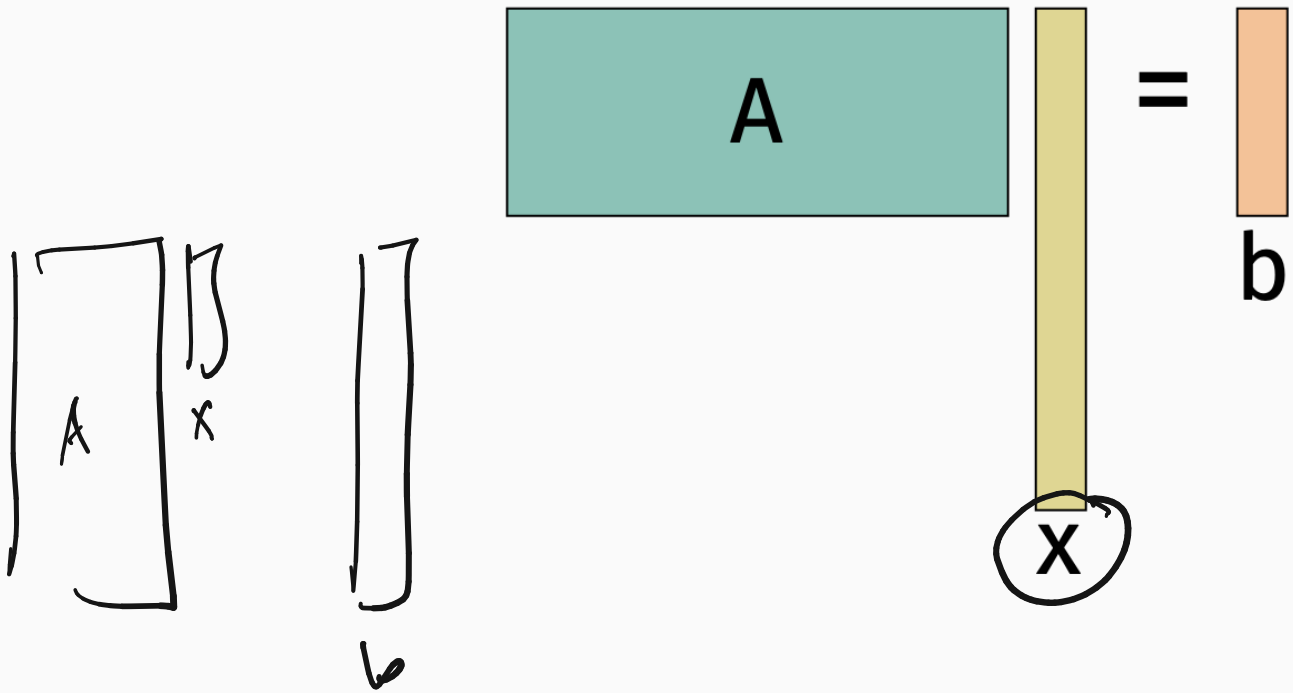
SPARSE RECOVERY/COMPRESSED SENSING

finding into low dimensions.

What do we know?

BASIC PROBLEM SETUP

Underdetermined linear regression: Given $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $m < n$, $\mathbf{b} \in \mathbb{R}^m$. Solve $\mathbf{Ax} = \mathbf{b}$ for $\mathbf{x}$.

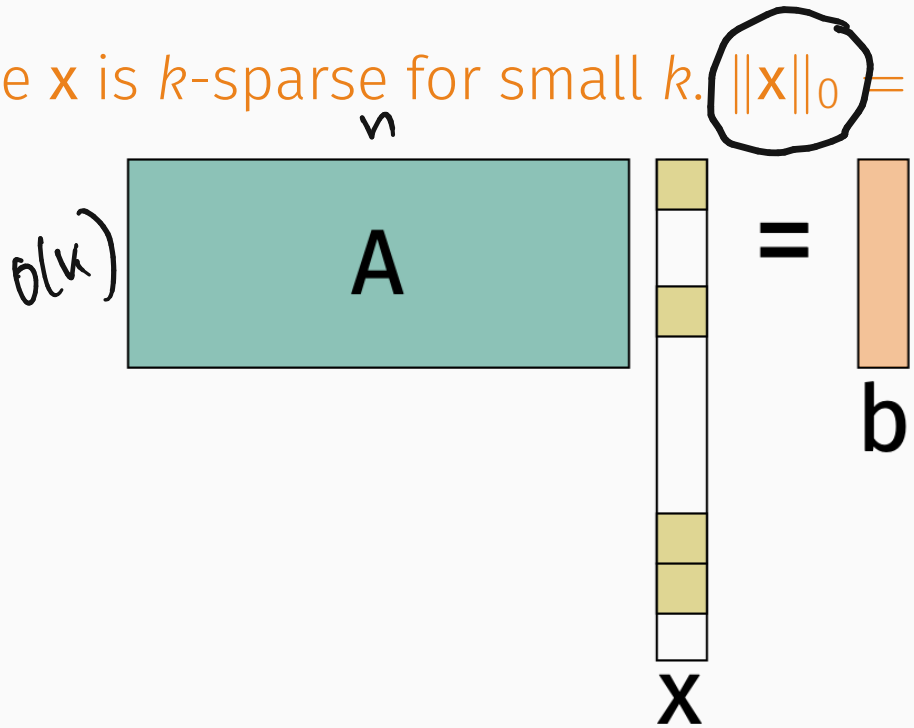


- Infinite possible solutions $\mathbf{x}$. In general, impossible to recover parameter vector.

SPARSITY RECOVERY/COMPRESSED SENSING

Underdetermined linear regression: Given $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $m < n$, $\mathbf{b} \in \mathbb{R}^m$. Solve $\mathbf{Ax} = \mathbf{b}$ for $\mathbf{x}$.

- Assume $\mathbf{x}$ is k-sparse for small k. $\|\mathbf{x}\|_0 = k$.



- In many cases can recover $\mathbf{x}$ with $m \ll n$ rows. In fact, often $m =\sim O(k)$ suffice.
- Need additional (strong) assumptions about $\mathbf{A}$!

QUICK ASIDE

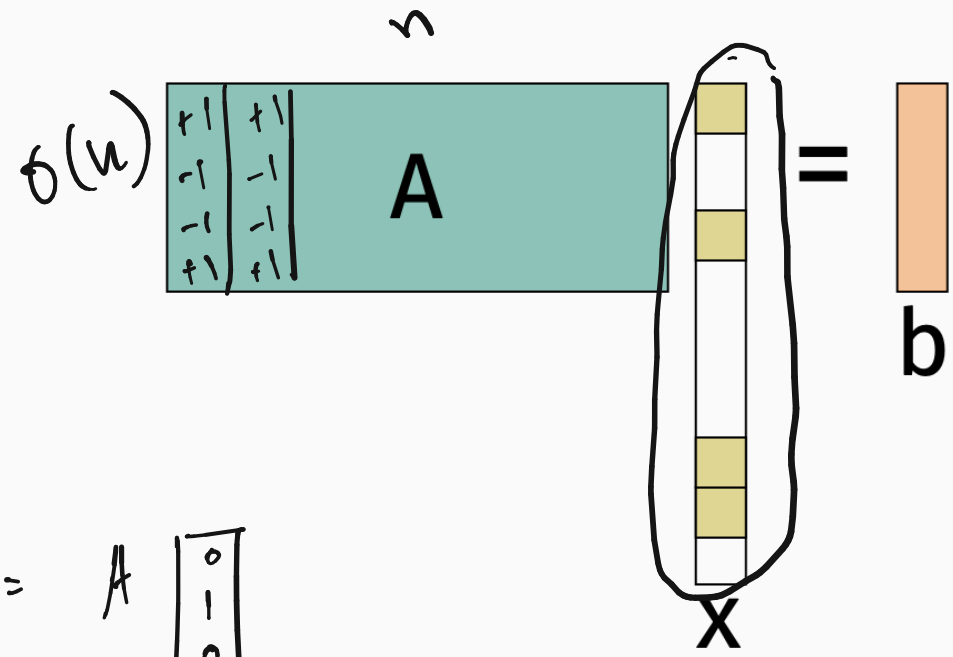
- In machine learning, we typically think about $\mathbf{A}$'s rows as data drawn from some universe/distribution:

	bedrooms	bathrooms	sq.ft.	floors	list price	sale price
home 1	2	2	1800	2	200,000	195,000
home 2	4	2.5	2700	1	300,000	310,000
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
home n	5	3.5	3600	3	450,000	450,000

- In many settings, we will get to choose $\mathbf{A}$'s rows $\mathbf{a}_1, \ldots, \mathbf{a}_m$. I.e. each $b_i = \mathbf{a}_i^T \mathbf{x}$ for some vector $\mathbf{a}_i$ that we select.
- In this setting, we often call b_i a linear measurement of $\mathbf{x}$ and we call $\mathbf{A}$ a measurement matrix.

ASSUMPTIONS ON MEASUREMENT MATRIX

When should this problem be difficult?



ASSUMPTIONS ON MEASUREMENT MATRIX

$Ax = Ay \rightarrow A(x-0) = 0$

$x-0$ is $2k$ sparse

Many ways to formalize our intuition

- $\mathbf{A}$ has Kruskal rank r. All sets of r columns in $\mathbf{A}$ are linearly independent.
  - Recover vectors $\mathbf{x}$ with sparsity $k = r/2$.
- ($\mathbf{A}$ is μ-incoherent.) $|\mathbf{A}_i^T \mathbf{A}_j| \leq \mu \|\mathbf{A}_i\|_2 \|\mathbf{A}_j\|_2$ for all columns $\mathbf{A}_i, \mathbf{A}_j$, $i \neq j$.
  - Recover vectors $\mathbf{x}$ with sparsity $k = 1/\mu$.
- **Focus today:** ($\mathbf{A}$ obeys the Restricted Isometry Property.)

RESTRICTED ISOMETRY PROPERTY

Definition ((q, ϵ)-Restricted Isometry Property)

A matrix $\mathbf{A}$ satisfies (q, ϵ)-RIP if, for all $\mathbf{x}$ with $\|\mathbf{x}\|_0 \leq q$,

$$(1 - \epsilon)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \epsilon)\|\mathbf{x}\|_2^2.$$

- Johnson-Lindenstrauss type condition.
- $\mathbf{A}$ preserves the norm of all q sparse vectors, instead of the norms of a fixed discrete set of vectors, or all vectors in a subspace (as in subspace embeddings).

FIRST SPARSE RECOVERY RESULT

Theorem (ℓ_0-minimization)

Suppose we are given $\mathbf{A} \in \mathbb{R}^{m \times n}$ *and* $\mathbf{b} = \mathbf{A}\mathbf{x}$ *for an unknown* k*-sparse* $\mathbf{x} \in \mathbb{R}^n$*. If* $\mathbf{A}$ *is* $(2k, \epsilon)$*-RIP for any* $\epsilon < 1$ *then* $\mathbf{x}$ *is the unique minimizer of:*

$$\min \|\mathbf{z}\|_0 \quad \text{subject to} \quad \mathbf{A}\mathbf{z} = \mathbf{b}.$$

- Establishes that information theoretically we can recover $\mathbf{x}$. Solving the ℓ_0-minimization problem is computationally difficult, requiring $O(n^k)$ time. We will address faster recovery shortly.

FIRST SPARSE RECOVERY RESULT

Claim: If $\mathbf{A}$ is $(2k, \epsilon)$-RIP for any $\epsilon < 1$ then $\mathbf{x}$ is the unique minimizer of $\min_{\mathbf{Az}=\mathbf{b}} \|\mathbf{z}\|_0$.

Proof: By contradiction, assume there is some $\mathbf{y} \neq \mathbf{x}$ such that $\mathbf{Ay} = \mathbf{b}$, $\|\mathbf{y}\|_0 \leq \|\mathbf{x}\|_0$.

$Ax = b$

$A(y - x) = 0$

$\leq 2k$ sparse

$\|A(y-x)\|_2 \in (1 \pm \epsilon)\|y - x\|_2$

must be non-zero

non-zero

$\|A(y-x)\|_2 = 0$

$2k$ sparse

$\|y - x\| \neq 0$

Important note: Robust versions of this theorem and the others we will discuss exist. These are much more important practically. Here's a flavor of a robust result:

- Suppose $\mathbf{b} = \mathbf{A}(\mathbf{x} + \mathbf{e})$ where $\mathbf{x}$ is k-sparse and $\mathbf{e}$ is dense but has bounded norm.
- Recover some k-sparse $\tilde{\mathbf{x}}$ such that:

$$\|\tilde{\mathbf{x}} - \mathbf{x}\|_2 \leq \|\mathbf{e}\|_1$$

or even

$$\|\tilde{\mathbf{x}} - \mathbf{x}\|_2 \leq O\left(\frac{1}{\sqrt{k}}\right) \|\mathbf{e}\|_1.$$

ROBUSTNESS

We will not discuss robustness in detail, but along with computational considerations, it is a big part of what has made compressed sensing such an active research area in the last 20 years. Non-robust compressed sensing results have been known for a long time:

Gaspard Riche de Prony, *Essay experimental et analytique: sur les lois de la dilatabilite de fluides elastique et sur celles de la force expansive de la vapeur de l'alcool, a differentes temperatures.* Journal de l'Ecole Polytechnique, 24–76. **1795**.

What matrices satisfy this property?

- Random Johnson-Lindenstrauss matrices (Gaussian, sign, etc.) with $m = O(\frac{k \log(n/k)}{\epsilon^2})$ rows are (k, ϵ)-RIP.

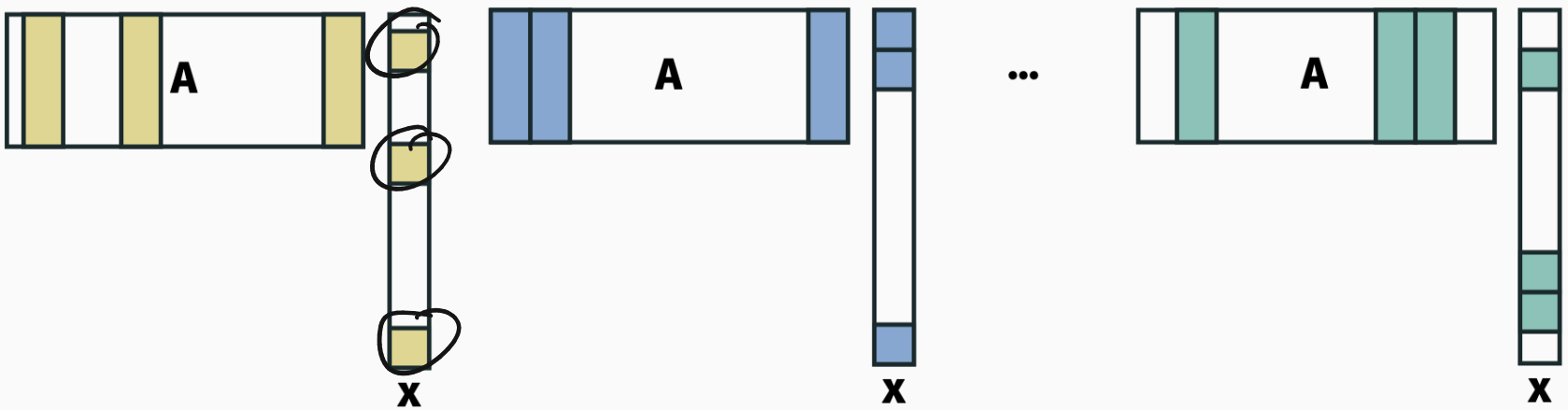
Some real world data may look random, but this is also a useful observation algorithmically when we want to design $\mathbf{A}$.

Definition ((q, ϵ)-Restricted Isometry Property – Candes, Tao '05)

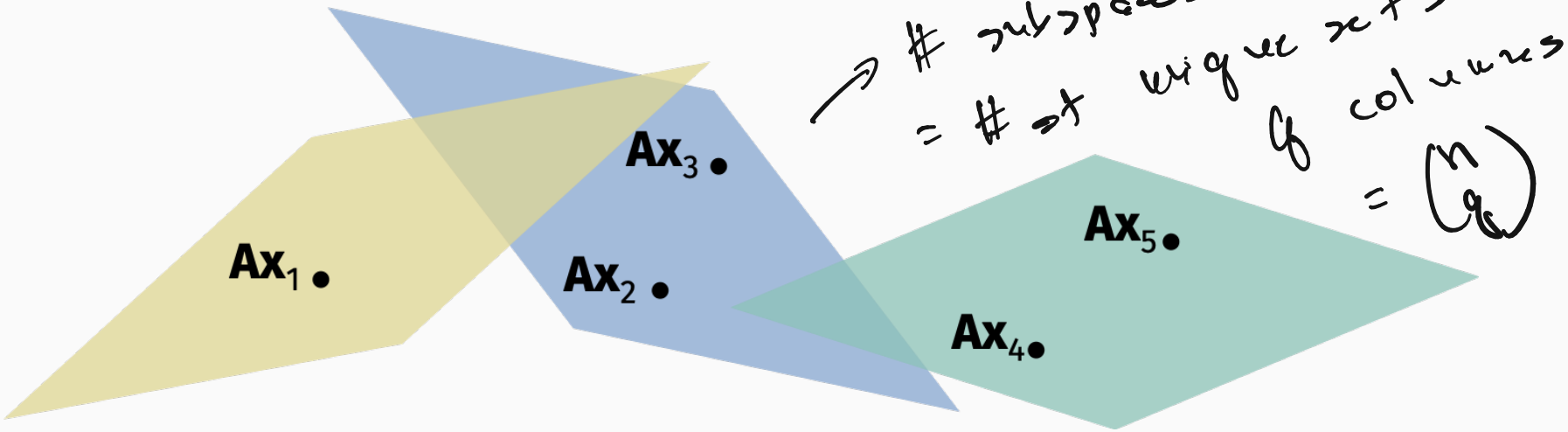
A matrix $\mathbf{A}$ satisfies (q, ϵ)-RIP if, for all $\mathbf{x}$ with $\|\mathbf{x}\|_0 \leq q$,

$$(1 - \epsilon)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \epsilon)\|\mathbf{x}\|_2^2.$$

The vectors that can be written as $\mathbf{A}\mathbf{x}$ for q sparse $\mathbf{x}$ lie in a union of q dimensional linear subspaces:



Candes, Tao 2005: A random JL matrix with $O(q \log(n/q)/\epsilon^2)$ rows satisfies (q, ϵ)-RIP.



Any ideas for how you might prove this? I.e. prove that a random matrix preserves the norm of every $\mathbf{x}$ in this union of subspaces?

RESTRICTED ISOMETRY PROPERTY FROM JL

Theorem (Subspace Embedding from JL)

Let $\mathcal{U} \subset \mathbb{R}^n$ be a q-dimensional linear subspace in $\mathbb{R}^n$. If $A \in \mathbb{R}^{m \times n}$ is chosen from any distribution $\mathcal{D}$ satisfying the Distributional JL Lemma, then with probability $1 - \delta$,

$$(1-\epsilon)\|\mathbf{v}\|_2^2 \leq \|A\mathbf{v}\|_2^2 \leq (1+\epsilon)\|\mathbf{v}\|_2^2$$

for all $\mathbf{v} \in \mathcal{U}$, as long as $m = O\left(\frac{q+\log(1/\delta)}{\epsilon^2}\right)$.

$\log\left(\frac{n \cdot n-1 \cdots n-q}{q \cdot q-1 \cdots 1}\right)$ $\delta = \frac{1}{\binom{n}{q}}$

Quick argument:

$\log\binom{n}{q} \leq \log(n^q) = q\log(n)$

$O\left(\frac{q + \log\binom{n}{q}}{\epsilon^2}\right) = O\left(\frac{q + q\log(n/q)}{\epsilon^2}\right)$

$O\left(\frac{q\log(n/q)}{\epsilon^2}\right)$

APPLICATION: HEAVY HITTERS IN DATA STREAMS

Suppose you view a stream of numbers in $1, \ldots, n$:

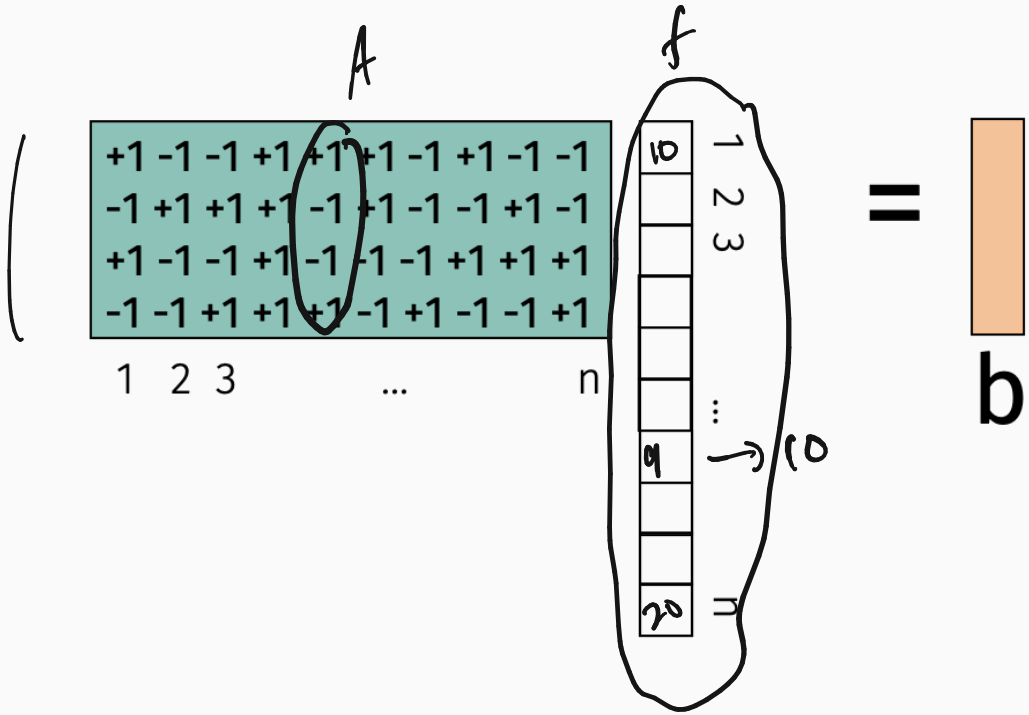
$$4, 18, 4, 1, 2, 24, 6, 4, 3, 18, 18, \ldots$$

After some time, you want to report which k items appeared most frequently in the stream.

E.g. Amazon is monitoring web-logs to see which product pages people view. They want to figure out which products are viewed most frequently. $n \approx 500$ million.

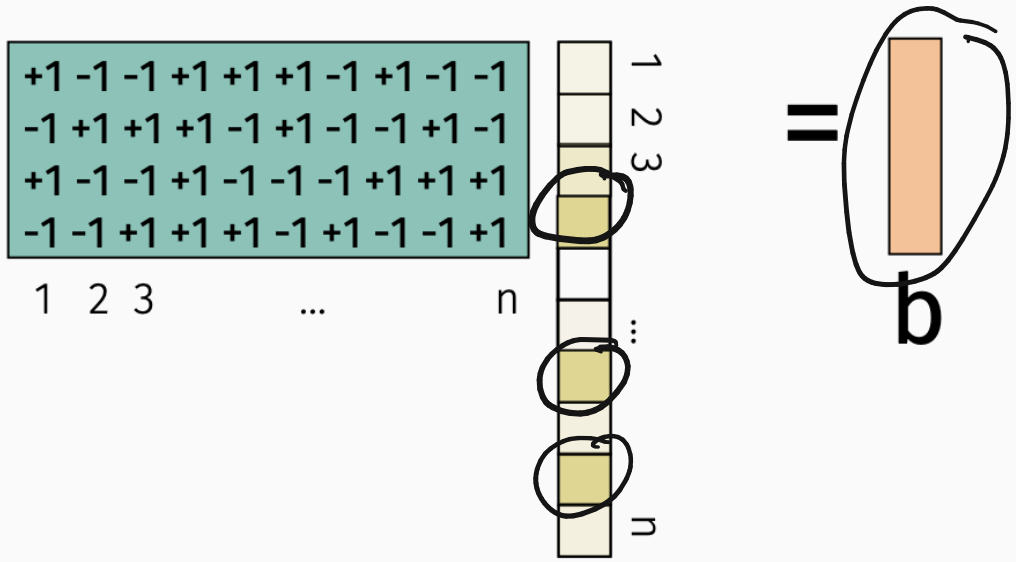
How can you do this quickly in small space?

APPLICATION: HEAVY HITTERS IN DATA STREAMS



- Every time we receive a number i in the stream, add column $\mathbf{A}_i$ to $\mathbf{b}$.

APPLICATION: HEAVY HITTERS IN DATA STREAMS



- At the end $\mathbf{b} = \mathbf{A}\mathbf{x}$ for an approximately sparse $\mathbf{x}$ if there were only a few "heavy hitters". Recover $\mathbf{x}$ from $\mathbf{b}$ using a sparse recovery method (like ℓ_0 minimization).

APPLICATION: HEAVY HITTERS IN DATA STREAMS

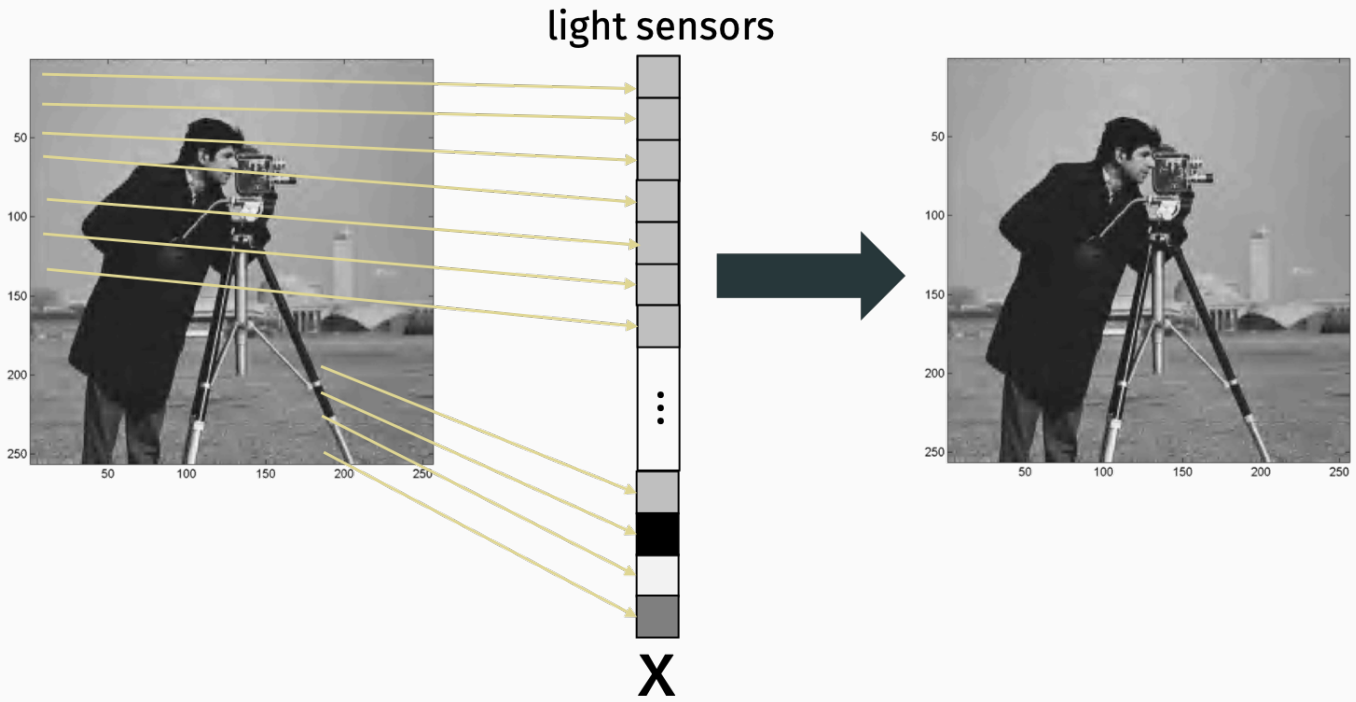
Naturally handles both insertions or deletions.

$\mathit{insert}(4), \mathit{insert}(18), \mathit{remove}(4), \mathit{insert}(1), \mathit{insert}(2), \mathit{remove}(2) \ldots$

E.g. Amazon is monitoring what products people add to their "wishlist" and wants a list of most tagged products. Wishlists can be changed over time, including by removing items.

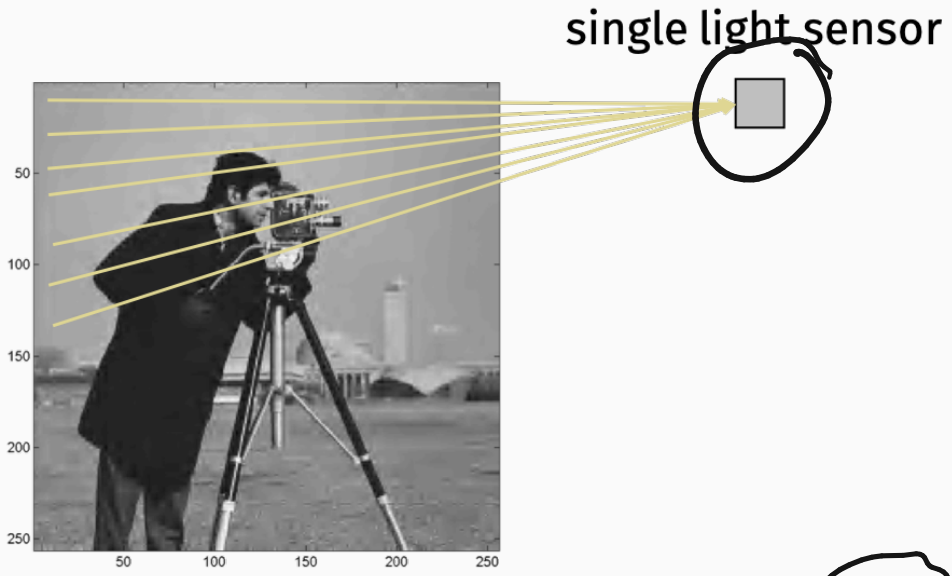
APPLICATION: SINGLE PIXEL CAMERA

Typical acquisition of image by camera:



Requires one image sensor per pixel captured.

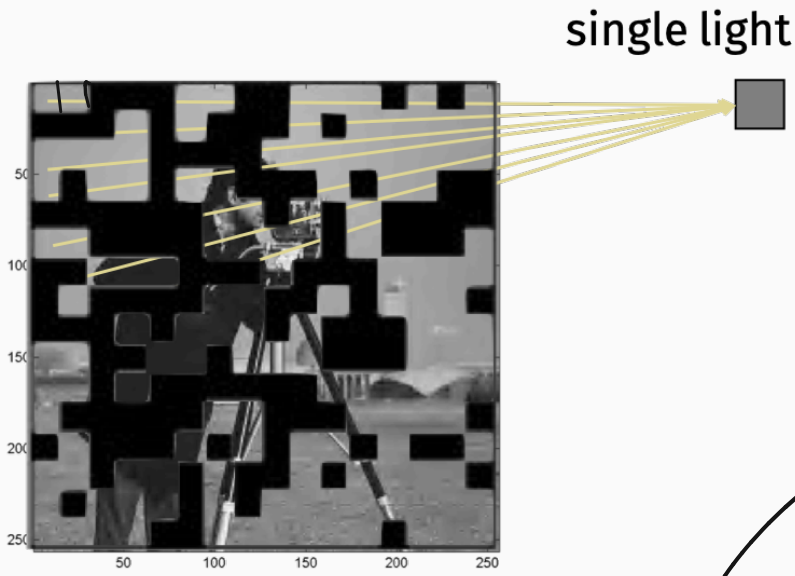
Compressed acquisition of image:



$$p = \sum_{i=1} x_i = \begin{bmatrix} 1/4 & 1/4 & \cdots & 1/4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Does not provide very much information about the image.

But several random linear measurements do!



$$p = \sum_{i=1} R_i x_i = \begin{bmatrix} 0 & 1 & 0 & 0 \dots & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Application: Single Pixel Camera

Applications in:

- Imaging outside of the visible spectrum (more expensive sensors).
- Microscopy.
- Other scientific imaging.

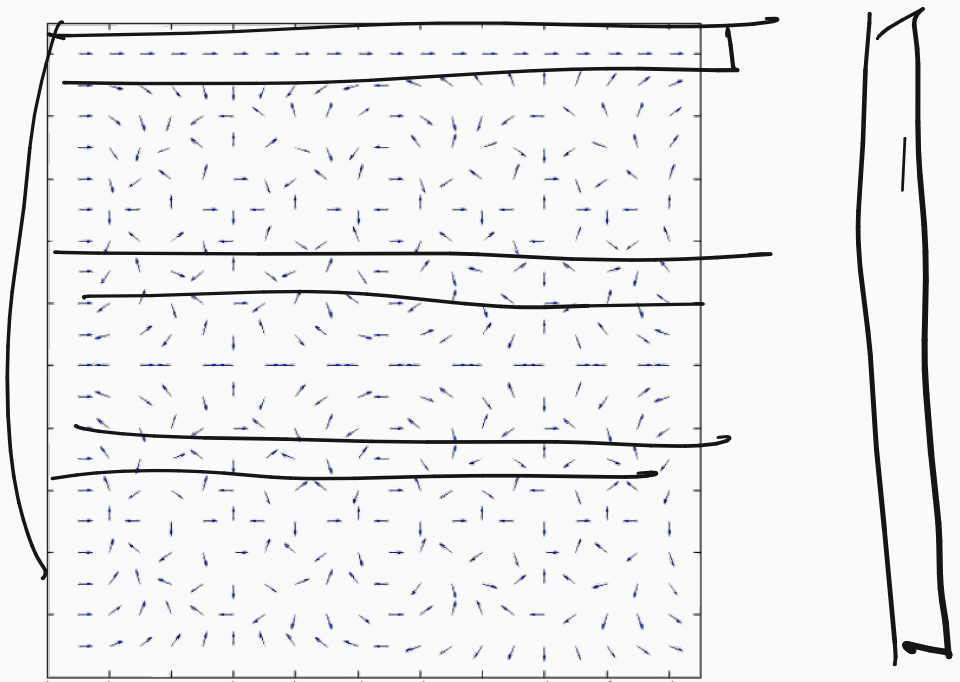
Compressed sensing theory does not exactly describe these problems, but has been very valuable in modeling them.

THE DISCRETE FOURIER MATRIX

The $n \times n$ discrete Fourier matrix $\mathbf{F}$ is defined:

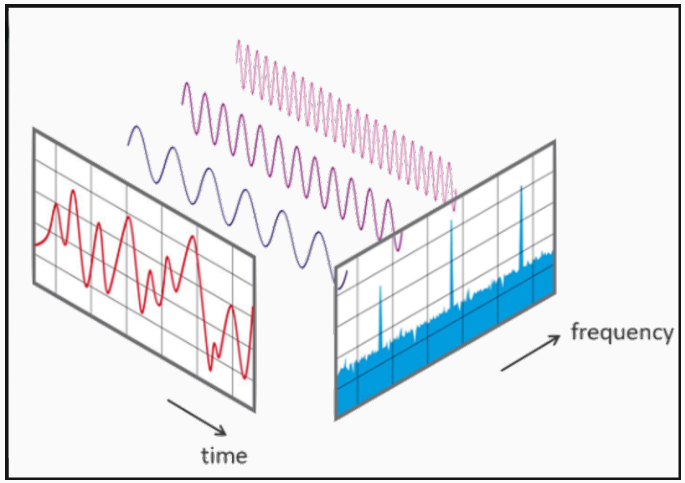
$$F_{j,k} = e^{\frac{-2\pi i}{n} jk},$$

where $i = \sqrt{-1}$. Recall $e^{\frac{-2\pi i}{n} j \cdot k} = cos(2\pi jk/n) - i \sin(2\pi jk/n)$.



THE DISCRETE FOURIER MATRIX

$\mathbf{Fx}$ is the Discrete Fourier Transform of the vector $\mathbf{x}$ (what an FFT computes).



Decomposes $\mathbf{x}$ into different frequencies: $[\mathbf{Fx}]_j$ is the component with frequency j/n.

Because $\mathbf{F}^*\mathbf{F} = \mathbf{I}$, $\mathbf{F}^*\mathbf{Fx} = \mathbf{x}$, so we can recover $\mathbf{x}$ if we have access to its DFT. $\mathbf{Fx}$.

RESTRICTED ISOMETRY PROPERTY

Setting $\mathbf{A}$ to contain a random $m \sim O\left(\frac{k \log^2 k \log n}{\epsilon^2}\right)$ rows of the discrete Fourier matrix $\mathbf{F}$ yields a matrix that with high probability satisfies (k, ϵ)-RIP. [Haviv, Regev, 2016].

Improves on a long line of work: Candés, Tao, Rudelson, Vershynin, Cheraghchi, Guruswami, Velingker, Bourgain.

You have seen some of the tools used prove this when we proved that a subsampled Hadamard matrix, which is a type of Fourier matrix, can be used to give a *JL* guarantee.

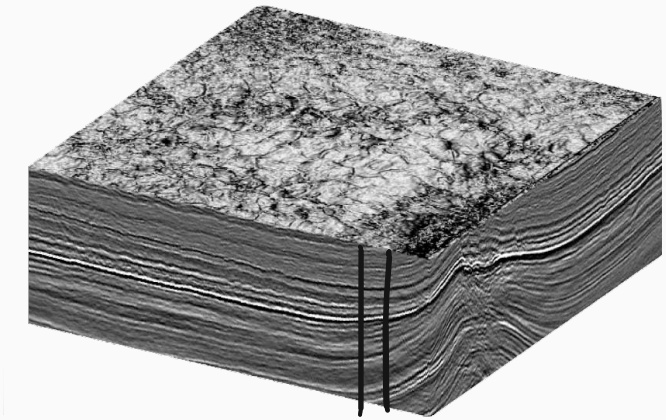
If $\mathbf{A}$ is a subset of q rows from $\mathbf{F}$, then $\mathbf{Ax}$ is a subset of random frequency components from $\mathbf{x}$'s discrete Fourier transform.

In many scientific applications, we can collect entries of $\mathbf{Fx}$ one at a time for some unobserved data vector $\mathbf{x}$.

APPLICATION: GEOPHYSICS

Warning: very cartoonish explanation of very complex problem.

Understanding what material is beneath the crust:



Think of vector $\mathbf{x}$ as scalar values of the density/reflectivity in a single vertical core of the earth.

How do we measure entries of Fourier transform $\mathbf{Fx}$?

APPLICATION: GEOPHYSICS

Vibrate the earth at different frequencies! And measure the response.

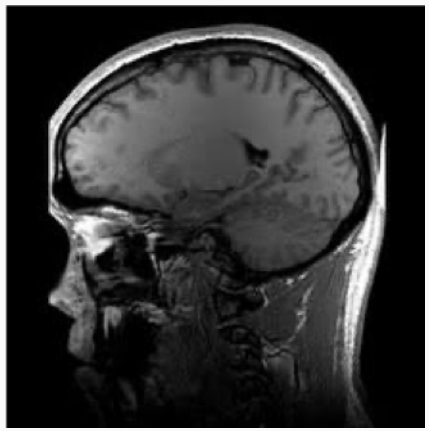
Vibroseis Truck

Can also use airguns, controlled explorations, vibrations from drilling, etc. The fewer measurements we need from **Fx**, the cheaper and faster our data acquisition process becomes.

Killer app: Oil Exploration.

Warning: very cartoonish explanation of very complex problem.

Medical Imaging (MRI)



Vector $\mathbf{x}$ here is a 2D image. Everything works with 2D Fourier transforms.

How do we measure entries of Fourier transform $\mathbf{Fx}$?

APPLICATION: GEOPHYSICS

Blast the body with sounds waves of varying frequency.

The fewer measurements we need from $\mathbf{Fx}$, the faster we can acquire an image.

- Especially important when trying to capture something moving (e.g. lungs, baby, child who can't sit still).
- Can also cut down on power requirements (which for MRI machines are huge).

Definition ((q, ϵ)-Restricted Isometry Property)

A matrix $\mathbf{A}$ satisfies (q, ϵ)-RIP if, for all $\mathbf{x}$ with $\|\mathbf{x}\|_0 \leq q$,

$$(1 - \epsilon)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \epsilon)\|\mathbf{x}\|_2^2.$$

Lots of other random matrices satisfy RIP as well.

One major theoretical question is if we can deterministically construct good RIP matrices. Interestingly, if we want $(O(k), O(1))$ RIP, we can only do so with $O(k^2)$ rows (now very slightly better – thanks to Bourgain et al.).

Whether or not a linear dependence on k is possible with a deterministic construction is unknown.

Theorem (ℓ_0-minimization)

Suppose we are given $\mathbf{A} \in \mathbb{R}^{m \times n}$ *and* $\mathbf{b} = \mathbf{A}\mathbf{x}$ *for an unknown* k*-sparse* $\mathbf{x}$*. If* $\mathbf{A}$ *is* $(2k, \epsilon)$*-RIP for any* $\epsilon < 1$ *then* $\mathbf{x}$ *is the unique minimizer of:*

$$\min \|\mathbf{z}\|_0 \qquad \textit{subject to} \qquad \mathbf{A}\mathbf{z} = \mathbf{b}.$$

Algorithm question: Can we recover $\mathbf{x}$ using a faster method? Ideally in polynomial time.

Convex relaxation of the ℓ_0 minimization problem:

Problem (Basis Pursuit, i.e. ℓ_1 minimization.)

$$\min_{z} \|z\|_1 \qquad \textit{subject to} \qquad Az = b.$$

- Objective is convex.
- Optimizing over convex set.

What is one method we know for solving this problem?

Equivalent formulation:

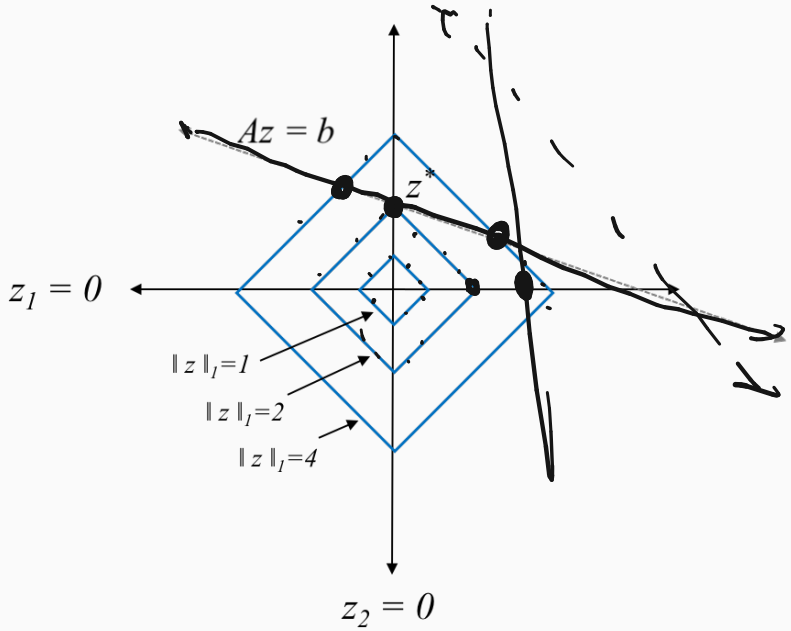
Problem (Basis Pursuit Linear Program.)

$$\min_{\mathbf{w},\mathbf{z}} \mathbf{1}^T\mathbf{w} \qquad \textit{subject to} \qquad \left(\mathbf{A}\mathbf{z} = \mathbf{b}, \mathbf{w} \geq 0, -\mathbf{w} \leq \mathbf{z} \leq \mathbf{w}.\right)$$

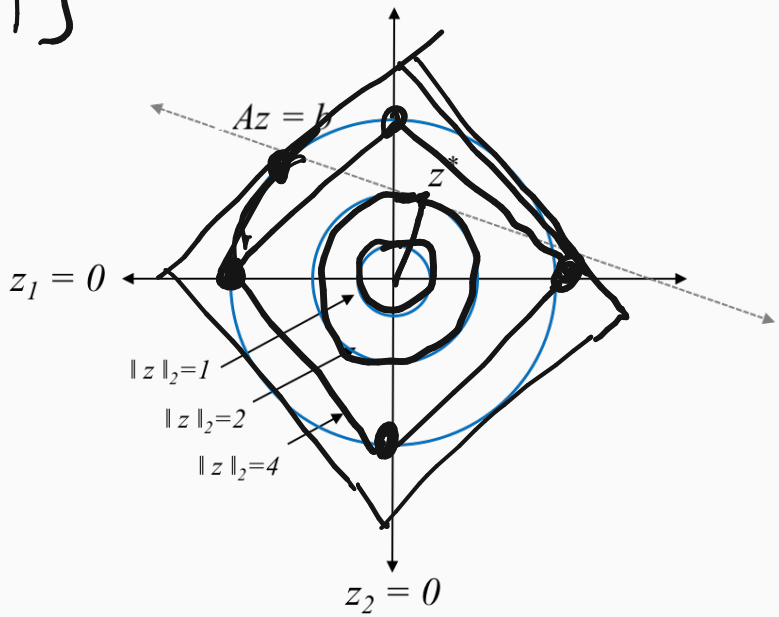
Can be solved using any algorithm for linear programming. An Interior Point Method will run in at ~~worst~~ $\sim O(n^{3.5})$ time.

BASIS PURSUIT INTUITION

Suppose **A** is 2×1, so **b** is just a scalar and **x** is a 2-dimensional vector.



Vertices of level sets of ℓ_1 norm correspond to sparse solutions.



This is not the case e.g. for the ℓ_2 norm.

Theorem

If $\mathbf{A}$ *is* $(3k, \epsilon)$*-RIP for* $\epsilon < .17$ *and* $\|\mathbf{x}\|_0 = k$*, then* $z^\star = \mathbf{x}$ *is the unique optimal solution of the Basis Pursuit LP).*

Similar proof to ℓ_0 minimization:

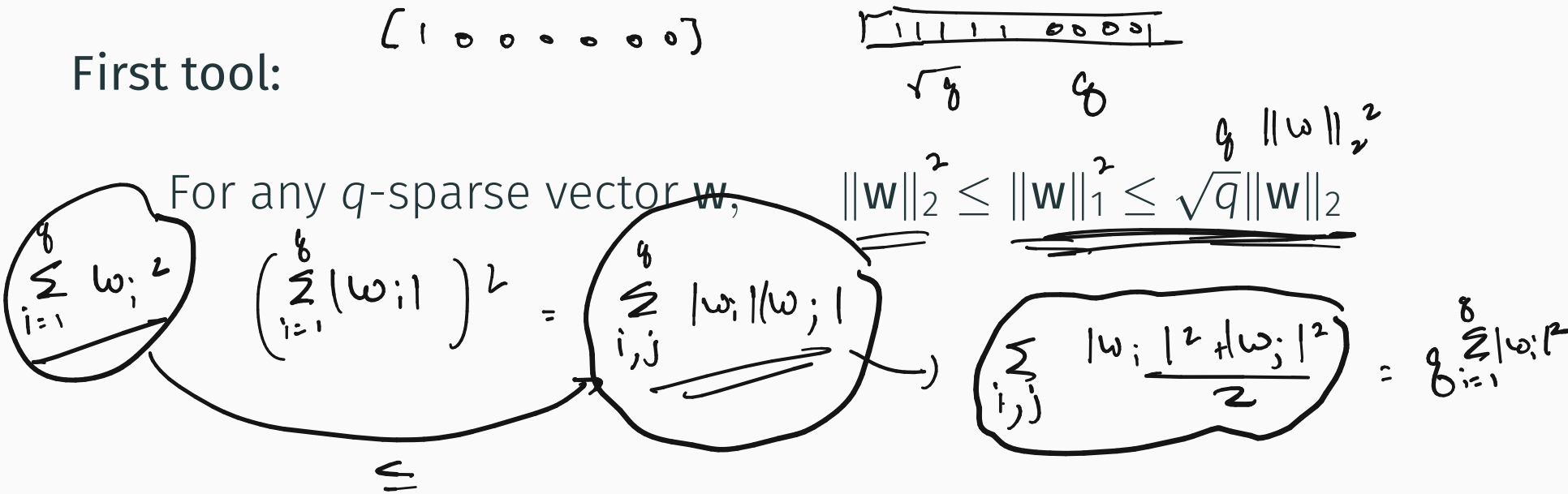
- By way of contradiction, assume $\mathbf{x}$ is not the optimal solution. Then there exists some non-zero Δ such that:
  - $\|\mathbf{x} + \Delta\|_1 \leq \|\mathbf{x}\|_1$
  - $\mathbf{A}(\mathbf{x} + \Delta) = \mathbf{A}\mathbf{x}$. I.e. $\mathbf{A}\Delta = 0$.

Difference is that we can no longer assume that Δ is sparse.

We will argue that Δ is approximately sparse.

First tool:

For any q-sparse vector $\mathbf{w}$, $\|\mathbf{w}\|_2^2 \le \|\mathbf{w}\|_1^2 \le \sqrt{q}\|\mathbf{w}\|_2$



Second tool:

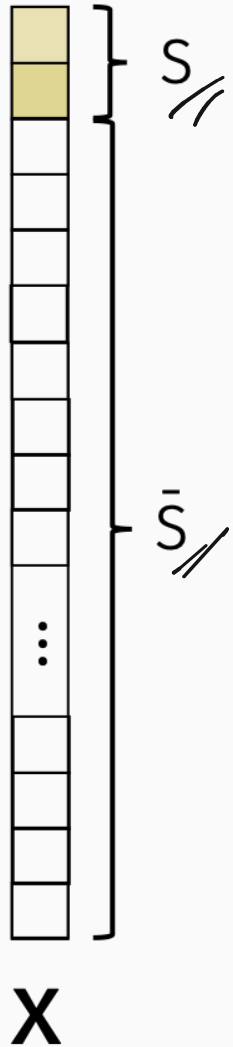
$a^2 + b^2 \ge 2ab$ $\quad (a-b)^2 \ge 0$ $\quad a^2 + b^2 - 2ab \ge 0$

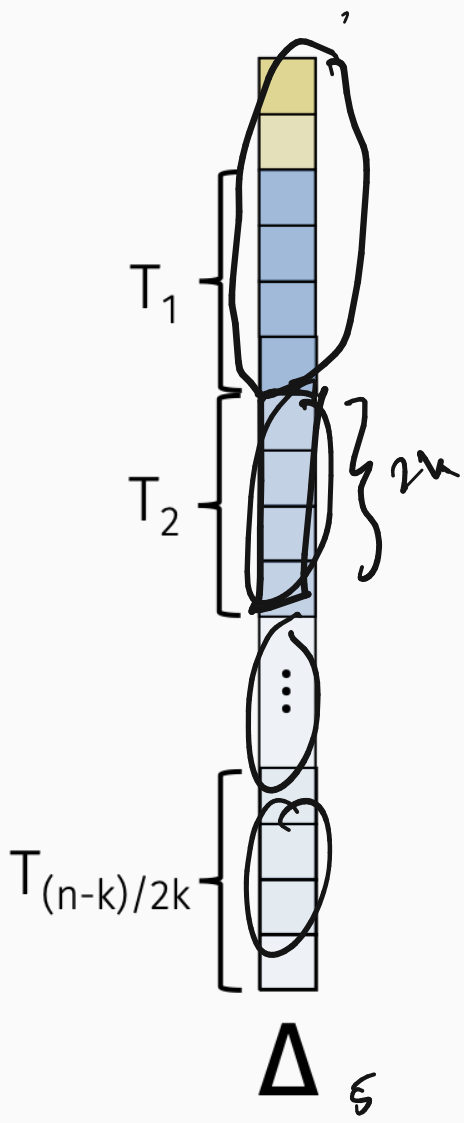
For any norm and vectors $\mathbf{a}, \mathbf{b}$, $\|\mathbf{a}+\mathbf{b}\| \ge \|\mathbf{a}\| - \|\mathbf{b}\|$

$\|a+b-b\| \le \|a+b\| + \|b\|$

$\|a\| \le \|a+b\| + \|b\|$

Some definitions:





Claim 1: $\|\Delta_S\|_1 \geq \|\Delta_{\bar{S}}\|_1$

$$\|x + \Delta\|_1 \leq \|x\|_1$$

$$\|x + \Delta_S\|_1 + \|\Delta_{\bar{S}}\|_1 \leq \|x\|_1$$

$$\left(\|x\|_1 - \|\Delta_S\|_1\right) + \|\Delta_{\bar{S}}\|_1 \leq \|x\|_1$$

$$\|\Delta_{\bar{S}}\|_1 - \|\Delta_S\|_1 \leq 0$$

Claim 2: $\|\Delta_S\|_2 \geq \sqrt{2} \sum_{j \geq 2} \|T_j\|_2$:

$$\|\Delta_S\|_2 \geq \frac{1}{\sqrt{k}} \|\Delta_S\|_1 \geq \frac{1}{\sqrt{k}} \|\Delta_{\bar{0}}\|_1 = \frac{1}{\sqrt{k}} \sum_{j \geq 1} \|\Delta_{T_j}\|_1$$

$$\|\Delta_{T_j}\|_1 \geq \sqrt{2k} \, \|\Delta_{T_{j+1}}\|_2 \geq \text{be} \max(|\Delta_{T_{j+1}}|)$$

$2k \cdot 2$ $\qquad \sqrt{2k \cdot 2^2} = \sqrt{2k} \cdot 2$

$$\|\Delta_S\|_2 \geq \frac{1}{\sqrt{k}} \sum_{j \geq 1} \sqrt{2k} \, \|\Delta_{T_{j+1}}\|_2 \geq \sqrt{2} \sum_{j \geq 2} \|T_j\|_2$$

Finish up proof by contradiction:

$$0 = \|A\Delta\|_2 \geq \|A\Delta_{S\cup T_1}\|_2 - \sum_{j\geq 2} \|A\Delta_{T_j}\|_2$$

$$\geq (1-\varepsilon)\|\Delta_{S\cup T_1}\|_2 - \sum_{j\geq 2}(1+\varepsilon)\|A\Delta_{T_j}\|_2$$

$$\geq (1-\varepsilon)\|\Delta_S\|_2 - (1+\varepsilon)\frac{1}{\sqrt{2}}\|\Delta_S\|_2$$

$$= \left((1-\varepsilon) - (1+\varepsilon)\frac{1}{\sqrt{2}}\right)\|\Delta_S\|_2$$

$$\varepsilon \approx .17$$

FASTER METHODS

A lot of interest in developing even faster algorithms that avoid using the "heavy hammer" of linear programming and run in even faster than $O(n^{3.5})$ time.

- **Iterative Hard Thresholding**: Looks a lot like projected gradient descent. Solve $\min_{\mathbf{z}} \|\mathbf{Az} - \mathbf{b}\|$ with gradient descent while continually projecting z back to the set of k-sparse vectors. Runs in time $\sim O(nk \log n)$ for Gaussian measurement matrices and $O(n \log n)$ for subsampled Fourer matrices.
- Other "first order" type methods: Orthogonal Matching Pursuit, CoSaMP, Subspace Pursuit, etc.

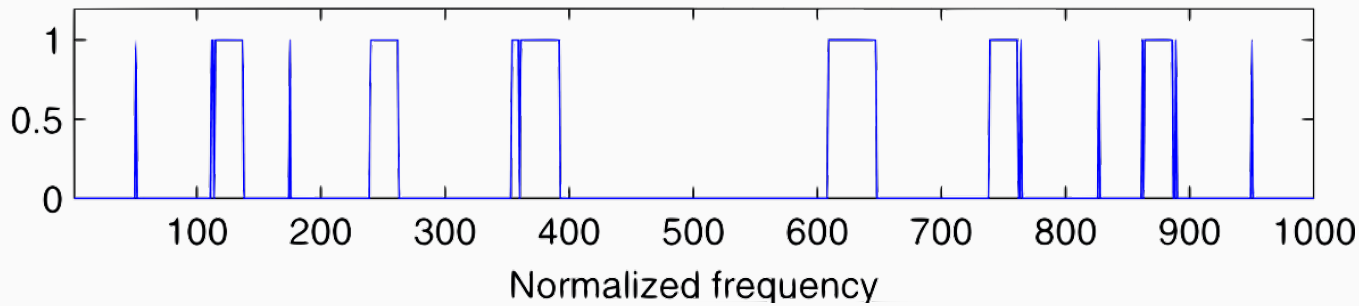
When $\mathbf{A}$ is a subsampled Fourier matrix, there are now methods that run in $O(k \log^c n)$ time [Hassanieh, Indyk, Kapralov, Katabi, Price, Shi, etc. 2012+].

Hold up...

Corollary: When $\mathbf{x}$ is k-sparse, we can compute the inverse Fourier transform $\mathbf{F}^*\mathbf{F}\mathbf{x}$ of $\mathbf{F}\mathbf{x}$ in $O(k \log^c n)$ time!

- Randomly subsample $\mathbf{F}\mathbf{x}$.
- Feed that input into our sparse recovery algorithm to extract $\mathbf{x}$.

Fourier and inverse Fourier transforms in sublinear time when the output is sparse.



Applications in: Wireless communications, GPS, protein imaging, radio astronomy, etc. etc.