Sublinear Time Recovery of Sparse Signals

Charles J. Colbourn⋆[1],
Daniel Horsley[2], Violet R. Syrotiuk[1]

[1]Arizona State University [2]Monash University

Network Mapping and Measurement, 2012

Outline

Compressive sensing

- A signal $\mathbf{x}$, which is a vector in $\mathbb{R}^k$, having at most t nonzero coordinates.
- A *sample* is a vector of weights $\mathbf{w} \in \mathbb{R}^k$, for which the sample measurement is $\mathbf{w}\mathbf{x}^T$.
- Goal: Construct a set of N samples so that the unknown signal $\mathbf{x}$ can be recovered from the sample measurements. The $N \times k$ matrix so formed is a *measurement matrix*.
- (Admittedly, this is an overly simplified version of compressive sensing!)

Compressive sensing

- Crucial issues include:
  - the time required to recover the signal
  - the ability to deal with noise

Column Replacement

- Suppose that we have
  - measurement matrices $A_1, \ldots A_m$. A_i has k_i columns, M_i rows, and can recover signals of sparsity s_i using recovery method $\mathcal{R}_i$.
  - a pattern matrix $P = (p_{ij})$ that is $m \times n$, in which the ith row contains (at most) k_i distinct symbols.
- *column replacement* of $(A_1, \ldots, A_m)$ into P:
  - Suppose that the k_i entries of row i of P index the k_i columns of A_i.
  - Replace p_{ij} by a copy of the column of A_i that is indexed by p_{ij}.
  - The result is a $(\sum_{i=1}^{m} M_i) \times n$ matrix.

Pattern Matrices

- What properties should the pattern matrix have to ensure that the column replacement matrix supports recoverability for a certain sparsity?

Separating and Distributing Hash Families
Definition

- Let $\mathbf{k} = (k_1, \ldots, k_m)$.
- A *separating hash family* $\mathrm{SHF}(m; n, \mathbf{k}, \{w_1, \ldots, w_s\})$ with $t = \sum_{i=1}^{s} w_i$ is an $m \times n$ array in which the ith row contains k_i symbols, and in which in every $m \times t$ subarray, and every way to partition the t columns into classes of sizes $w_1, \ldots, w_s$, there is at least one row in which symbols in different classes are different.
- A *distributing hash family* $\mathrm{DHF}(m; n, \mathbf{k}, t, s)$ is an $\mathrm{SHF}(m; n, \mathbf{k}, \{w_1, \ldots, w_s\})$ for *every* way to choose $w_1 + \cdots + w_s = t$.

An Example

DHF$(10; 13, (9, 9, 9, 5, 4, 3, 3, 3, 3, 2), 5, 2)$

$$
\begin{bmatrix}
6 & 7 & 8 & 3 & 4 & 0 & 2 & 2 & 3 & 0 & 5 & 1 & 1 \\
3 & 1 & 1 & 7 & 2 & 6 & 8 & 4 & 3 & 0 & 2 & 0 & 5 \\
8 & 5 & 1 & 4 & 2 & 3 & 2 & 6 & 7 & 0 & 1 & 3 & 0 \\
0 & 0 & 3 & 0 & 1 & 0 & 0 & 2 & 4 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 & 0 & 3 & 2 & 0 & 2 & 0 & 1 & 0 & 2 \\
0 & 2 & 0 & 2 & 2 & 0 & 0 & 1 & 1 & 1 & 1 & 2 & 0 \\
0 & 0 & 2 & 1 & 1 & 1 & 2 & 0 & 0 & 2 & 2 & 0 & 1 \\
1 & 1 & 2 & 2 & 2 & 0 & 1 & 0 & 0 & 2 & 1 & 0 & 0 \\
1 & 0 & 1 & 2 & 0 & 0 & 2 & 0 & 0 & 1 & 2 & 2 & 1 \\
0 & \star & \star & \star & \star & 1 & \star & \star & 1 & \star & \star & 0 & 1
\end{bmatrix}
$$

$\star$ means you can select any element.

An Example

DHF(10; 13, (9, 9, 9, 5, 4, 3, 3, 3, 3, 2), 5, 2)

Look at separating columns $\{1, 12\}$ from $\{6, 9, 13\}$:

↓					↓			↓			↓	↓
6	7	8	3	4	0	2	2	3	0	5	1	1
3	1	1	7	2	6	8	4	3	0	2	0	5
8	5	1	4	2	3	2	6	7	0	1	3	0
0	0	3	0	1	0	0	2	4	0	0	1	0
1	1	0	1	0	3	2	0	2	0	1	0	2
0	2	0	2	2	0	0	1	1	1	1	2	0
0	0	2	1	1	1	2	0	0	2	2	0	1
1	1	2	2	2	0	1	0	0	2	1	0	0
1	0	1	2	0	0	2	0	0	1	2	2	1
0	$\star$	$\star$	$\star$	$\star$	1	$\star$	$\star$	1	$\star$	$\star$	0	1

Strenghtening Hash Families

Definition

- A $((d_1, \ldots, d_m), \tau)$-*strengthening* separating hash family SHF$(m; n, \mathbf{k}, \{w_1, \ldots, w_s\})$ with $t = \sum_{i=1}^{s} w_i$ is an SHF$(m; n, \mathbf{k}, \{w_1, \ldots, w_s\})$ in which in every set C of t columns, for every way to partition C into classes of sizes $w_1, \ldots, w_s$ and for every subset T of τ columns with $|C \cap T| = \min(t, \tau)$, there is at least one row ρ in which the symbols in different classes are different and on the columns of T at most d_ρ different symbols appear .
- The extension to DHFs is immediate.

Compressive Sensing
Recoverability

Charles J. Colbourn*, Daniel Horsley, Violet R. Syrotiuk

- A measurement matrix A *has* (ℓ_0, t)*-recoverability* when it permits exact recovery of $\mathbf{x}$ using $A\mathbf{x} = \mathbf{b}$, given A and $\mathbf{b}$, and the fact that $\mathbf{x}$ is t-sparse.
- A measurement matrix A *has* (ℓ_1, t)*-recoverability* when, for each t-sparse signal $\mathbf{x}$, $\mathbf{x}$ is the unique solution to $\min\{||\mathbf{z}||_1 : A\mathbf{z} = A\mathbf{x}\}$.

Compressive Sensing

Hash Families for ℓ_0

Charles J. Colbourn⋆, Daniel Horsley, Violet R. Syrotiuk

Theorem

Let $\mathbf{k} = (k_1, \ldots, k_m)$ and $\mathbf{q} = (q_1, \ldots, q_m)$ be tuples of positive integers. Let $\mathbf{d} = (2q_1, \ldots, 2q_m)$. For $1 \leq i \leq m$, let $A_i \in \mathbb{R}^{r_i \times k_i}$ be a measurement matrix that meets the (ℓ_0, q_i)-null space condition. Let P be a $(\mathbf{d}, 2t)$-strengthening SHF$(m; n, \mathbf{k}, (1, t))$, and let B be the column replacement of $A_1, A_2, \ldots, A_m$ into P. Then B meets the (ℓ_0, t)-null space condition.

Compressive Sensing

Hash Families for ℓ_1

Theorem

Let $\mathbf{k} = (k_1, \ldots, k_m)$ *and* $\mathbf{q} = (q_1, \ldots, q_m)$ *be tuples of positive integers. For* $1 \leq i \leq m$, *let* $A_i \in \mathbb{R}^{r_i \times k_i}$ *be a measurement matrix that meets the* (ℓ_1, q_i)*-null space condition. Let* P *be a* $(\mathbf{q}, t)$*-strengthening* DHF$(m; n, \mathbf{k}, t+1, 2)$*, and let* B *be the column replacement of* $A_1, A_2, \ldots, A_m$ *into* P*. Then* B *meets the* (ℓ_1, t)*-null space condition.*

Using arbitrary recovery methods

- This is much more general than ℓ_0- and ℓ_1-recovery.
- It can use arbitrary measurement matrices $\mathsf{A}_1, \ldots, \mathsf{A}_m$ as ingredients, each having a possibly different recovery method $\mathcal{R}_i$, $i = 1, \ldots, m$.
- Although we can do somewhat better when signals are restricted to be positive (and noise is positive), we explore the general case.

Column Replacement

- We are to support recovery of t-sparse signals in $\mathbb{R}^n$.
- P is an $(\mathbf{d}, t)$-strengthening SHF$(m; n, \mathbf{k}, \{\{\tau, (t+1-\tau)\} : 1 \leq \tau \leq t\})$.
- For each $1 \leq i \leq m$, A_i is an $r_i \times k_i$ measurement matrix, equipped with a recovery algorithm $\mathcal{R}_i$, which determines the d_i-sparse vector $\mathbf{z}_i$ that solves $A_i\mathbf{z}_i = \mathbf{y}_i$.
- B is the column replacement of $\mathrm{A}_1, \ldots, \mathrm{A}_m$ into P; it has *bands* $\mathrm{B}_1, \ldots, \mathrm{B}_m$ where B_i is the column replacement of A_i into the ith row of P.

Column Replacement

The result

Theorem

Let $\mathbf{z} \in \mathbb{R}^n$. Then the t-sparse solution $\mathbf{y}$ to $\mathrm{B}\mathbf{y} = \mathbf{z}$ can be recovered, given an oracle for $\mathcal{R}_i$ for $1 \leq i \leq m$.

Recovery

The setup

- Use each row of the matrix P to form a partitions of the column indices, which we take to be $\{1, \ldots, n\}$, as follows.
- For $1 \leq i \leq m$, partition $\{1, \ldots, n\}$ into classes $C_{i1}, \ldots, C_{i,k_i}$ so that j and ℓ are in the same class if and only if $p_{ij} = p_{i\ell}$.
- Let $S_{ij} = \{x_\ell : \ell \in C_{ij}\}$.

Recovery

The mechanics

- Now we find the coordinates in which the signal is positive, then those in which it is negative, and the remainder are zero.
- To find the positive ones,
  - For $1 \leq i \leq m$, determine the solution $\mathbf{z}_i$ to $A_i\mathbf{z}_i = \mathbf{y}_i$ via method $\mathcal{R}_i$.
  - Let $\mathbf{z}_i^+ = (\max(0, z_{ij}) : 1 \leq j \leq k_i)$.
  - Call a band i *maximum positive* if $||\mathbf{z}_i^+||_1$ attains the largest value.
  - Let M index the maximum positive bands.
  - Then it can be shown that a coordinate x_ℓ is positive if and only if when $x_\ell \in S_{\rho j}$, $z_{\rho j} > 0$.

Recovery

Observations

- The ingredient measurement matrices
  - need not contain entries from the same element set,
  - need not support recovery in the same manner,
  - need not support the same sparsity of signals, and
  - need not treat signals of the same lengths.
- The sparsity supported by the column replacement matrix can be much larger than that supported by the ingredients.

Recovery
Observations

Charles J. Colbourn⋆, Daniel Horsley, Violet R. Syrotiuk

- In particular, for a 'small' ingredient matrix we can afford to use a much more computationally intensive recovery method, and thereby hope to obtain more substantial compression.
- A major problem is that, even once all the small recoveries are done, it appears that we must examine every coordinate to determine which are significant. Can we do better?

Sublinear Time Recovery

A sketch

- The idea is to identify the significant coordinates from the positive (negative) sets in maximum positive (negative) bands *without examining each coordinate*.
- To do this,
  - Let P_i be the union of all positive sets in band $i \in M$.
  - Then we are to determine all coordinates in $\bigcap_{i \in M} P_i$.
- We do not want to list the members of P_i, or to explicitly compute the intersection.
- So we define the sets *implicitly*.

Sublinear Time Recovery

Some details

- We form a distributing hash family whose columns are indexed by polynomials of low degree over the finite field, and whose rows are indexed by elements of the finite fields.
- The entries are the evaluation of the column polynomial at the point given by the row index.
- (These *linear* hash families have been extensively studied by Simon Blackburn and Peter Wild.)

Sublinear Time Recovery

Some details

- The classes formed in row ρ partition the polynomials according to their value when evaluated at ρ.
- Given the list of all positive classes in maximum positive bands, we reduce to solving many systems of linear equations over the field to determine which columns yield only positive classes.
- When the order of the field and the degree of the polynomials are varied, we obtain schemes for different levels of sparsity and different compression ratios.

Sublinear Time Recovery

Some details

- Most importantly, we identify significant coordinates without ever listing the members of any class, and so we obtain recovery times that are sublinear in the number of coordinates.

Remaining Issues

- Resilience to noise can be easily incorporated in the ingredient matrices, but remains a concern for the column replacement matrix.
- We have some preliminary results in this direction, but as yet each requires additional constraints on the hash families employed.