# Sparse recovery using sparse random matrices

## Or: Fast and Effective Linear Compression

Piotr Indyk

MIT

Joint work with: R. Berinde, A. Gilbert, H. Karloff, M. Ruzic, M. Strauss

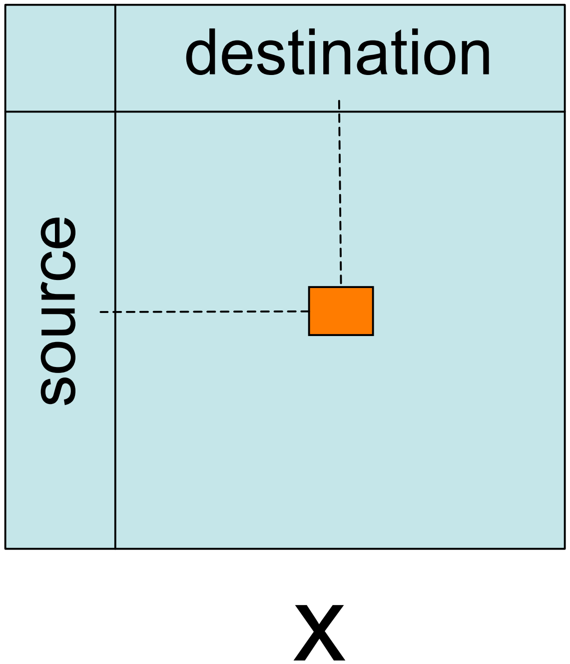
# Linear Compression

- High-dimensional data: $x$
- Low-dimensional sketch: $Ax$
- Goal: design $A$ so that given $Ax$ we can recover an "approximation" $x^*$ of $x$
  - Sparsity parameter $k$
  - Want $x^*$ such that $||x^*-x||_p \leq C \, ||x'-x||_q$

    over all $x'$ that are $k$-sparse (at most $k$ non-zero entries)
  - The best $x'$ contains $k$ coordinates of $x$ with the largest abs value
- Short history:
  - Learning (Fourier coefficients)
    - Fourier matrices, algebraic methods
  - Streaming (Heavy hitters)
    - Mostly sparse binary matrices, combinatorial methods
  - Compressed sensing
    - Dense matrices (Gaussian, Fourier), geometric methods

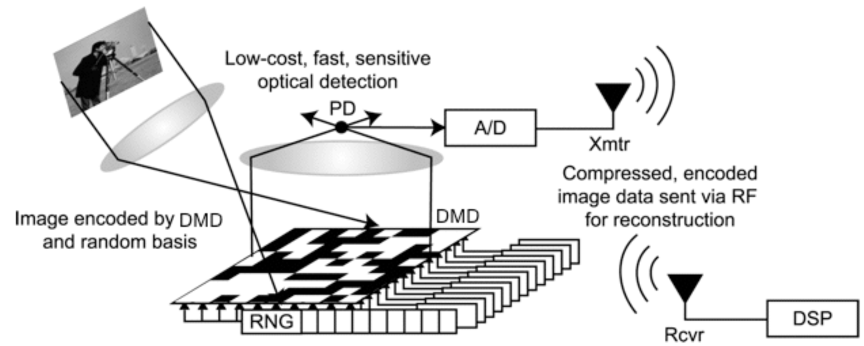
# Application I: Monitoring Network Traffic

- Router routs packets (many packets)
  - Where do they come from ?
  - Where do they go to ?
- Ideally, would like to maintain a traffic matrix x[.,.]
  - Easy to update: given a (src,dst) packet, perform $x_{src,dst}$++
  - Requires way too much space! ($2^{32}$ x $2^{32}$ entries)
  - Need to compress x, increment easily
- Using linear compression we can:
  - Maintain sketch Ax under increments to x, since $A(x+\Delta) = Ax + A\Delta$
  - Recover x* from Ax



# Other applications

- Single pixel camera




- High throughput screening (Anna, Sat, 3:30 pm)

- …

# Parameters

- Given: dimension $n$, sparsity $k$
- Parameters:
  - Sketch length $m$
  - Time to compute/update $Ax$
  - Time to recover $x^*$ from $Ax$
  - Randomized/Deterministic/Explicit matrix $A$
  - Measurement noise, universality, …

# Results

(best known in blue)

| Paper | A/E | Sketch length | Encode time | Column sparsity/ Update time | Decode time | Approx. error | Noise |
|---|---|---|---|---|---|---|---|
| [CCFC02, CM06] | E<br>E | $k \log^c n$<br>$k \log n$ | $n \log^c n$<br>$n \log n$ | $\log^c n$<br>$\log n$ | $k \log^c n$<br>$n \log n$ | $\ell_2 \leq C\ell_2$<br>$\ell_2 \leq C\ell_2$ | |
| [CM04] | E<br>E | $k \log^c n$<br>$k \log n$ | $n \log^c n$<br>$n \log n$ | $\log^c n$<br>$\log n$ | $k \log^c n$<br>$n \log n$ | $\ell_1 \leq C\ell_1$<br>$\ell_1 \leq C\ell_1$ | |
| [CRT06] | A<br>A | $k \log(n/k)$<br>$k \log^c n$ | $nk \log(n/k)$<br>$n \log n$ | $k \log(n/k)$<br>$k \log^c n$ | LP<br>LP | $\ell_2 \leq \frac{C}{k^{1/2}}\ell_1$<br>$\ell_2 \leq \frac{C}{k^{1/2}}\ell_1$ | Y<br>Y |
| [GSTV06] | A | $k \log^c n$ | $n \log^c n$ | $\log^c n$ | $k \log^c n$ | $\ell_1 \leq C \log n \ell_1$ | Y |
| [GSTV07] | A | $k \log^c n$ | $n \log^c n$ | $\log^c n$ | $k^2 \log^c n$ | $\ell_2 \leq \frac{C}{k^{1/2}}\ell_1$ | |
| [NV07] | A<br>A | $k \log(n/k)$<br>$k \log^c n$ | $nk \log(n/k)$<br>$n \log n$ | $k \log(n/k)$<br>$k \log^c n$ | $nk^2 \log^c n$<br>$nk^2 \log^c n$ | $\ell_2 \leq \frac{C(\log n)^{1/2}}{k^{1/2}}\ell_1$<br>$\ell_2 \leq \frac{C(\log n)^{1/2}}{k^{1/2}}\ell_1$ | Y<br>Y |
| [GLR08] (k "large") | A | $k(\log n)^{c \log\log\log n}$ | $kn^{1-\alpha}$ | $n^{1-\alpha}$ | LP | $\ell_2 \leq \frac{C}{k^{1/2}}\ell_1$ | |
| This talk | A | $k \log(n/k)$ | $n \log(n/k)$ | $\log(n/k)$ | LP | $\ell_1 \leq C\ell_1$ | Y |

March'08

| Paper | A/E | Sketch length | Encode time | Update time | Decode time | Approx. error | Noise |
|---|---|---|---|---|---|---|---|
| [DM08] | A | $k \log(n/k)$ | $nk \log(n/k)$ | $k \log(n/k)$ | $nk \log(n/k) \log D$ | $\ell_2 \leq \frac{C}{k^{1/2}}\ell_1$ | Y |
| [NT08] | A<br>A | $k \log(n/k)$<br>$k \log^c n$ | $nk \log(n/k)$<br>$n \log n$ | $k \log(n/k)$<br>$k \log^c n$ | $nk \log(n/k) \log D$<br>$n \log n \log D$ | $\ell_2 \leq \frac{C}{k^{1/2}}\ell_1$<br>$\ell_2 \leq \frac{C}{k^{1/2}}\ell_1$ | Y<br>Y |
| This talk | A | $k \log(n/k)$ | $n \log(n/k)$ | $\log(n/k)$ | $n \log(n/k)$ | $\ell_1 \leq C\ell_1$ | Y |

# General approach

- Dichotomy:
  - Sparse matrices: faster algorithms
  - Dense metrices: shorter sketches
- Approach:
  - Unify
  - Best of both worlds

# Dense matrices: ideas

- Restricted Isometry Property [Candes-Tao]:
  - A satisfies $(k,C)$-RIP if for all $k$-sparse vectors $x$

$$||x||_2 \le ||Ax||_2 \le C\ ||x||_2$$

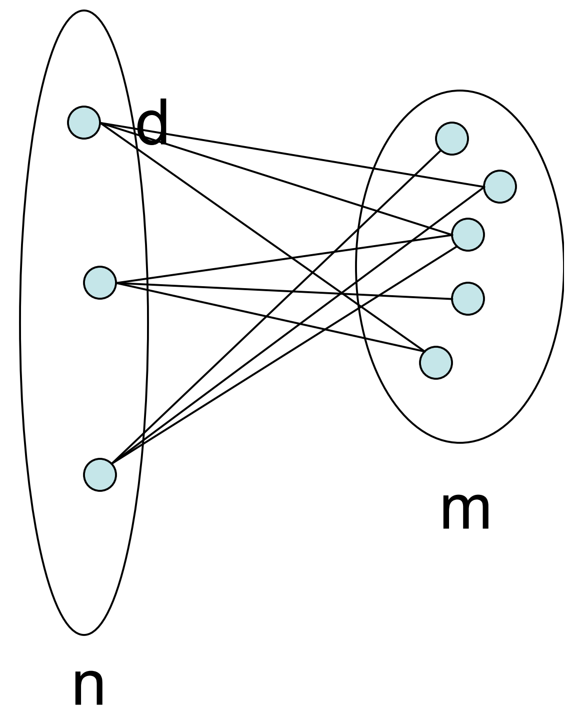
- Examples:
  - Random Gaussian: $m=O(k \log (n/k))$
  - Random Fourier: $m=O(k \log^{O(1)} n)$
- Recovery algorithms:
  - Linear Programming :
    - Find $x^*$ such that $Ax=Ax^*$ and $||x^*||_1$ minimal
  - Orthogonal Matching Pursuit:
    - Iteratively find large coordinates of the residual $x-x^*$
    - Update $x^*$
- Both rely on RIP

# Dealing with Sparsity

- Consider "random" $m \times n$ adjacency matrices of $d$-regular bipartite graphs
- Do they satisfy RIP ?
  - No, unless $m=\Omega(k^2)$ [Chandar'07]
- However, they do satisfy the following RIP-1 property: for any $k$-sparse $x$

$$d\,(1-2\varepsilon)\,||x||_1 \leq ||Ax||_1 \leq d||x||_1$$

  if the graph is a $(\,k,\, d(1-\varepsilon)\,)$-expander
  [Berinde-Gilbert-Indyk-Karloff-Strauss'08]
  - Randomized: $m=O(k \log (n/k))$
  - Explicit: $m=k$ quasipolylog $n$
- What is the use of RIP-1 ?



# $A$ satisfies RIP-1 $\Rightarrow$ LP works

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

- Compute a vector $x^*$ such that $Ax=Ax^*$ and $||x^*||_1$ minimal
- Then we have, over all k-sparse $x'$

$$||x-x^*||_1 \leq C \min_{x'} ||x-x'||_1$$

  - $C \rightarrow 2$ as the expansion parameter $\varepsilon \rightarrow 0$
- Can be extended to the case when $Ax$ is noisy

# A satisfies RIP-1 $\Rightarrow$ OMP works

[Indyk-Ruzic'08]

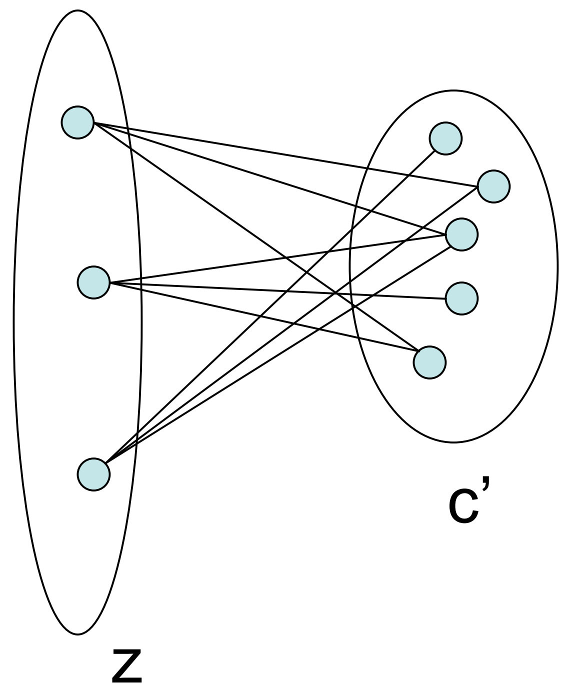
- Algorithm I: Expander Matching Pursuit
  - Very fast running time $O(n \log(n/k))$
  - Uses multiple parameters
- Algorithm II (new): "Sparse Matching Pursuit" (influenced by [Needell-Tropp'08] )
  - Slower running time of $O(n \log D \log(n/k))$
  - Only one parameter $k$

# "Sparse Matching Pursuit"

- Algorithm:
  - $x^*=0$
  - Repeat $T$ times
    - Let $c' = Ax - Ax^* = A(x-x^*)$
    - Compute $z$ such that $z_i$ is the median of its neighbors in $c$
    - $x^*=x^*+z$
    - Sparsify $x^*$

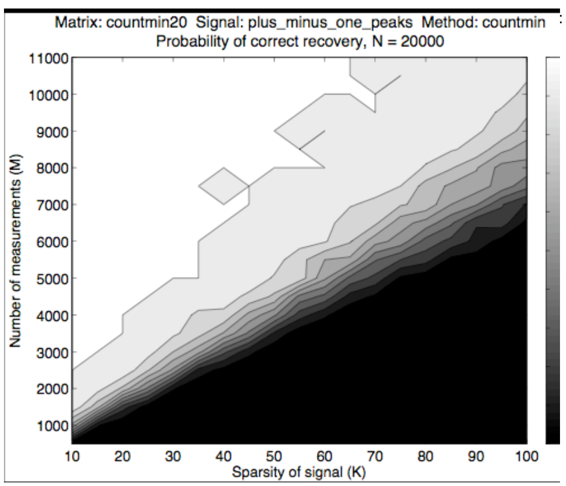
      (set all but $k$ largest entries of $x^*$ to 0)
- After $T=O(\log D)$ steps we have, over all $k$-sparse $x'$

$$||x-x^*||_1 \leq C \min_{x'} ||x-x'||_1$$



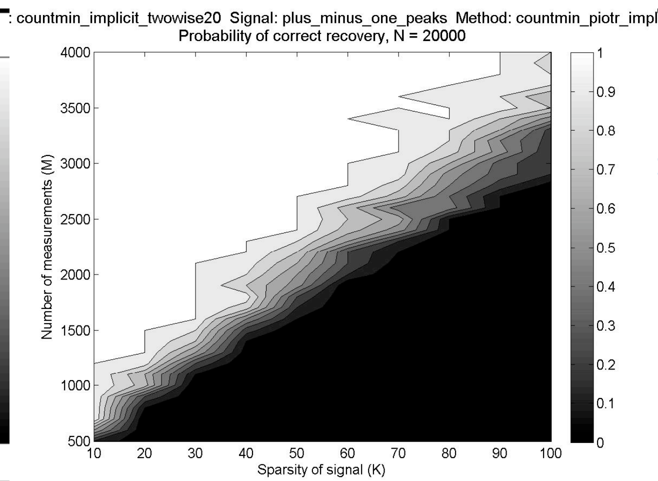
# Experiments

- Probability of recovery of random k-sparse +1/-1 signals from m measurements
  - Sparse matrices with d=20 1s per column
  - Signal length n=20,000

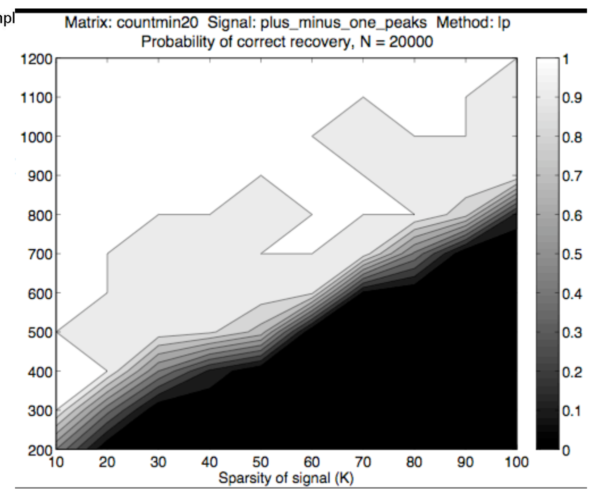



Countmin

[Cormode-Muthukrishnan'04]




Sparse Matching Pursuit (20 iterations)




Linear Programming

Same as for Gaussian matrices!

# Conclusions

- Sparse approximation possible with sparse matrices:
  - RIP-1 vs. expansion
  - Unify geometric and combinatorial view
- State of the art: can do 2 out of 3:
  - Near-linear encoding/decoding
  - O(k log (n/k)) measurements
  
  } This talk
  - Approximation guarantee with respect to L2 norm
- Open problems:
  - 3 out of 3 ?
  - Precise understanding ?
  - Further applications ?

# Resources

- References:
  - R. Berinde, A. Gilbert, P. Indyk, H. Karloff, M. Strauss, “Combining geometry and combinatorics: a unified approach to sparse signal recovery”, 2008.
  - R. Berinde, P. Indyk, “Sparse Recovery Using Sparse Random Matrices”, 2008.
  - P. Indyk, M. Ruzic, “Near-Optimal Sparse Recovery in the L1 norm”, 2008.