UNIVERSITÄT DES SAARLANDES

max planck institut informatik

Sublinear Algorithms

Lecture 05: Measurements II

European Research Council
Established by the European Commission

Previous lecture:

Combinatorial Group Testing

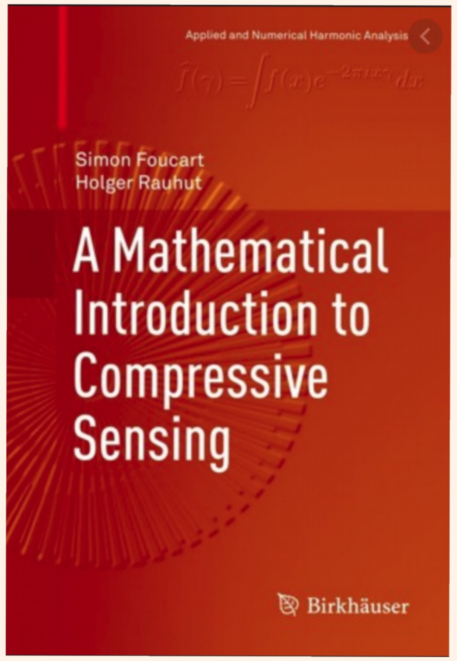
or

Sparse Recovery from Disjunctive *Measurements*

This lecture

Sparse Recovery from Linear Measurements

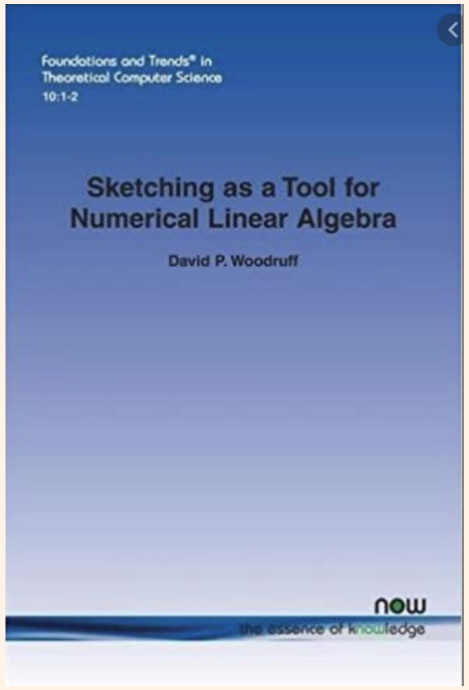
Linear Sparse Recovery



Design a matrix (linear sketch) *M*,
such that given

$$y = Mx, x \in \mathbb{R}^n$$

You may recover *x* if it is *k*-sparse



Constraints:

-Belong to a specific ensemble (signal processing)

- Have a compact representation (streaming)

- Allow fast matrix-vector multiplication (numerical linear algebra)

This lecture
Exactly *k*-sparse
and
Unconstrained Case

Guarantees

(*similarly to Group Testing*)

$$\mathrm{supp}(x) = \{i \in [n] : x_i \neq 0\} \qquad \Sigma_{k,n} = \{x \in \mathbb{R}^n : |\mathrm{supp}(x)| \leq k\}$$

$$M \in \mathrm{R}^{m \times n}$$

$$y = Mx, x \in \Sigma_{k,n}$$

Uniform Sparse Recovery → One Matrix to rule them all

$$\Pr\{\exists x \in \Sigma_{k,n} : \textsc{Find}(y) \neq x\} \leq \delta$$

Non-Uniform Sparse Recovery → One Matrix with high probability for each

$$\forall x \in \Sigma_{k,n} : \Pr\{\textsc{Find}(y) \neq x\} \leq \delta$$

Uniform Sparse Recovery

When M does not suffice for sparse recovery?
Let x, x' be k-sparse:

$$Mx = Mx' \longrightarrow M(x - x') = 0 \longrightarrow \underbrace{x - x'}_{2k-\text{sparse}} \in \ker(M)$$

Suffices that every *m by 2k* submatrix is invertible

Vandermorde matrix: $\begin{vmatrix} 1 & a & b & c \\ 1 & a^2 & b^2 & c^2 \\ 1 & a^3 & b^3 & c^3 \\ 1 & a^4 & b^4 & c^4 \end{vmatrix}$

a,b,c are different : Vandermorde matrix is invertible

Pick an invertible n x n Vandermorde matrix
Keep the first *2k* rows

2k x *2k* submatrix is invertible*: 2k* measurements suffice

-Beyond the $k \log n$ bound, in contrast to group testing
-entries of M might need way too many bits to write down!

Non-Uniform Guarantee

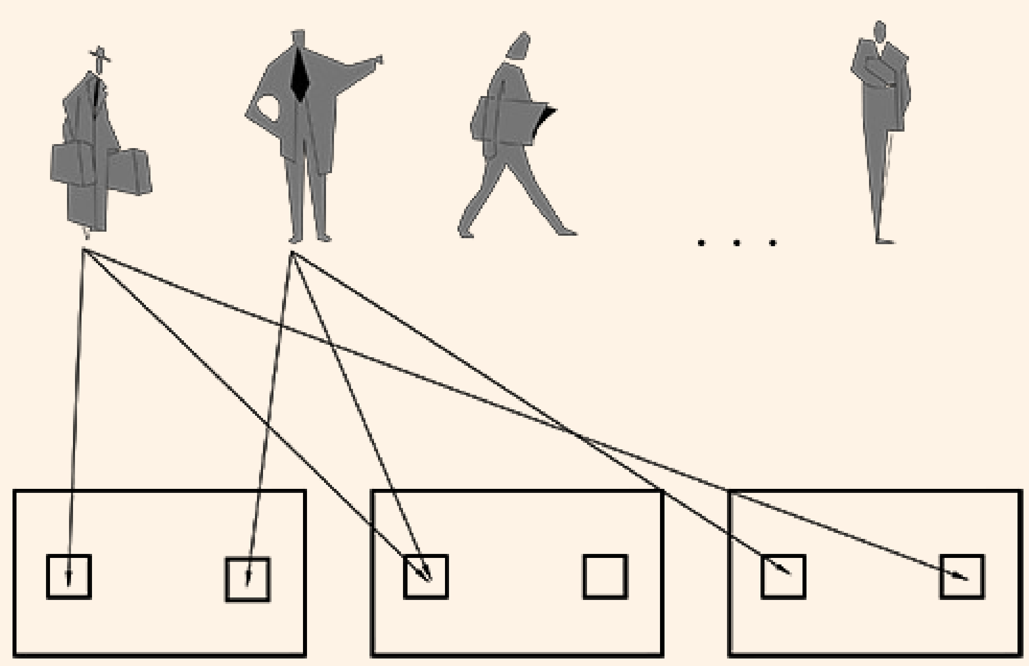
Pick (random) hash functions $h_r : [n] \to [4k]$ for $r \in [R]$

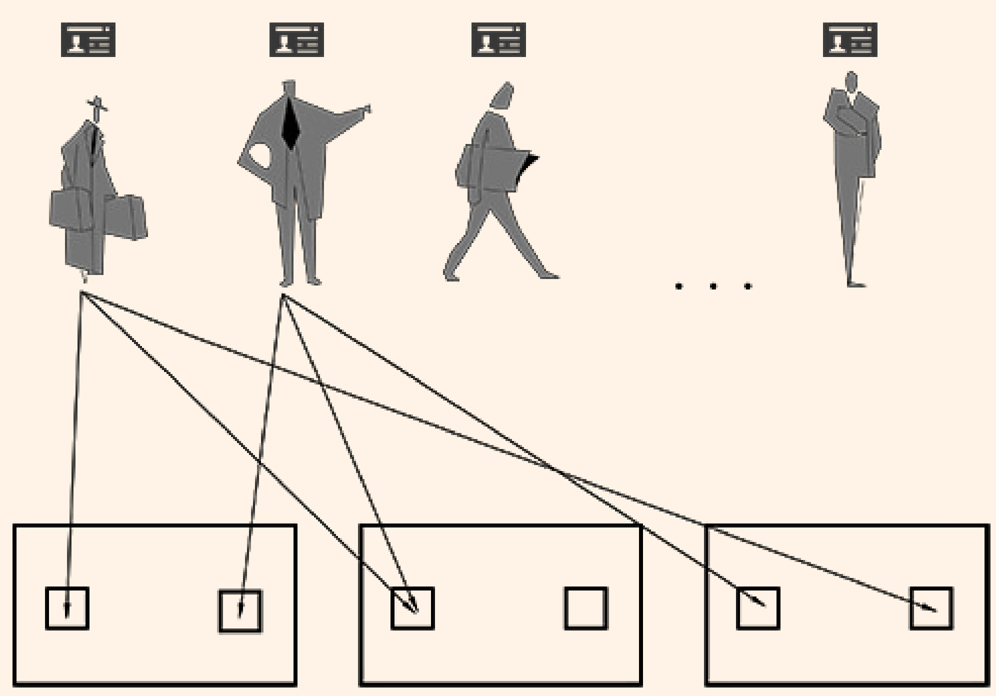
$$\forall (b, r) \in [4k] \times [R]$$

perform measurements

Incorporate the identity of each element $\displaystyle\sum_{i:h_r(i)=b} x_i$

$$\sum_{i:h_r(i)=b} i \cdot x_i$$





Pick (random) hash functions $h_r : [n] \to [4k]$ for $r \in [R]$

$$\sum_{i:h_r(i)=b} x_i, \sum_{i:h_r(i)=b} i \cdot x_i$$

1-sparse at $i^\star$ → Dividing the values of the measurements in which is hashed to:

$$\frac{i^\star \cdot x_{i^\star}}{x_{i^\star}} = i^\star$$

Reduce *k*-sparse to *1*-sparse case

Fix *i* with non-zero x_i.

$$\Pr\{h_r(i) = h_r(i')\} = \frac{1}{4k}$$

↓

Probability *of i participating in a non -1-sparse instance:* $\frac{k}{4k} = \frac{1}{4}$

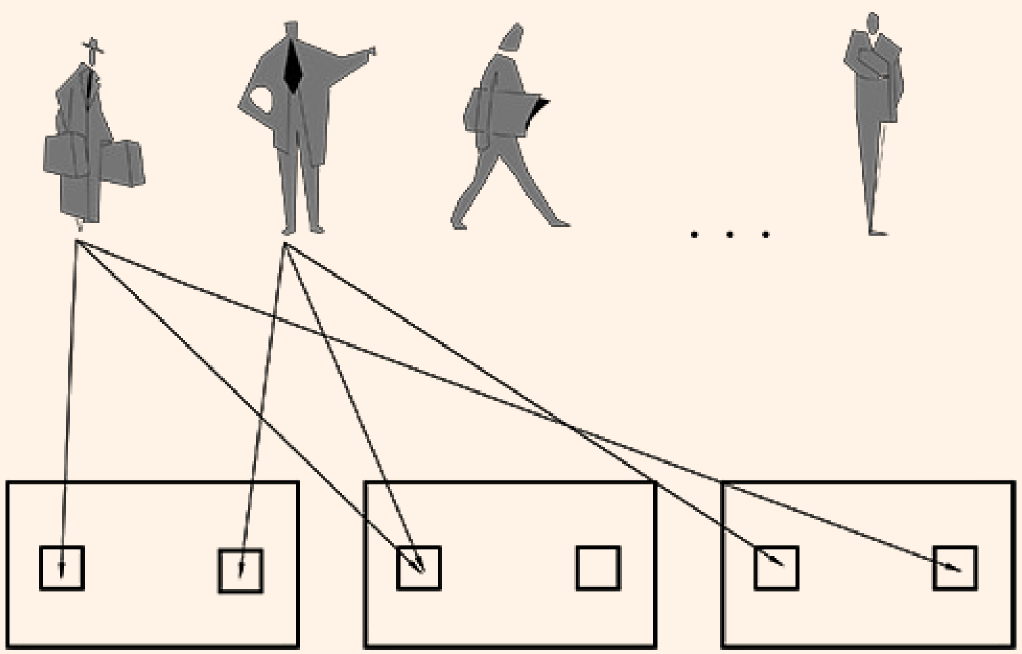
↓

$R = \Theta(\log k)$

Probability i participating in a non-1-sparse instance in most repetitions $\frac{1}{\text{poly}(k)}$

→ All elements in the support will be recovered in at least half of reps

What about false positives?



$$x_2 + x_4, 2 \cdot x_2 + 4 \cdot x_4$$

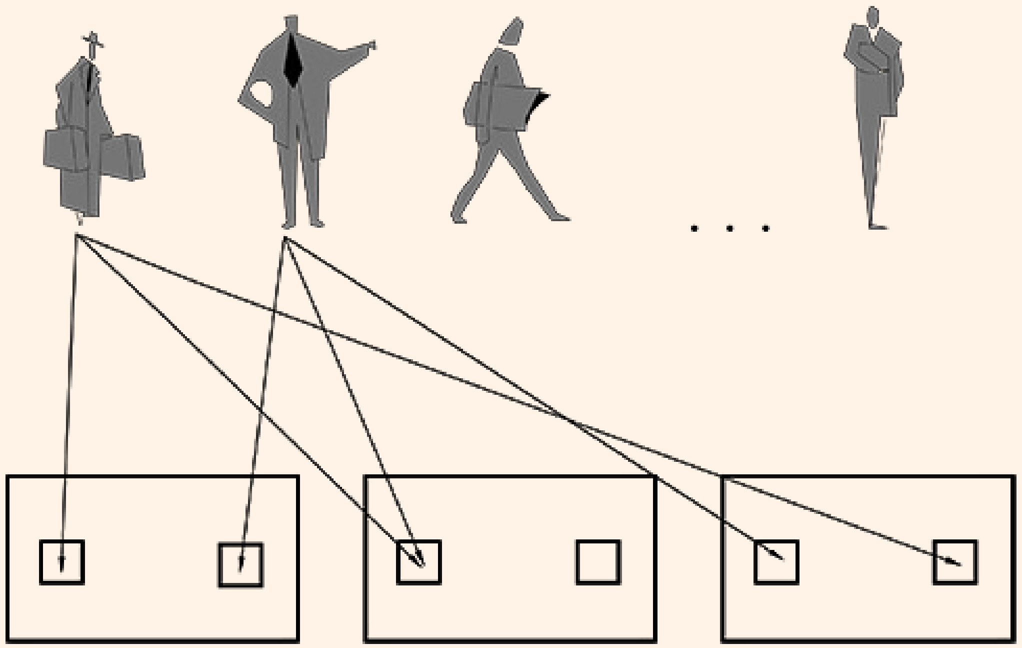
$$\frac{2 \cdot x_2 + 4 \cdot x_4}{x_2 + x_4} = \frac{2+4}{1+1} = 3$$

At the end, at most 8k elements *i* in hand:
A superset of the support

Keep on the side another linear sketch:
Count-Min with R= Θ(log k))

Query only those 8k elements
(in the problem set?)

Putting it together



#Measurements: O(k log k)

Running Time: O(k log k)

Storing down the hash functions: O(log k) words
(pairwise independence suffices)

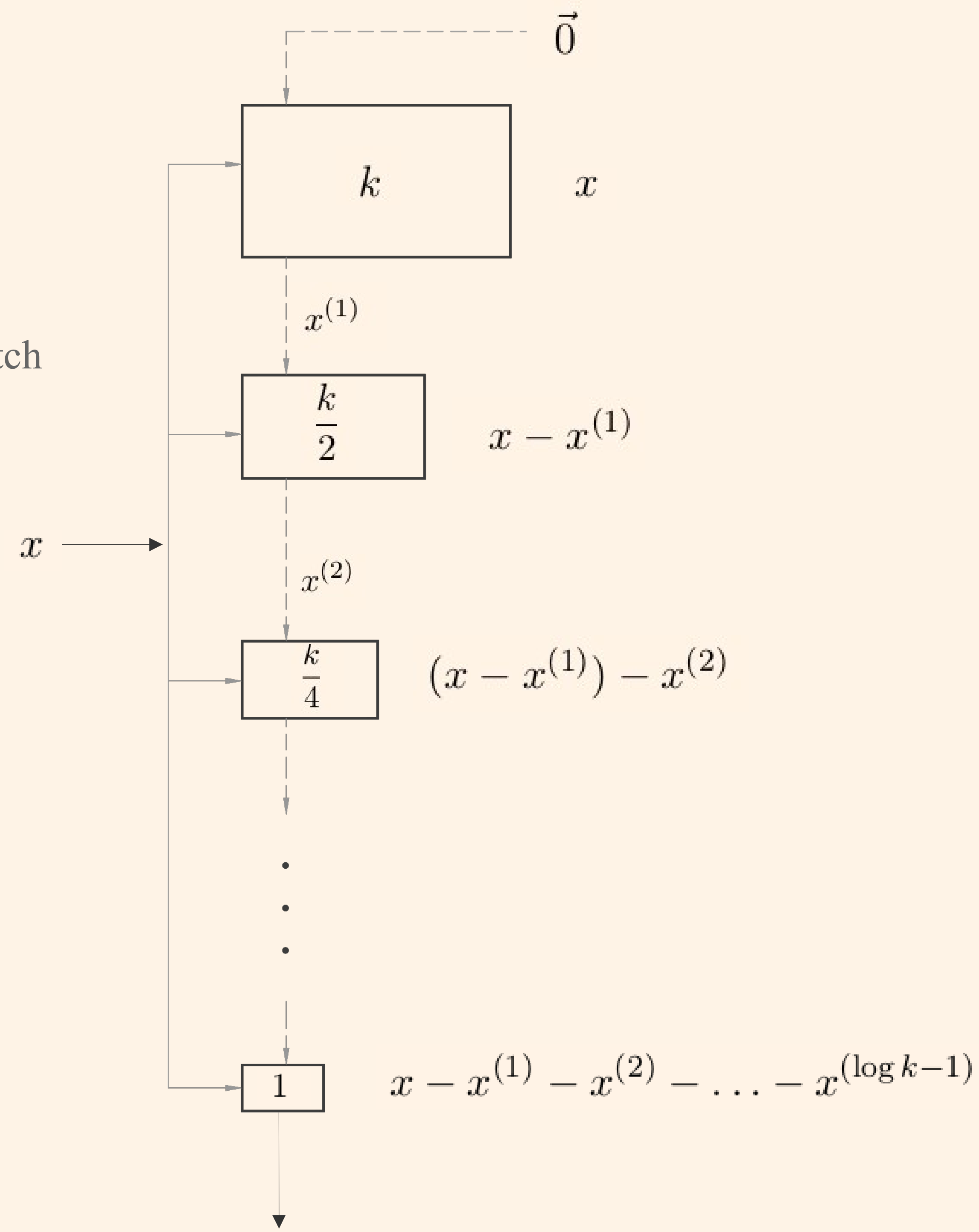
Update Time/Column Sparsity: O(log k)

O(k) measurements suffice

(quick overview)

Set R= O(1) instead of Θ(log k)

Feed the output of every linear sketch to the next one



Thank you