Sparse Recovery Using Sparse Matrices

Piotr Indyk

MIT

Compressed Sensing: Recap

- Want to acquire a signal $x=[x_1 \ldots x_n]$
- Acquisition proceeds by computing Ax (+noise) of dimension $m \ll n$
- From Ax we want to recover an approximation x^* of x
  - Note: x^* does not have to be k-sparse in general
- Method: solve the following program:

$$\text{minimize } \|x^*\|_1$$
$$\text{subject to } Ax^*=Ax$$

- Guarantee (L1/L1): for some $C>1$

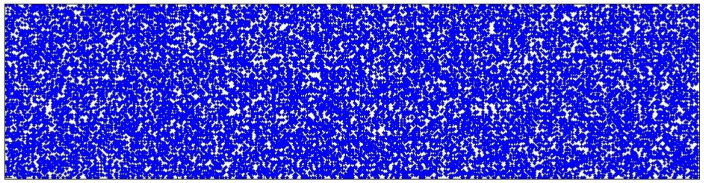
$$\|x-x^*\|_1 \le C \min_{k\text{-sparse } x''} \|x-x''\|_1$$

as long as A satisfies (ck, δ)-RIP property:

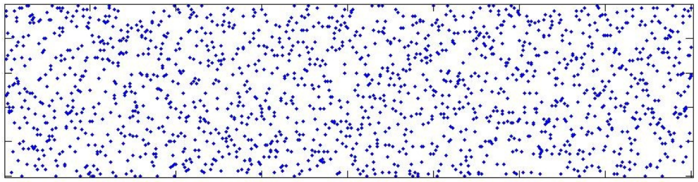
$$(1-\delta) \|x\|_2 \le \|Ax\|_2 \le (1+\delta) \|x\|_2$$

Choices for matrix A

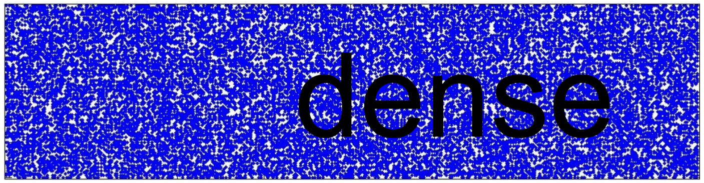
- Dense matrices:
  - Compressed sensing



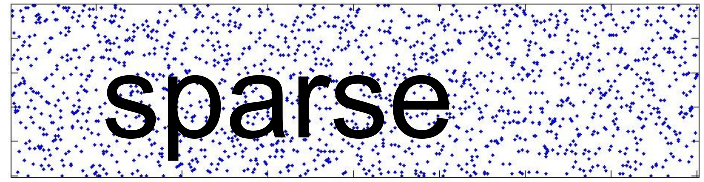
- Sparse matrices:
  - Data stream algorithms



- "Traditional" tradeoffs:
  - Sparse: computationally more efficient ($n \log n$ vs $poly(n)$)
  - Dense: shorter sketches ($k \log (n/k)$ vs $k \log n$)
  - …
- The "best of both worlds" ?
  - Today and next week



vs.



- Restricted Isometry Property (RIP) [Candes-Tao'04]

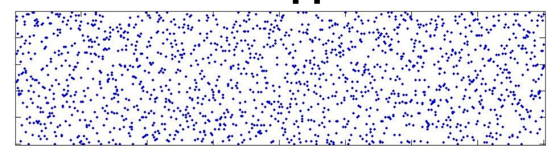
$$\Delta \text{ is } k\text{-sparse} \Rightarrow ||\Delta||_2 \leq ||A\Delta||_2 \leq C\, ||\Delta||_2$$

- Holds w.h.p. for random Gaussian/Bernoulli: $m=O(k \log (n/k))$
- Consider m x n 0-1 matrices with d ones per column
- Do they satisfy RIP ?
  - No, unless $m=\Omega(k^2)$ [Chandar'07, Nelson-Nguyen'13]
- However, they can satisfy the following RIP-1 property [Berinde-Gilbert-Indyk-Karloff-Strauss'08]:

$$\Delta \text{ is } k\text{-sparse} \Rightarrow d\,(1-\varepsilon)\, ||\Delta||_1 \leq ||A\Delta||_1 \leq d||\Delta||_1$$

- Sufficient (and necessary) condition: the underlying graph is a $(\, k,\, d(1-\varepsilon/2)\,)$-expander

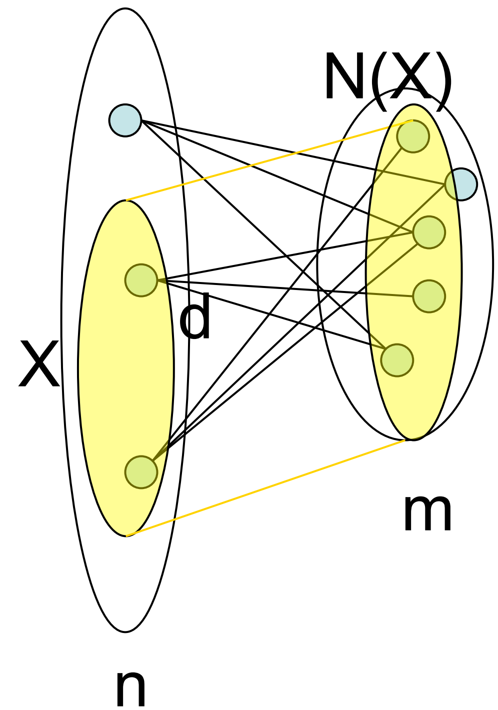
Expanders



- A bipartite graph is an $(l, d(1-\varepsilon))$-expander if for any left set X, $|X| \leq l$, we have $|N(X)| \geq (1-\varepsilon) d\, |X|$
- Objects well-studied in theoretical computer science and coding theory
- Constructions:
  - Probabilistic: $m = O(l \log (n/l))$
  - Explicit: $m = l$ quasipolylog n
- High expansion implies RIP-1:

  Δ is k-sparse $\Rightarrow d(1-2\varepsilon)\, \|\Delta\|_1 \leq \|A\Delta\|_1 \leq d\|\Delta\|_1$

  [Berinde-Gilbert-Indyk-Karloff-Strauss'08]



Proof: $d(1-\varepsilon/2)$-expansion $\Rightarrow$ RIP-1

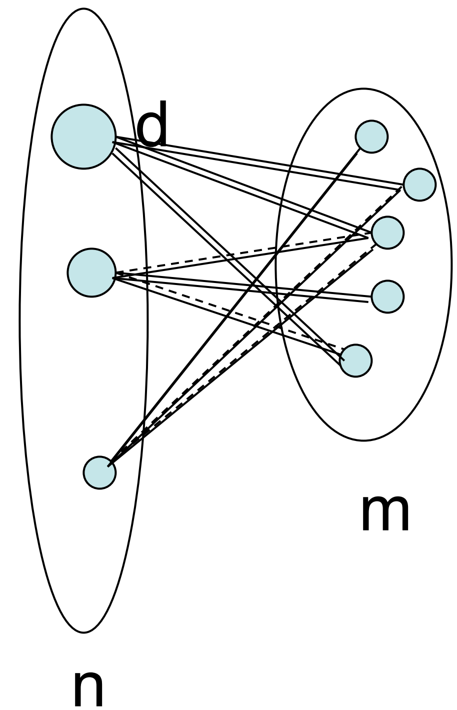
- Want to show that for any k-sparse Δ we have

  $$d\,(1-\varepsilon)\,\|\Delta\|_1 \le \|A\,\Delta\|_1 \le d\|\Delta\|_1$$

- RHS inequality holds for ***any*** Δ
- LHS inequality:
  - W.l.o.g. assume

    $$|\Delta_1| \ge \ldots \ge |\Delta_k| \ge |\Delta_{k+1}| = \ldots = |\Delta_n| = 0$$

  - Consider the edges $e=(i,j)$ in a lexicographic order
  - For each edge $e=(i,j)$ define $r(e)$ s.t.
    - $r(e)=-1$ if there exists an edge $(i',j)<(i,j)$
    - $r(e)=1$ if there is no such edge
- Claim 1: $\|A\Delta\|_1 \ge \sum_{e=(i,j)} |\Delta_i| r_e$
- Claim 2: $\sum_{e=(i,j)} |\Delta_i| r_e \ge (1-\varepsilon)\, d\|\Delta\|_1$

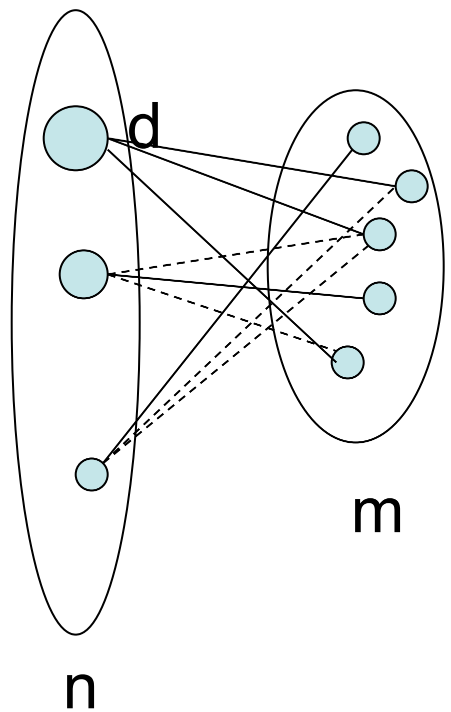


$\sum_{e=(i,j)} |\Delta_i| r_e \geq (1-\varepsilon)\, d ||\Delta||_1$

- Let $z_e = |\Delta_i|$ for $e=(i,j)$
- Lower bound $\sum_e z_e r_e$
- What do we know about z_e, r_e
  - z_e is non-increasing
    (from the reordering)
  - In any prefix of r of length ds, there are $<\varepsilon/2\ d\ s$ minuses
    (from the expansion)
  - There are no minuses in the first d elements of r
    (the first node has no overlapping edges)
- The worst-case sequence r is as follows (in blocks of length d):

$$\underbrace{+,+,\ldots,+,}_{0 \text{ minuses}}\quad \underbrace{-,\ldots,-,+,\ldots,+\ ,}_{\varepsilon/2\ d \text{ minuses}}\quad \underbrace{,-,\ldots,-,+,\ldots,+\ ,}_{\varepsilon/2\ d \text{ minuses}}\ \ldots\ldots$$

■

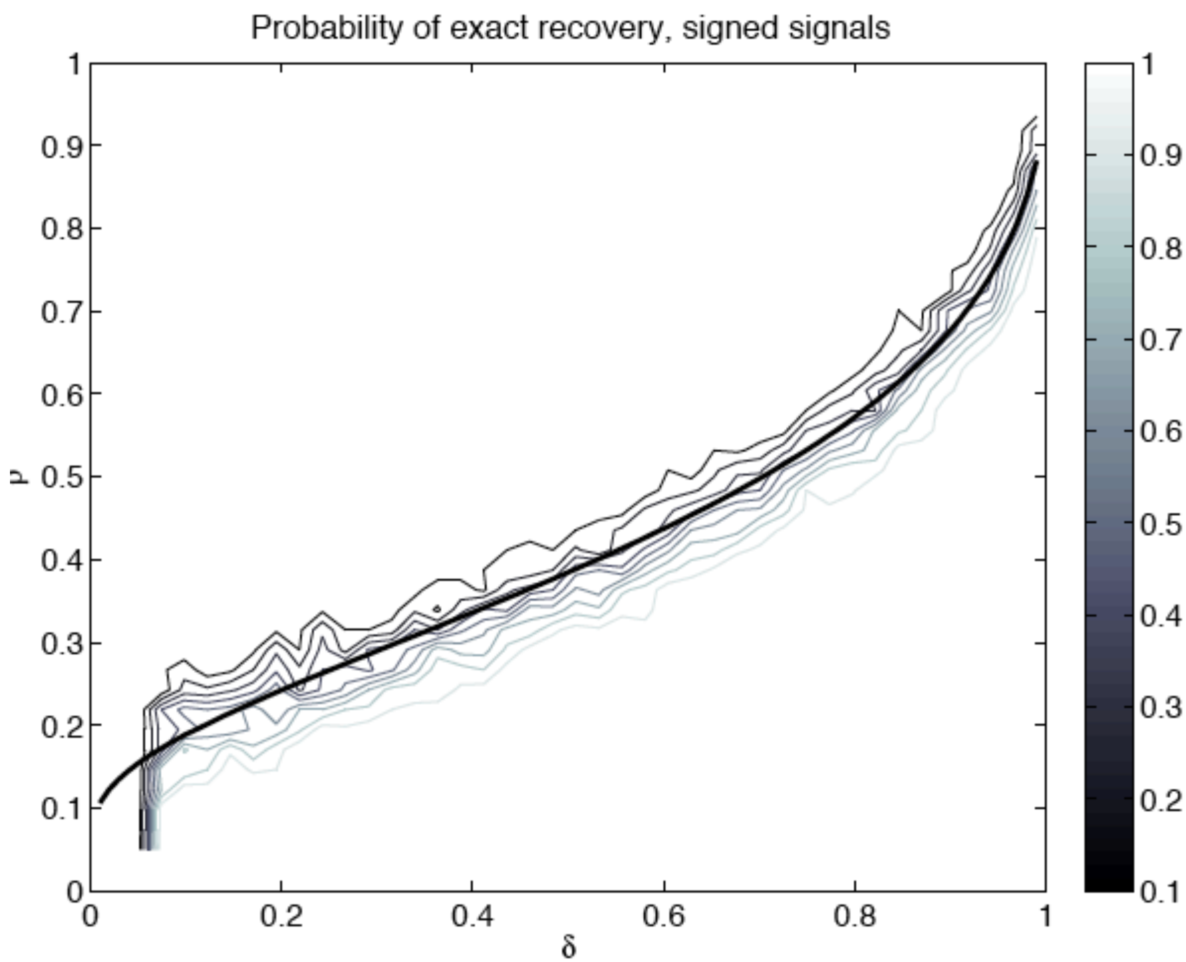


Implication of RIP-1

- We have RIP in the L1 norm
- Now what ?
- We will show that RIP-1 also implies the null space property (notes), and therefore L1 minimization works
- Altogether, this gives a a scheme with:
  - $O(k \log (n/k))$ measurements
  - Recovery time still poly(n)
  - However, thanks to the use of sparse matrices, the running time is faster

    (the linear program iteratively computers Az for vectors z, which is fast if A is sparse)

Sharp transition



References

- Survey:

  A. Gilbert, P. Indyk, “Sparse recovery using sparse matrices”, Proceedings of IEEE, June 2010.